An Analysis of Computer Systems for the Secure Creation and Verification of User Instructions

Andreas Gutmann

A dissertation submitted in partial fulfillment

of the requirements for the degree of

Doctor of Philosophy

of

University College London.

Department of Computer Science

University College London

December 8, 2020

I, Andreas Gutmann, confirm that the work presented in this thesis is my own. Where information has been derived from other sources, I confirm that this has been indicated in the work.

Abstract

The ongoing digitisation of previously analogue systems through the Fourth Industrial Revolution transforms modern societies. Almost every citizen and businesses operating in most parts of the economy are increasingly dependent on the ability of computer systems to accurately execute people's command. This requires efficient data processing capabilities and effective data input methods that can accurately capture and process instructions given by a user. This thesis is concerned with the analysis of state-of-the-art technologies for reliable data input through three case studies.

In the first case study, we analyse the UI of Windows 10 and macOS 10.14 for their ability to capture accurate input from users intending to erase data. We find several shortcomings in how both OS support users in identifying and selecting operations that match their intentions and propose several improvements.

The second study investigates the use of transaction authentication technology in online banking to preserve the integrity of transaction data in the presence of financial malware. We find a complex interplay of personal and sociotechnical factors that affect whether people successfully secure their transactions, derive representative personas, and propose a novel transaction authentication mechanism that ameliorates some of these factors.

In the third study, we analyse the Security Code AutoFill feature in iOS and macOS and its interactions with security processes of remote servers that require users to handle security codes delivered via SMS. We find novel security risks arising from this feature's design and propose amendments, some of which were implemented by Apple.

From these case studies, we derive general insights on latent failure as causes for human error that extend the Swiss Cheese model of human error to non-work environments. These findings consequently extend the Human Factors Analysis and Classification System and can be applied to human error incident investigations.

Impact Statement

Our findings are helpful to policy practitioners, legislators, and regulators who assess and oversee the Fourth Industrial Revolution. We gain insights about key challenges for economically advantageous transformations that aim to reuse tried and tested state-of-the-art technologies. Our proposed improvements to several such state-of-the-art technologies address the immediate security and privacy risks we identified in these technologies and make them more suitable for capturing and securing reliable data input to cyber-physical systems.

This thesis also benefits organisations and individuals who intend to make data on their digital storage inaccessible before decommissioning their devices, e.g. to comply with privacy and security legislation or for personal reasons. We identify likely causes for a long history of apparently accidental data breaches from decommissioned storage devices in the user interfaces of Windows 10 and macOS 10.14, potentially affecting hundreds of millions of users. We propose improvements for both operating systems and recommend the European Data Protection Board to issue official guidance on GDPR-conform data erasure.

Our findings are furthermore to the benefit of organisations and individuals who rely on the security of processes that involve security codes received via SMS on iPhones, as well as agencies and organisations that oversee or operate mechanisms for dispute resolution of those services that incorporated such processes. We identify security vulnerabilities in how security codes are processed by iOS, which can be exploited in attacks against organisations' security processes for user authentication, device identification, and transaction authentication – potentially placing hundreds of millions of users at risk. Apple implemented changes akin our proposed improvements, acknowledged our contributions in their security advisory for iOS 14 and joined an initiative with Google to publish a corresponding specification for the formatting and processing of security codes in SMS at the World Wide Web Consortium. We subsequently submitted further security vulnerability with this feature in iOS 14.

Our insights help developers and researchers of transaction authentication technologies. Improved understanding of users' mental models helps practitioners in designing better online banking platforms. The personas we derive help in the development and evaluation of new transaction authentication mechanisms. We propose a novel transaction authentication mechanism for online banking. At the time of submission, our industry collaborator, a leading provider of such technologies, explores opportunities to patent our mechanism and integrate it into their products to improve the security of hundreds of millions of users. We also provide a first step towards the empirical assessment of the term *gross negligence*, which could help the UK's Financial Conduct Authority to respond to a request by the UK Parliament's Treasury Select Committee for a list of dos and don'ts for *gross negligence* in online banking.

Lastly, this thesis extends our understanding of the contributions of human error to accidents and incidents, which is relevant to those overseeing or conducting such investigations. We extend the Swiss Cheese model of human error and the HFACS for human error accident and incident investigations to non-work environments and propose new methods for data collection at early stages of such investigations.

Acknowledgements

I want to full-heartedly thank my supervisor, Prof. Steven J. Murdoch, for his balanced approach to guidance and independence during my PhD, neither making me feel drowning nor clipping my wings. I learned so much more from Steven than it would be reasonable for me to recapitulate here!

I also want to thank the ‘Privacy & Usability’ Innovative Training Network for providing me with superb training and funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No 675730, and all affiliates of this project for the many joyful interactions and the opportunity to keep growing together. I want to thank my colleagues at University College London and the OneSpan Innovation Centre for their contributions to a thriving environment of intellectual growth and constructive knowledge exchange.

Lastly, I want to thank my previous supervisors who taught me a lot about computer security during my graduate studies. Prof. Jörn Müller-Quade (Karlsruhe Institute of Technology) first nurtured my interest in computer security when he taught me basic concepts and principles about the theory and application of cryptography. Under the guidance of Prof. Kanta Matsuura (University of Tokyo), I deepened my interest in making computer security applications usable by non-expert users.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

The Fourth Industrial Revolution is taking shape in the form of rapid, large-scale digitisation of previously analogue systems and processes. This affects most economic activities “from farming to finance, from culture to construction, from fighting climate change to combatting terrorism”, as described by European Commission President Ursula von der Leyen [7]. This process is sometimes referred to as Industry 4.0, a declared political goal of the European Union to increase economic output and quality of life for people living in Europe.

Previously, the First Industrial Revolution utilised the pressure exerted when vaporising water to propel machines and mechanise production. This changed the economy from solely relying on manual labour to a focus on the maintenance and supervision of machines. The Second Industrial Revolution came with the discovery of electricity and its use in industrial processes to increase the scale of production. The Third Industrial Revolution materialised with advancements in electronics and information technology. In particular the development of integrated circuits as cheap, reliable, and versatile building blocks enabled the full automation of many industrial processes.

The Fourth Industrial Revolution, facilitated by the fusion of physical objects with digital technologies, emerged in Europe. These cyber-physical systems, manifested in technologies from smartphones to the Internet of Things, blur the lines between physical, digital, and biological spheres, and affect almost every industry in every country by digitising production, management, and governance of goods and services [8]. As these changes affect most areas of the economy, they transform them from involving significant numbers of manual workers in the production of goods and services to knowledge workers who operate machines but otherwise produce the most valuable goods and services with their mind [9].

An analytical study by the EU's Directorate-General for Internal Policies [10] discussed operability as a significant challenge for this digitisation of previously analogue processes. The report stressed the relevance of those challenges resulting from the interoperability of distributed systems and processes, and the importance to secure the integrity of these processes through reliable data input and processing. Accordingly, well-functioning operating systems are considered essential preconditions to achieve reliable data input in an environment of highly heterogeneous components [11] and have, thus, become a key research stream for the end-to-end digitisation of analogue processes [12].

The dangers of unreliable data input mechanisms are evident when considering the large number of digital systems already around us, in particular those that are critical to us as individuals or as a society. Some systems, if provided with the wrong data input, could cause several bodily harm, economic damage, social disruption, and more. For example, in January 2020, a trained operator at the Provincial Emergency Operations Centre of Ontario, Canada, erroneously issued a nuclear emergency alert via radio, television, and on mobile handsets through Canada's National Public Alerting System [13]. The operator had only intended to test the system at the beginning of his work shift, as scheduled for all operators.

Beyond this example, reliable data input is not only relevant for trained workers who operate devices as part of their employment. While digital goods are mostly consumed, digital services, on the other hand, are usually interacted with by lay-people who's input is fed back into the system due to the interactivity of many service offerings. Moreover, many digital products blur the line between goods and services, such as digital services integrated within digital goods and software-as-a-service products. These conditions constitute additional challenges for the digitisation of processes since reliable input for both experts and lay-people can be an essential requirement throughout the production and life-cycle of digital services. Thus, reliable data input needs to be achieved with trained and untrained operators in mind.

The ability of cyber-physical systems to accurately perceive the intended input from their users, and execute their command under most conditions and in almost every environment, is inherently linked to a well-designed user interface (UI). This is deemed particularly important for services built upon cyber-physical systems [14]. One particular example of an industry that, as a result of early and high consumer expectations, has already seen a significant transformation and digitisation are financial services and banking [15]. Analogue processes that have existed for hundreds of years [16] have been digitised in a transformation that started more than twenty years ago [15]. Today, digital banking has been widely adopted, with some European countries reporting adoption rates by more than 90% of the population [17]. As a consequence of the rapid digitisation and corresponding technology-driven innovation seen in the financial services industry, the European Parliament [18] and European Commission [19] emphasised the significant importance of a security-by-design approach as being essential to ensure the resilience and integrity of digitised and distributed systems and processes.

These challenges are likely to be shared by most industries facing a digital transformation. Solutions developed and deployed successfully by those who addressed these challenges first are likely to become reference points for those who follow. Consequently, such technologies are set to define the commonly considered state-of-the-art for security-by-design approaches to reliable data input in the long term. For example, the Advanced Message Queuing Protocol was developed by the banking industry [20] in 2003 and is now considered an established messaging protocol for middleware implementations in IoT systems with particular focus on reliability, security, and interoperability [21].

In this context, it is important to assess some of the current state-of-the-art implementations of systems for reliable data input. To that end, we identified three requirements for securing the integrity of input data to distributed systems and processes, and present case study analyses of respective state-of-the-art implementations by major economic entities. Chapters 3, 4 and 6 of this work are each dedicated to one of the identified requirements and present a case study analysis of corresponding implementations. The subsequent chapters of this thesis are as follows:

Chapter 2 In this chapter, we review and discuss the most relevant threats to the reliability of data input, corresponding threat models, and relevant security technologies to operate in those seetings.

Chapter 3 In this chapter, we are concerned with systems' ability to capture accurate data of their users' behaviour and intend to achieve reliable data input. We present a case study in which we analyse the UI implementations of delete and erase functions in Windows 10 and macOS 10.14.

Chapter 4 In this chapter, we are concerned with systems' ability to preserve the integrity of input data *during the communication* between distributed systems and in the presence of malicious entities. We present a case study in which we analyse the implementation of transaction authentication in online banking by several European banks.

Chapter 5 In this chapter, we expand the work done in the previous chapter to propose solutions for security risks we had identified. We present two analytical studies to collate and analyse previously gathered data, and to propose a novel security mechanism to address specific security issues which beset existing mechanisms used in online banking.

Chapter 6 In this chapter, we are concerned with systems' ability to preserve the integrity of input data *during the distributed processing* and in the presence of malicious entities. We present a case study in which we analyse the interplay of iOS 12 and a remote authentication server to authorise users and transactions with security codes delivered via SMS.

Chapter 7 In this chapter, we place our findings from chapters 3, 4 and 6 in the context of, and extend, established models and theories about human error.

Chapter 8 In this chapter, we summarise the contributions made throughout this thesis and present possible future research directions on the topics we traversed.

1.1 Key Contributions

The key contributions of the work presented in this thesis are as follows:

- We find risks for reliable data input from independent components that are integrated into a technology solution if they may evolve independently and over time may become unsuitable for the role they were supposed to fulfil.

- We find risks for reliable data input from solutions that did not evolve in line with their environments.

- We identify likely causes of accidental data breaches in Windows 10 and macOS 10.14 and make 7 recommendations for OS vendors, policy practitioners, and data protection authorities.

- We identify several previously unknown factors in online banking users' mental models that can contribute when they fail to secure their transactions. We make 5 recommendations for banks to change specifics in their online banking implementations and their communication with their customers, as well as for the relevant regulators and legislators.

- We present three representative personas of transaction authentication users based on two distinct dimensions: People's "trust in security of online banking by design and default" and their "knowledge about technical aspects of online banking security".

- We propose a novel transaction authentication mechanism to address some of the security issues we identified in current online banking mechanisms. We present a security analysis and usability evaluation of this mechanism.

- We identify novel security risks in how iOS and macOS interact with security codes delivered via SMS. Our findings were acknowledged by the vendor who implemented changes akin to some of our recommendations. We present further security issues we identified and submitted to the vendor's product security team alongside our recommended changes.

- We extend the Swiss Cheese model of Human Error and the associated Human Factors Analysis and Classification System to non-workplace environments and recommend new methodologies for data collection activities during human error incident investigations.

1.2 Published Work

This dissertation has resulted in the following peer reviewed publications:

- Andreas Gutmann and Mark Warner. Fight to Be Forgotten: Exploring the Efficacy of Data Erasure in Popular Operating Systems. In *Privacy Technologies and Policy*, pages 45–58, Cham, 2019. Springer International Publishing
  This research will be presented in chapter 3.

- Andreas Gutmann and Steven J Murdoch. Taken Out of Context: Security Risks with Security Code AutoFill in iOS & macOS. In *Who Are You?! Adventures in Authentication Workshop (WAY 2019)*. USENIX, 2019
  This research will be presented in chapter 6.

1.3 Work Done in Collaboration

Large parts of this work have been conducted in collaboration with other researchers. In chapter 3, Mark Warner provided background knowledge about technical implementations of delete and erase functions in section 3.2.2, sighted and verified the documentation created throughout the study, as described in section 3.3.2, and contributed the discussion about metaphors in section 3.5.4. In chapter 4, Mark Warner contributed as the second coder of qualitative data, as described in section 4.4.2. Steven J. Murdoch sighted and verified the documentation created throughout the study presented in chapter 6, as described in section 6.4, and generally advised on all chapters in this thesis.

Chapter 2

Literature reviewed

This chapter provides a high-level overview of the most relevant threats to the reliability of data input and relevant security technologies used to operate in those settings. While we mostly focus on securing sensitive user input, we note that the topics we discuss here can be directly extended to use cases of non-sensitive user input.

We follow guidelines published by the European Union Agency for Cybersecurity on the classification of state-of-the-art technologies for Information Security applications [24]. Thereby, the state-of-the-art is distinguished from the more innovative scientific knowledge and research as well as from the more established and generally accepted rules of technologies. Existing scientific knowledge and research comprises of technologies that are highly dynamic and have not yet reached market maturity. State-of-the-art technologies likely origin from scientific knowledge and research but are less dynamic due to a necessary degree of standardisation after being launched on the market. Generally accepted rules of technologies are highly standardised in their application with few to no innovation happening and are often described in corresponding standards. We emphasise that this definition deviates from what is commonly understood as state-of-the-art for patent law and described in Article 54 of the European Patent Convention as “everything made available to the public by means of a written or oral description, by use, or in any other way” [25].

In the remainder of this chapter, we first present threats to reliable data input in sections 2.1 and 2.2. In section 2.3 we are concerned with technologies to mitigate such risks locally, most of which are scientific knowledge, and discuss limitations and shortcomings of these approaches in section 2.3.4. Finally, we describe state-of-the-art technologies to mitigate such risks through remote attestation in section 2.4.

2.1 Human Error as Threat to Reliable Data Input

Reliable data input depends primarily on the systems' ability to accurately capture users' intended input. This requires suitable data input mechanisms which can be located, identified, and operated by users of those systems. We expect from well-designed system to yield a low human error incidence rate; at least in the absence of malicious interference. Human error can be distinguished between mistakes and slips [26], referring to the execution of a wrong plan or the correct plan but with flawed execution, respectively. Mistakes can be caused if users misunderstand or -predict the consequences of their action, e.g. when pressing a button. Slips can be caused if systems unduly penalise inadvertent behaviour, e.g. manually terminating software with unsaved changes. Sometimes human error can be both a mistake and a slip, but deliberate violation of a correct sequence of actions is not considered human error [27].

If users of a system commonly do not operate it correctly, inadequate system design is likely the root cause [27]. Erroneous actions are context-dependent and as such inherently linked to the environment in which they are executed [28]. It is possible to predict erroneous actions through the systematic study of systems and processes, and consequently mitigate the causes for and effects of human error through more appropriate system design [29]. Designs that minimise the incidence of human error, and mitigate the consequences thereof, are called error-tolerant (or sometimes human-error-tolerant) design [30] in analogy to fault-tolerant design. For example, software commonly prompts users with the option to save the programs current state before being terminated. Assessing the number and severity of human errors, and the ease with which users can recover therefrom, are key components of usability assessments [31]. As such, the human error incident rate of a system is best reduce by studying its usability and consequently improving its design [32, 26].

2.2 Adversarial Threats to Reliable Data Input

Adversarial threats to a system are enacted by malicious entities tampering with the system's benign workflow, resulting in an unwanted system state. The main security risks for the correct performance of a system, with respect to the reliability of sensitive data input and as direct consequences of adversarial threats that successfully manifested themselves, are related to malicious content manipulation, i.e. tampering with previously registered or concurrent data input, creation of fake data input, or prompting the user to some security-diminishing data input. In this section, we describe a set of adversarial threats to reliable data input that are of relevance for the subsequent chapters in this thesis through their corresponding threat actors and models.

2.2.1 Threat actors

A threat actor in an entity that is partially or wholly responsible for an adverse incident that has the potential to violate a security policy or otherwise cause an unwanted system state. Threat actors can be categorised as external, internal or partner. External threat actors generally do not benefit from preexisting trust or privileged access, while internal or partner threat actors may benefit from some level of trust or privileged access. The threat actor may be an individual or a group of entities. An adverse incident caused by a threat actor could be intentional or accidental and its purpose could be malicious or benign.

There is no exhaustive and agreed-upon list of all possible threat actors for all possible systems. Instead, the most commonly referenced threat actors could be considered stereotypical descriptions of entities behind some of the most commonly encountered threats. While it can be beneficial for security analyses to consider different threat actors, they are usually categorised and described with a focus on their goals and motivation (and possibly their social status) but only loosely linked to their technical capabilities. Since it would be impossible to name all possible threat actors that could be of relevance for the systems considered throughout this thesis, we instead look at a number of prominent threat actors that are not the focus of our investigations. Yet, that does entail whether the investigated technologies offer protection against such threat actors, as this would depend on the relevant technical capabilities of each specific entity.

In the light of the aforementioned, we do not particularly consider nation-state or country-sponsored threat actors [33], insider threat actors [34], physical threat actors [35], and tech abusers [36]. To be more precise, and without loss of generality, we consider threat actors out of scope that could manipulate the entire digital process (e.g. full control over the remote service with which the user is interacting), coerce the user to provide input other than their unaltered intentions (e.g. through physical intimidation or legal charges), or exploit access policies (e.g. a company's employee acting in violation of the security policy or a tech abuser with a registered account for the user's devices).

2.2.2 Relevant technical threats

Threat models are descriptions of the methods threat actors could use during attacks and some of the goals they might pursue. They are used in system design and engineering to model those adversaries and describe the threats that were considered by the designers and engineers. In this section, we describe those threats we consider most relevant when systems require reliable data input.

2.2.2.1 Untrusted Code Execution

The single largest adversarial source of security risks for the reliability of data input stems from the execution of untrusted code that could lead to unwarranted activity on the system. It would be difficult, if not impossible, to detect and prevent such behaviour of untrusted code through static analysis before it is being executed [37]. Despite these risks, untrusted code is routinely and automatically executed on most personal, multi-purpose IT devices, e.g. as many applications need to dynamically load content at run-time for software updates, to browse the Internet, and for device synchronisation. Security risks from the execution of malicious, erroneous, or otherwise unwanted code prevail even in the absence of technical exploits and code obfuscation [38], as the user might (unwittingly) install such software themselves [39]. The scale of these risks becomes apparent when considering the existence of an entire industry specialised on enticing users into the installation of Potentially Unwanted Programs [40, 41].

2.2.2.2 Phishing

The term phishing describes fraudulent impersonation of another entity via direct communication, e.g. email or text message, to lure victims into insecure behaviour. Most commonly, the consequence of a successful phishing attack is the download of Potentially Unwanted Programs (e.g. malware), leakage of sensitive information with the subsequent impersonation of the victim in front of other entities (e.g. presenting the victim's username and password for authentication), or deceiving the user into otherwise unwarranted behaviour (e.g. instructing the victim to conduct an unauthorised transfer or modify the relevant bank details for future invoices).

Phishing attacks can be based on technical vulnerabilities, such as spoofing [42], or utilise social engineering techniques, such as exploitation of scarcity [43] or obedience to authority and compliance with social norms [44]. Most frequently, phishing attacks deploy a combination of both measures to present their target with a convincing story and authentic experience, similar to an interaction with the impersonated person or service [45].

2.2.2.3 Man-in-the-Middle (MitM)

A Man-in-the-Middle attack is an impersonation attack aimed at secretly intercepting–and possibly manipulating–the communication between two or more entities on a network, thus impersonating one or more entities with respect to the other entities. Thereto, the attacker usually forwards any authentication traffic while manipulating cryptographic key negotiation material. Once established, messages sent by a legitimate communication partner to a victim are received by the attacker instead, who can forward, drop, or insert messages to either party.

MitM attacks are almost exclusively of technical nature without exploitation of the respective user's behaviour. Common attack vectors are unencrypted connections such as public WiFi, cache poisoning of the Address Resolution Protocol [46] or Domain Name System [47], and forged Secure Sockets Layer (SSL) / Transport Layer Security (TLS) certificates [48]. Notably, only the latter could be detected by a user based on certificate warnings. Traditional authentication methods based on shared secrets, such as passwords and biometrics, are vulnerable to this attack once the communication channel has been compromised.

2.2.2.4 Man-in-the-Browser (MitB)

As a special case of untrusted code execution with particular significance, mentions of Man-in-the-Browser attacks date back to the year 2005, when Bruce Schneier [49] first wrote about shortcomings of traditional Two Factor Authentication (2FA) against Trojan horse malware attacks on online banking platforms. The following year, Gühring [50] described findings of in-the-wild Trojan horse malware campaigns that successfully mislead and defrauded users, and coined the term Man-in-the-Browser attack as a Man-in-the-Middle attack on the visual communication between the user and their browser. Such malware has the ability to *"modify the transactions on-the-fly, as they are formed in browsers, and still display the user's intended transaction, [making them impossible to] be detected by the user at all, as they are using real services, the user is correctly logged-in as normal, and there is no difference to be seen"* [50]. These attacks are completely opaque to the user and the host web application.

Security mechanisms such as 2FA and public key infrastructure (PKI), which deal with user authentication and access control, are ineffective against this attack because it takes place on the transaction level rather than the authentication level. A large number of distinct banking malware families capable of MitB attacks are known [51], making financial malware one of the most prevalent threats faced by European consumers today [52].

2.3 Securing Data Input Locally

As described in section 2.2, the execution of untrusted code and its interference with benign processes is the single largest adversarial threat to the reliability of data input (and processing). Logical isolation through a process called sandboxing is essential to mitigate most risks from untrusted code execution [53]. This section is concerned with approaches based on process isolation to address risks to reliable data input and will conclude in a discussion of why the proposed methods are insufficient to provide a high level of assurance.

The immediate benefit of process isolation is that domains with higher privileges can monitor the behaviour of those considered less trustworthy, which has been used in various domains, e.g. system fortification [54, 55], intrusion detection [56], malware analysis [57], and web browsing [58]. The breakdown of such isolation, e.g. through unwarranted access to arbitrary kernel-memory locations [59], is a major security vulnerability. The ultimate goal is to reduce the Trusted Computing Base of relevant systems and most implementations are based on Trusted Execution Environments, both of which will be discussed next.

2.3.1 Trusted Computing Base (TCB)

A Trusted Computing Base is comprised of the totality of hardware, software, analogue processes, and individual actors whose correct operation and decision-making are essential for the security of a system. It can include components of an Operating System such as computer files and processes, and parts of an organisation such as individual employees and management processes. A system security perspective–with focus on the CIA-Triad of Confidentiality, Integrity, and Availability–encourages a reduction of the system's attack surface by minimising the size of the TCB while including sufficient redundancy to avoid any single-point-of-failure. A small TCB commonly excludes (parts of) the Operating System and can be combined with tamper-resistant hardware to further reduce the attack surface of physically exposed systems.

Sensitive tasks should only be processed within the TCB. Since the input to sensitive tasks may come from outside the TCB, and in particular, for Graphical

User Interfaces as the core facilitator of user input and output in modern computing systems, additional control measures are mandated. Especially for sensitive user input, the Orange Book (also known by its title *Trusted Computer System Evaluation Criteria*) mandates a trusted communication path between the user and the TCB, which “shall be activated exclusively by a user of the TCB and shall be logically isolated and unmistakably distinguishable from other paths” [60].

2.3.2 Trusted Execution Environment (TEE)

A Trusted Execution Environment is a hardware-enabled sandboxing approach to create a separate execution environment with a small TCB, dedicated to the secure processing of sensitive tasks in isolation from the main execution environment of a system. The high-level goals of a TEE are to isolate the processing of sensitive tasks from each other and the rest of the system, to provide secure storage for data at rest (e.g. cryptographic keys), to digitally sign messages processed in the TEE for remote attestation of message origin (including the state of the TEE), to offer secure provisioning for secrecy and integrity during data migration, and to ensure a trusted communication path between modules hosted in a TEE and the system’s peripherals [61]. Yet, even the comparably small TCB of a TEE cannot guarantee a secure and bug-free system [62].

A common use case for such sandboxing is banking applications on mobile phones. This provides an environment for the secured processing of sensitive tasks such as displaying a summary of transaction context data and capturing sensitive user input. As the TCB for mobile phones commonly includes part of the operating system (OS), additional security measures such as run-time application self-protection, application cloning countermeasures, and that the devices have not been rooted are usually warranted.

2.3.3 Methods to Secure Reliable Data Input Locally

The trusted communication path of a TEE does not necessarily fulfil the stringent requirements set out in the Orange Book for trusted computer systems, i.e. a reliable method for the user to identify which process they are interacting with [63]. The former provides means for (software hosted in) the TEE to validate that an input signal origins from (or output signal is delivered to) the correct peripheral device, e.g. keyboard, mouse, or camera. The latter requires the ability for the users of a system to validate whether they are interacting with a specific module hosted in the TEE. Reliable data input requires both capabilities. In the example of banking applications on mobile phones given in section 2.3.2, the TEE provides no means for users to reliably distinguish real banking applications from well-made imitations. Several approaches to improve interfaces and secure sensitive user input in accordance with or close to the requirements of the Orange Book have been proposed and will be described in the remainder of this section.

Securing individual applications. The methods presented here focus on the interaction with one privileged application at the cost of parallel interactivity with multiple applications on the same screen, i.e. a user would be unable to interact with two applications on the same screen if one of them requires reliable data input. While these conditions might be undesirable for many use cases, they excel on mobile devices where concurrent interaction with multiple applications on the same screen is usually not a supported feature anyway.

A method for reliable data input through a combination of TEE with a hypervisor was proposed by Zhou *et al.* [64]. The hypervisor itself has OS-like responsibilities to manage all trusted software and their associated resources, including memory management, process scheduling, device drivers, and verified boot loader. Users are expected to verify the integrity of a trusted software module in the TEE using a separate device and public key-based challenge-response authentication.

A method to establish reliable data input for mobile devices without the need for separate hardware was proposed by Li *et al.* [1]. This method requires two frontal light-emitting diodes (LED) and a hardware random number generator, both exclusively accessibly by the TEE. The system then periodically randomises the foreground and background colour of applications hosted in the secure environment and synchronises the LED values, as depicted in fig. 2.1, which enables user-side verification on the same device. Reliable (and secret) data input is then achieved by additionally randomising the location of keyboard keys, as depicted in fig. 2.2, and buttons on the screen after each atomic user input, e.g. selecting a key.

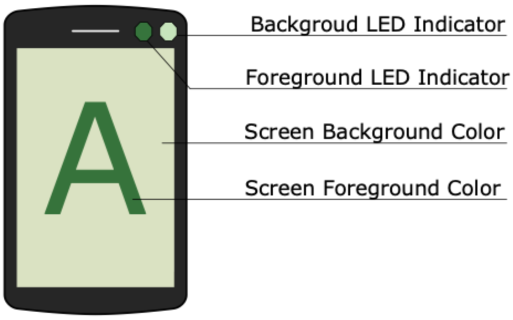


Figure 2.1: Depiction of two LED showing foreground and background colour to distinguish between trusted and untrusted applications, reproduced from [1, Fig. 3].

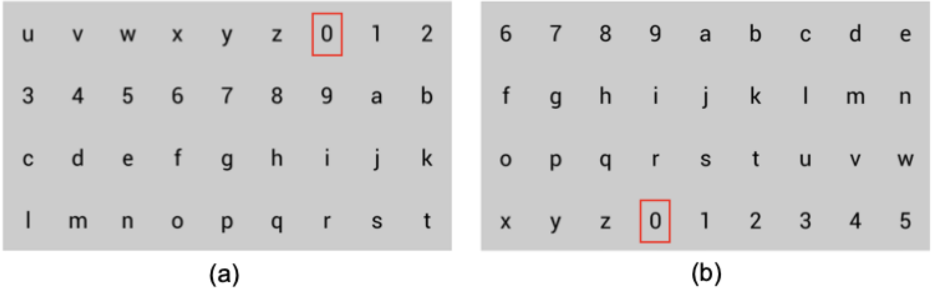
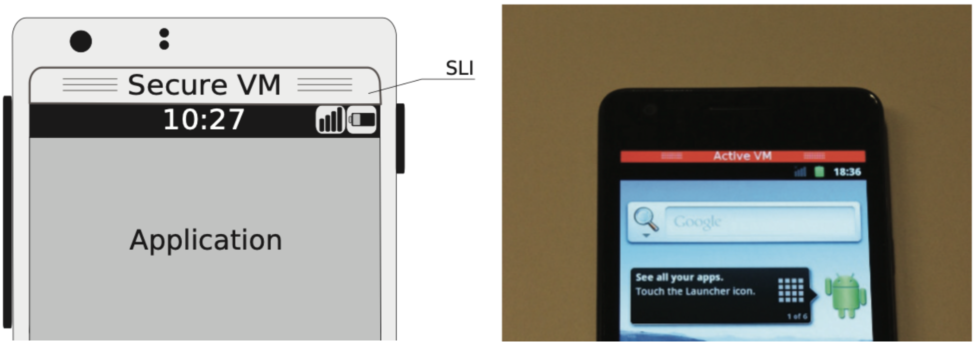


Figure 2.2: Two randomised keyboard layouts, reproduced from [1, Fig. 4].

Reliable data input in the context of a corporate Bring Your Own Device (BYOD) use case was explored by Lange and Liebergeld [2]. The proposed mechanism to separate corporate and private environments uses a dedicated area on the top of a mobile phone's screen to provide a label isolated from the remainder of the screen, as depicted in fig. 2.3a. The name and colour of that label are set during provisioning of the device and help users to distinguish between the respective environments, as depicted in fig. 2.3b. The authors noted limitations in the user's ability to recognise and distinguish different colour under different ambient light conditions. Danisevskis *et al.* [3] proposed a similar mechanism to include geometric patterns which could perform better under varying ambient light conditions, as depicted in fig. 2.4.



(a) SLI denotes the Security Level Indicator, a privileged area at the top of the screen. Reproduction from [2, Fig. 4].

(b) The Security Level Indicator is red and displays the text label of the active environment. Reproduction from [2, Fig. 7].

Figure 2.3: The mechanism proposed by Lange and Liebergeld [2] to distinguish between corporate and private environments.

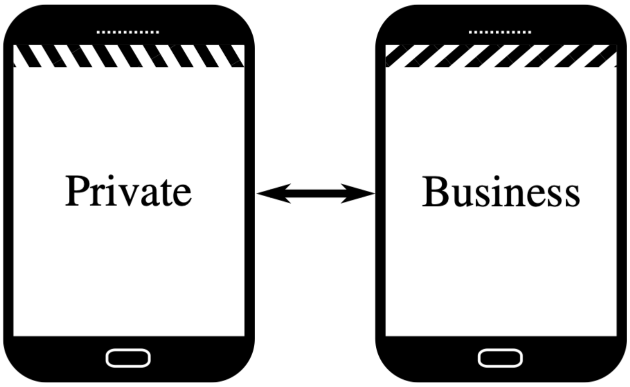


Figure 2.4: Two geometrical patterns to distinguish between corporate and private environments, reproduced from [3, Fig. 1].

Securing multi-window screens. Methods for reliable data input of multi-window screens support the interaction with privileged and unprivileged applications concurrently.

The only method with large scale adoption, and presumably simplest method, is presented in isolation due to its limited scalability. In this method, during any possible system state, specific input signals from the user are directly linked to the activation of specific functions. It requires a dedicated physical button or other unique input combination for each function and thus does not scale well with the number of supported functions. The TCB includes any software capable of overwriting those settings, including large parts of the OS. For example, a Home-button is commonly used on mobile phones to access the Home screen, while pressing the Control-Alt-Delete key combination in many Windows OS invokes the GINA[1] screen.

Weiser and Werner [65] proposed a method to establish reliable data input based on a pre-shared secret between the user and the module hosted in the TEE while relying on trusted drivers and display server. The user can verify the authenticity of a module if it displays this shared secret on the screen, while the TEE, trusted drivers, and trusted display server ensure that this secret does not leak to unauthorised applications.

Tygar and Whitten [66] proposed that users could detect malicious software with user-level privileges impersonating higher privileged dialogues if OS would feature a system-wide window personalisation through customisable (background) images. Shapiro *et al.* [67] suggested a trusted display server to dim the maximum brightness of background windows, add a bright-coloured border to the currently active window, and label the currently active window in a trusted area at the bottom of the screen. Feske and Helmuth [4] proposed similar methods with the additions of a trusted boot loader generating trusted labels to display on the borders of each window and colour-coded borders on all windows to help identify whether they are foreground or background, as depicted in fig. 2.5.

Securing remote content. The last set of methods proposed in the literature, and discussed in the following, focuses on the challenge of reliably distinguishing dynamically loaded remote content from local processes, isolating sensitive remote content from untrusted local processes, and enabling reliable data input for sensitive remote processes.

Ye *et al.* [68] were concerned with exposing when untrusted code loaded from a web server would mimic the signals of a trusted browser. They suggested adding coloured borders with a randomised geometric pattern to browser windows. Two colours would be used to distinguish between browser-only content (e.g. settings and certificate information) and windows featuring web server content (e.g. the

[1] The Graphical Identification and Authentication screen implements the systems authentication policy and is intended for all identification and authentication user interactions.

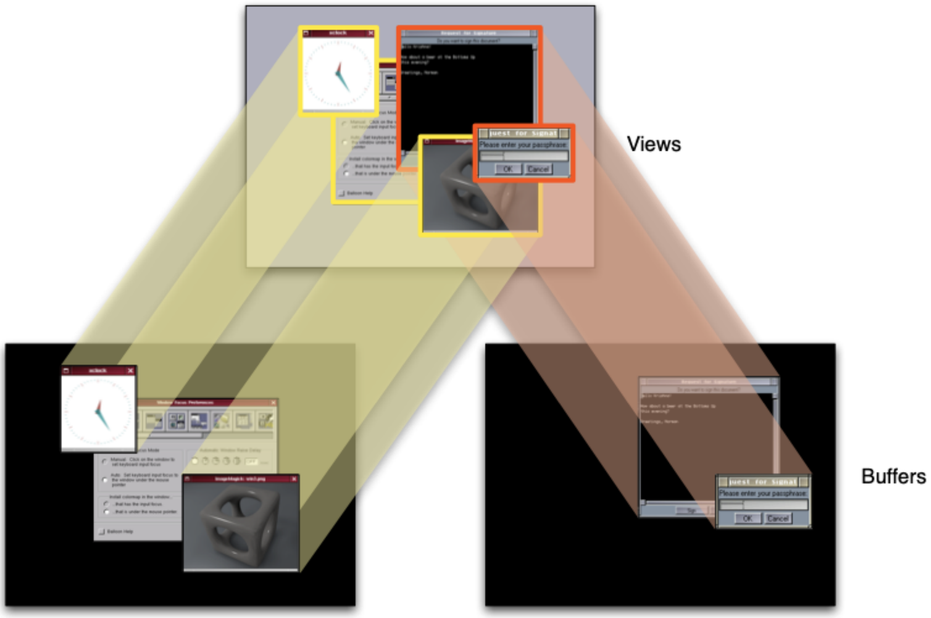


Figure 2.5: Two applications with multiple windows each are rendered separately and augmented with identifying information before being joined on the device's screen, reproduced from [4, Fig. 3].

main browser window). A trusted area on the screen features the currently trustworthy geometric pattern.

Eskandarian *et al.* [5] focused on the problem to differentiate between trusted and untrusted sections of code within the same application, referencing the use case of MitB attacks (see section 2.2.2.4) against online banking platforms. Their system requires a TEE and multiple trusted hardware devices between the main system and each relevant peripheral devices, e.g. keyboard and display. These trusted hardware devices support symmetric encryption (including local memory for non-repeating counters, a cryptographic key pre-exchanged with the TEE, etc.), and feature visual indicators (e.g. LEDs) to help users distinguish between encrypted and unencrypted communication modes (with the TEE and the OS, respectively). Remote attestation between the TEE and the web server is used to establish the integrity of the webpage while loading. Hypertext Markup Language (HTML) tags are used to identify sensitive data, e.g. password input fields, which are further processed in the TEE. Document Object Model (DOM) subtrees are sent by the TEE over the encrypted channel to the display as an overlay on top of the unencrypted data, i.e. the remainder of the webpage. A dedicated area on the bottom of the browser, equally under the control of the TEE, is used to displays salient context data, e.g. domain of the website and name-tag of any currently active form fields, as depicted in fig. 2.6. The visual indicators on the trusted hardware devices indicate whether the user is currently interacting with the TEE or an untrusted application.

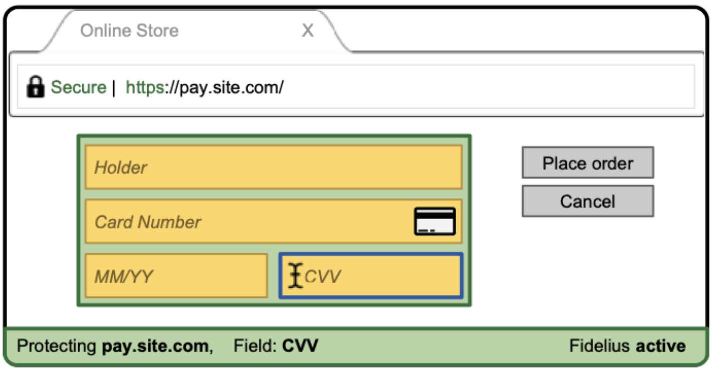


Figure 2.6: Depiction of a trusted overlay to identify trustworthy content hosted in the TEE, reproduced from [5, Fig. 2].

2.3.4 Review of Proposed Methods

Most of the proposed methods remain untested in large scale implementations and few have been subject to comprehensive user studies, raising questions over whether the proposed measures could have the anticipated effects and would establish reliable and trusted communication paths between software and users.

Many of the mechanisms used to distinguish between trusted and untrusted software rely on visual indicators such as LED status lights, background images, coloured window borders, or shared secrets to help users distinguish between them. The consequence of software-based security indicators is an extensively large TCB (including Operating System components that could read or write such settings) and the effectiveness of hardware-based security indicators such as LED status lights has been questioned before [69]. Furthermore, research has shown that the sole presence (or absence) of security indicators or warning messages does not necessarily lead to more secure behaviour [70], and even tech-savvy users frequently misunderstand the meaning of common security indicators [71]. We argue that the proposed trusted screen areas also share conceptual similarities with the security indicators evaluated in [70, 71] and conclude that their effectiveness should be considered equally questionable.

Other mechanisms rely on the integrity of trusted hardware devices either to secure the communication channel between TEE and peripheral devices or to verify the integrity of an established communication channel. Both add considerable cost, effort, and complexity to the installation and operation of a system while still being subject to concerns about the questionable effectiveness of security indicators discussed above.

Lastly, we believe that there is a reason for additional scepticism about the effectiveness of most of the proposed methods due to the absence of empirical evaluations of specific implementations. For example, it is unknown whether users would be able to correctly interpret added context information or whether long-term use and habituation would diminish the user's attention to the salient context information. Even for well-established methods such as the Windows GINA screen, which appears after pressing the Control-Alt-Delete key combination, it has previously been questioned whether users understand the security implications [72] and subsequent research has indeed found users disclosing their Windows credentials to websites imitating the appearance of a software installation window [73].

In summary, the methods previously proposed in this section might alleviate some of the risks associated with the threats described in section 2.2 but face severe limitations.

2.4 Securing Data Input Remotely

In the absence of reliable means to safeguard the entry of sensitive data on a device for remote processing, as discussed in section 2.3.4, alternative technologies for remote attestation of user input and intent are warranted. Such technologies operate most commonly through the transmission of security codes and are commonly deployed to attest authenticity and authorisation to safeguard the integrity of sensitive user input and assert user intentions. In most cases, security codes are transmitted over another communication channel, whereby the respective user is expected to manually quote such codes on the primary communication channel to complete the respective process. In other implementations, security codes are transmitted via a bidirectional, secondary communication channel and the respective user only needs to push a button to complete each process. In this section, we describe the most common categories of these technologies and their applications.

2.4.1 User Authentication and Device Authorisation

The purpose of user authentication and device authorisation is to validate the correctness of an entity's asserted identity. This security measure is commonly combined with access control measures, granting specific privileges only to validated identities.

One Time Password (OTP) Many systems rely on OTPs for increased security during user authentication when registering a login attempt for the corresponding account. To complete the login, the legitimate user is expected to retrieve this code from their device and quote it – something an impersonator would be unable to do without access to that device.

OTPs are valid for single use and potentially only within a short time-frame. They can be used as an element during 2FA, commonly combined with a password or biometrics. This offers increased protection against account takeover risks [74], e.g. exploitation of common / easy-to-guess or reused credentials and replay-attacks, in which an adversary observed a successful login and mimics the user's actions to gain access to their account. While OTPs cannot offer full protection against the phishing and MitM attacks described in sections 2.2.2.2 and 2.2.2.3, respectively, *per se*, the additional security layer adds complexity to the corresponding attack methods. As a consequence, 2FA with OTPs provides good security against most Account Takeover threats [75], although highly sophisticated and targeted attacks must still be considered a significant risk [76].

SMS OTP is the most common implementation, with estimations that 25% of businesses with online login platforms support sending security messages via SMS (which increases to an estimated 31% for mobile channels, e.g. services delivered via mobile phone applications) [77]. Businesses commonly relying on this method include those from areas such as financial, health care, identity management, and legal services [78].

One Time Authorisation (OTA) Another use of security codes is to support the activation of specific features in existing software installations or the registration of new installations. This process can furthermore create a permanent link between a software installation and the specifics of the respective communication channel, such as a phone number or email address, and double-function as a unique user ID for services such as instant messengers. Example applications in this category are most instant messengers and some government services, such as the IRS2Go app from the U.S. Internal Revenue Service and the HMRC app from U.K. HM Revenue & Customs.

Similar to OTPs, OTAs are commonly valid for single use and potentially only within a short time-frame. They are most commonly delivered via SMS or email and need to be quoted in the respective software to complete the installation or activation. Although their main purpose is utility rather than security, it is expected that authorisation codes are exclusively delivered to the respective, legitimate parties.

2.4.2 Transaction Authentication Numbers

Transaction Authentication Numbers (TAN) are commonly used for remote transactions to validate that data input received by a service provider match with the intentions of a legitimate user of the affected account. They can be dynamically linked to the corresponding transaction, entailing that each code could only be used once and only to authorise that specific transaction – and commonly only within a short time-frame. Transaction Authentication is an essential requirement to defend against MitB attacks [79], which was described in section 2.2.2.4.

TANs are most commonly used by the financial services industry. They can fulfil the mandatory security requirements for online banking and card-present transactions in the EU under the Payment Service Directive 2 (PSD2). They are also part of the global 3D Secure fraud prevention scheme to secure online payments, i.e. card-not-present transactions, as first proposed by Murdoch and Anderson [80].

3D Secure has been adopted by schemes such as *Verified by Visa*, *Mastercard SecureCode*, and *American Express SafeKey*.

Users can receive security codes for transaction authentication via different methods and in the following, we describe the most common categories.

Visual Cryptogram Visual Cryptogram technologies require that the user has a trusted device with a pre-exchanged secret key[2], e.g. a dedicated hardware device or an app on their mobile phone. When the service provider (e.g. the bank) receives transaction instructions for a user account, it sends back an encrypted summary of the salient *transaction context data* of that transaction augmented with a TAN. In the case of online banking, such summaries contain–at least–information about the destination account number and transaction value, and the TAN is cryptographically linked to that transaction. This cyphertext is then displayed in the browser (or app) from which the instructions were issued as Visual Cryptogram. Figure 2.7 is an example of a Visual Cryptogram.

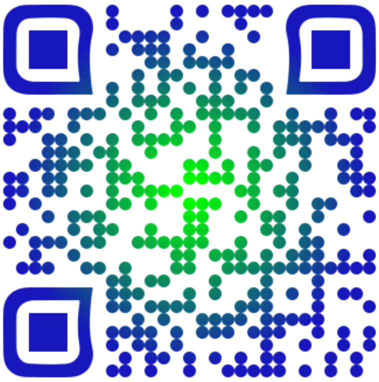


Figure 2.7: Example of a Visual Cryptogram.

Next, the legitimate user is expected to use their trusted device to scan the visual cryptogram. Only that user's trusted device can decrypt the visual cryptogram to display the transaction context data and correct TAN on its screen. Finally, the user is expected to read the transaction context data and, if it matches their intention, authorise the transaction by quoting the TAN on the device which they had

[2] In some installations, such as the device depicted in fig. 2.8, the user's banking card must be inserted to present that key.

originally used to issue the transaction instructions (i.e. browser or app). Figure 2.8 depicts an example of an authenticator for Visual Cryptograms.

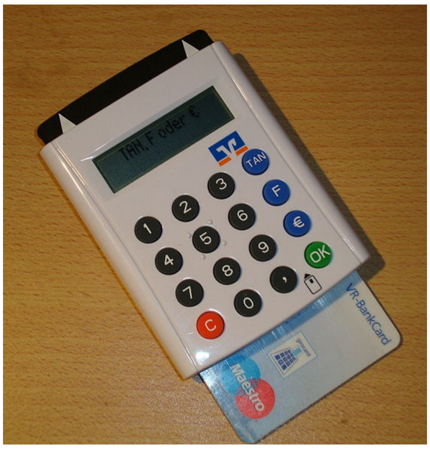


Figure 2.8: Example of a Visual Cryptogram authenticator.

Re-keying Re-keying technologies require that the user has a trusted device with a pre-exchanged secret key, e.g. a dedicated hardware device or an app on their mobile phone. In the case of Re-keying, the user is expected to type into their trusted device the salient transaction context data of any transactions sent to the service provider, e.g. bank. In the case of online banking transactions, this contains–at least–information about the destination account number. Utilising the pre-exchanged secret key, the device or app generates a TAN which is cryptographically linked to the input data. The user is then expected to quote this TAN on the device which they had originally used to issue the transaction instructions (i.e. browser or app). Finally, the bank can verify that the legitimate user's trusted device would have generated the received TAN given the received transaction instructions. Figure 2.9 depicts an example of a Re-keying authenticator.

Text messages and push notifications Text messages and push notifications require that the user has a trusted mobile phone. When the service provider (e.g. the bank) receives transaction instructions for a user account, it sends a summary of the salient transaction context data of that transaction and a TAN to the corresponding user's registered mobile phone. In the case of online banking, such summaries contain, at least, information about the destination account number, transaction value, and the TAN, whereby the latter is cryptographically linked to that transaction. The user

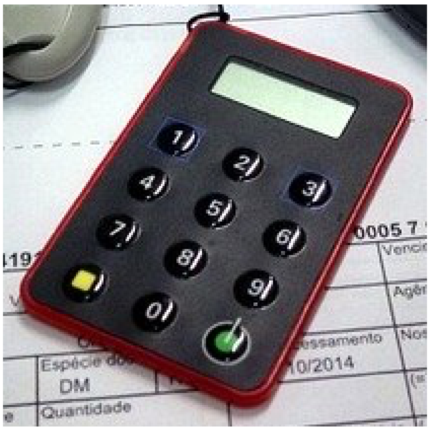


Figure 2.9: Example of a Re-keying authenticator.

is expected to read the transaction context data and, if it matches their intention, authorise the transaction by quoting the TAN on the device which they had used to issue the transaction instructions (i.e. browser or another app).

Critics of such implementations of transaction authentication have pointed out that it could allow users to receive TANs on the same device they used to issue the transaction, which might violate assumptions in the service provider's threat model. As a consequence, many banks disallow this practice in their Terms and Conditions, avoid using responsive or adaptive designs for their online banking platforms, and deploy technical measures to attempt detecting if a banking app is installed on the same device that the user registered to receive TANs.

Additional critique is voiced in particular about SMS for being unencrypted and, thus, at risk of illegitimate access, e.g. through Man-in-the-Middle attacks [81] or malicious apps [82] on the user's device. Another related and commonly referenced example refers to the Signaling System 7 (SS7), a communication channel used to exchange data between network devices in international and local telephone networks. SS7 can be exploited to intercept some or all traffic to a SIM card [83], such as SMS or phone calls. A data breach at Reddit Inc. in the summer of 2018 is a prominent example [84]. Another widely known security risk refers to an attacker obtaining a valid SIM card registered on the user's phone number, frequently called a SIM swap scam. Here, the attacker tries to con the network provider into issuing a new SIM card and either collects it from their mail or impersonates the victim and collects the new SIM in person. An alleged theft of cryptocurrency in summer 2018 is an example of this type of scam [85].

2.5 Summary

In this chapter we reviewed the relevant literature for reliable data input. We considered human error threats due to low usability of systems in section 2.1 and adversarial threats due to low security of systems in section 2.2. In both cases it would be impossible to eradicate all such threats: unpredictability and variability of behaviour is part of the human nature and any secure mechanism can be attack by breaching the security of the system which operates the mechanism. It is for this reason that any investigations into the reliability of data input need to consider both the usability and security of the system. Indeed, high levels of usability and security usually coincide as shortcomings in either area are likely to affect the other. In particular in section 2.2.2, we presented explicit attack methods that are being referenced again during chapters 4 and 6.

We have also reviewed the literature on technical means to secure data input (against the attacks described in section 2.2.2) locally on the user's device or through means of remote attestation in sections 2.3.3 and 2.4, respectively. For the former, we argued in section 2.3.4 that findings from the available literature, as well as the fact that empirical evaluations are largely absent, showed that their effectiveness is questionable. For the latter, we presented a set of technologies for remote attestation that have found widespread adoption with implementations in large-scale deployments, and remain in focus during the later chapters of this thesis.

Chapter 3

Fight to be Forgotten: Exploring the Efficacy of Data Erasure in Popular Operating Systems

A modified version of this chapter has previously been published as

> Andreas Gutmann and Mark Warner. Fight to be forgotten: Exploring the efficacy of data erasure in popular operating systems. In *Privacy Technologies and Policy*, pages 45-58, Cham, 2019. Springer International Publishing.

and presented at the *Annual Privacy Forum 2019*.

3.1 Introduction

A long history of longitudinal and intercultural research has identified decommissioned storage devices (e.g., USB memory sticks) as a serious privacy and security threat. Sensitive data deleted by previous owners have repeatedly been found on second-hand USB sticks through forensic analysis. Such data breaches are unlikely to occur when data is securely *erased*, rather than being *deleted*.

The ability to securely erase data can be a key requirement for compliance with regulations and legislation such as the General Data Protection Regulation (GDPR). The *right to erasure* (or *right to be forgotten*) in Article 17 of the GDPR is considered by some to be the most difficult obligation to comply with [86, p. 64]. It states that data subjects can, with certain exceptions, have their personal data erased by the responsible data controller. Moreover, it states that personal data should also be erased without undue delay under other circumstances. For example, where the data is no longer required for the purposes for which it was originally collected, or when the data subject withdraws consent on which the processing was based. The UK's national data protection authority, the Information Commissioner's Office (ICO), states that data which is subject to a valid erasure request must be placed "beyond use, even if it cannot be immediately overwritten" and can, in certain circumstances, pose a significant data protection risk [87].

The terms *delete* and *erase* are often used interchangeably. The Merriam Webster thesaurus lists both words as related[1], whilst the Oxford and Cambridge dictionaries list them as synonyms[2]. Yet, in computer science, these words have a different meaning, and the distinction between the two has consequences for compliance with data protection legislation.

From a technical perspective, these terms describe different concepts. Erase typically describes purposeful overwriting of data with other data – rendering it immediately irretrievable – whilst delete typically refers to data being forgotten[3] by

[1] `https://www.merriam-webster.com/thesaurus/delete` (Accessed: 30 March 2020)

[2] `https://en.oxforddictionaries.com/thesaurus/delete` (Accessed: 30 March 2020) and `https://dictionary.cambridge.org/dictionary/english/delete` (Accessed: 30 March 2020)

[3] Colloquial expression.

the OS and marked as available for overwriting. This allows new data to be stored in its place when required, but the old data often remains retrievable until it has been overwritten.

It is perhaps unsurprising that confusion exists between these two terms due to their linguistic similarity and interchangeable use in everyday conversation. Yet, problems can emerge if a data controller is unaware of the technical differences, with significant risks developing that could lead to exposure through non-compliance with data protection legislation. For example, deleting rather than erasing data from a decommissioned storage device could result in a data breach. As most delete and erase operations are executed through a computer's OS, the User Interface (UI) of these OS are well-positioned to provide users with guidance on the appropriate use of delete and erase operations to limit confusion between these terms.

In this chapter, we present a case study on the design of UI file removal functions in the state-of-the-art systems macOS 10.14 and Windows 10, and their ability to reliably capture users' intended input in the context of legal requirements for data erasure. We use accidental data breaches from decommissioned USB sticks as the context for a streamlined Cognitive Walkthrough to explore the gap between the legal data protection requirements for the erasure of data, and file removal functions in these OS. In doing so, we discover linguistic confusion within the UI of both OS, which could lead to increased uncertainty when data controllers undertake their legal obligation to erase data. Specifically, we found (1) inconsistencies in the language used around delete and erase functions, (2) insecure default options, and (3) unclear or incomprehensible information around delete and erase functions. Subsequently, We discuss how this could result in data controllers becoming non-compliant with a legal obligation for erasure, putting data subjects at risk of accidental data breaches from the decommissioning of storage devices.

As a result, our research identifies a need for guidelines and best practice on GDPR compliant erasure. We present a set of implications for practice that could be used to improve consistency between UIs and data protection legislation. Finally, our research evidences the importance of further investigations into the suitability of those tools most commonly used by non-experts when reliable data input is required, e.g. to comply with regulatory requirements.

3.2 Background

In this section we first explore previous research into people's data hygiene, taking a particular focus on the hygiene of decommissioned storage devices. We then explore some of the technical nuances of delete and erase operations using modern-day technologies.

3.2.1 Personal Data Hygiene

A large number of publications dating back to 2003 provide both longitudinal and intercultural insights into people's data hygiene. Researchers typically buy second-hand storage devices on the open market, forensically analyse them, and report their findings. The first of these studies was conducted on second-hand hard-disk drives (HDD) purchased in the US between 2000 and 2002, and published in 2003 [88]. The first longitudinal assessment happened in the UK through annual investigations from 2005 [89] until 2009 [90]. Similar studies have been conducted on second-hand USB storage devices (e.g., [91]), and mobile phones (e.g., [92]). Studies of this nature are also not limited to the UK market, with similar research being carried out in other parts of the world (e.g., Australia [93] and USA [94]). Consistent across these studies is the presence of sensitive personal data from a large number of decommissioned drives due to failures in the erasure process. Jones *et al.* [90], for example, forensically analysed USB sticks bought in the UK and recovered personal data which included: birth certificates, videos of children at a school, client data, and police staff records.

In addition, memory chips from decommissioned devices are commonly recycled into new electronics, even though some of their old content may still be available and could be recovered [95, 96]. The risk of data breaches from recycled memory chips is likely to increase due to Directive 2012/19/EU on *waste electrical and electronic equipment*. Article 4 aims at encouraging "cooperation between producers and recyclers" to integrate more recycled material in new equipment and Article 5 gives priority to achieving high recycle rates for small IT devices such as USB sticks.

Diesburg *et al.* [94] compared people's data hygiene practices with their intentions when decommissioning USB sticks, and found people regularly confusing delete and erase functions. The authors recovered data from 83.3% of USB sticks where previous owners anticipated it being "very hard" to recover.

In summary, people often fail to appropriately erase sensitive data when decommissioning USB sticks, and these devices can cause data breaches when sold as second-hand devices or recycled into new electronics.

3.2.2 Delete and Erase Functions

When files are written to a storage device, the device must be running some type of file system (e.g., FAT, NTFS). The job of file systems is to keep a record of the existence and location of all files and folders written to the storage device. When a file is deleted, the record of the file is deleted, but the file's content remains and can usually be recovered. Over time, when additional files are written to the device, the deleted files may become overwritten, at which point they are no longer recoverable [97].

To improve the security around file deletion, *DoD 5220 Block Erase* requires that a file is overwritten (erased) a minimum of three times and then verified. An even higher level of security is obtained by erasing an entire storage device, ideally using the device's internal secure erase function. These functions can either execute a slow secure wipe operation or in more modern drives can quickly delete cryptographic keys that were used to encrypt each file on the device, making the data permanently unintelligible [97].

For highest levels of assurance, the UK's National Cyber Security Centre (NCSC) recommends for secure sanitisation of hard disk drives and other magnetic media to degauss them using a strong magnetic field, for devices with a solid-state disk component to only store encrypted data and use data erasure or factory reset tools provided by the manufacturer, and for other flash-based media (such as

USB sticks) to physically destroy them using an “office shredder or disintegrator designed to produce particles no greater than 6 mm” [98].

3.3 Methodology

We investigate and compare the UI for removing files in both macOS 10.14 and Windows 10. We focus on these two OS as they account for more than 97% of the desktop/laptop OS market share [99]. A portable storage device is deemed necessary to ensure comparable testing conditions with both OS and we select a USB stick as representative for devices in this category due to its ease of handling. We perform an exploratory data collection using a streamlined Cognitive Walkthrough (CW) method to gain insights into how users may perceive the functionality of file removal operations in macOS 10.14 and Windows 10.

CW is a commonly used method for evaluating how well a system supports users towards achieving their goals. It places a particular focus on the user’s cognitive activities, e.g. their goals and knowledge [100]. This method is characterised by having an evaluator work through a series of tasks from the user’s perspective, and to evaluate the system’s ability to provide users with cues and prompts to guide them towards task completion. Note that an inherent characteristic of all usability inspection methods is that they do not involve study participants as representatives for typical end users [101]. Indeed, only the pluralistic walkthrough provides the option to involve such participants in group meetings together with developers and human factors experts.

We oriented ourselves on the process described by Rieman *et al.* [102] and Spencer [103] to prepare our CW. The context is defined by the UI’s of macOS 10.14 and Windows 10. The user has basic familiarity with both OS and understands that the terms *erase* and *delete* denote similar concepts. The two goals were to (1) erase a single file on a USB memory stick and (2) erase all files on a USB memory stick. The necessary sequence of actions consists of locating the target for erasure, the appropriate UI elements to erase the file, and lastly erasing the file.

We installed both OS on separate devices and ensured that they were fully updated[4]. We followed the streamlined CW approach by Spencer [103], conducting a step-by-step analysis of how the UI could guide the user attempting to execute the necessary sequence of correct actions. At each step of this process, we assess the visual cues available for the next action and the feedback given to the user after each action.

3.3.1 Forensic analysis

Before each CW, we restored the test USB stick to its *factory state* and analysed it with *FTK Imager Lite 3.4.3.3*[5] to confirm that no residual data was residing on the device. We then created a text file containing *lorem ipsum* placeholder text and saved this file inside a folder on the USB stick. At the end of each CW, we forensically analysed the USB stick with FTK Imager Lite to determine whether the CW had resulted in a delete or erase operation.

3.3.2 Process

The CW were conducted by Andreas Gutmann and evidenced with screenshots and note-taking. Mark Warner sighted the screenshots and notes, and verified that they fulfilled the necessary sequence of actions and were consistent with a typical user being guided by UI cues and prompts.

3.4 Results

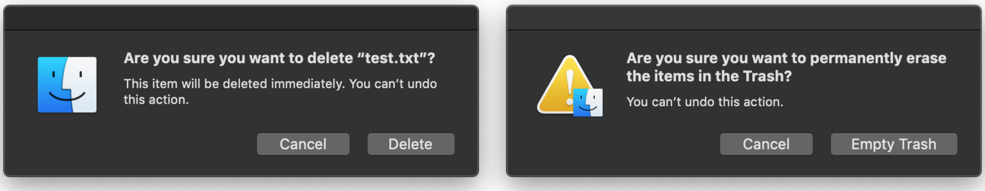
In this section we report on the results from our CW following the process described in section 3.3. Although we maintained a detailed record of step-by-step user actions during each CW, we limit our reporting to UI screens presented to users that are relevant to either delete or erase functions. We present findings from a total of nine goal-oriented CW using two different OS. We then report the results from our forensic analyses which determine the effectiveness of these functions. In doing so, we can identify any inconsistencies between the UI's reported functionality and the underlying technical operation.

[4] All updates available via the built-in update functions in early January 2019 were installed on the respective devices.

[5] `https://web.archive.org/web/20190902114111/https://forensicswiki.org/wiki/FTK_Imager`

3.4.1 macOS 10.14

Goal: Erase a single file. To remove a file from a USB stick, the user can locate the USB stick in the *Finder* application and move the file to *Trash*. As the file is still visible in the *Trash*, the user can attempt to further remove it using either of two methods. (1) The user can right mouse button click on the file to open the context menu and select *"Delete Immediately..."*. This opens a new dialogue window, which will inform the user that this action will immediately delete the file (see fig. 3.1a) and cannot be undone. The CW concludes when the user confirms the operation by selecting the *"Delete"* button. (2) The user can right mouse button click on the *Trash* symbol in the *Dock* to open the context menu and select *"Empty Trash"*. This opens a new dialogue window, which informs the user that this action will permanently erase all files in the *Trash* and cannot be undone (see fig. 3.1b). The CW concludes when the user confirms the operation by selecting the *"Empty Trash"* button. Under both conditions, our forensic analysis was able to recover the test file.



Are you sure you want to delete "test.txt"?

This item will be deleted immediately. You can't undo this action.

Are you sure you want to permanently erase the items in the Trash?

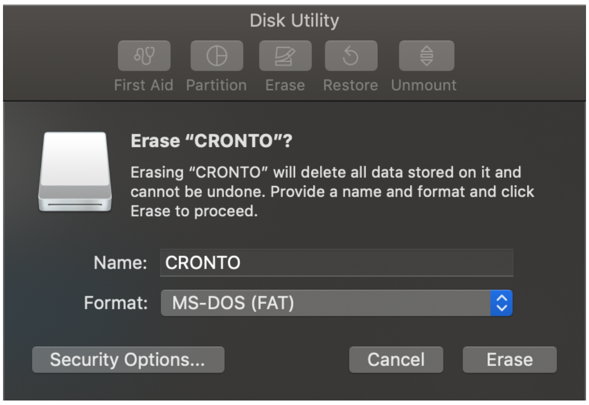
You can't undo this action.

(a) Dialogue when deleting a single file from the *Trash*.

(b) Dialogue when deleting all files from the *Trash*.

Figure 3.1: macOS 10.14 dialogues when deleting the test file from the *Trash*.

Goal: Erase all files on a USB stick. To remove all files on the USB stick, the user has two options. (1) They can remove all files similar to the removal of a single file (see above). Using this method entails that the files are deleted and likely to be recoverable under a forensic examination, as described above. Alternatively, (2) the user can launch the *Disk Utility* application and select the *"Erase"* option on the top feature bar. This opens a new dialogue window, which informs the user that this action will delete all data stored on the USB stick and cannot be undone. (see fig. 3.2). The CW concludes when the user confirms the operation by selecting the *"Erase"* button. Our subsequent forensic analysis was able to recover the test file.



Erase "CRONTO"?

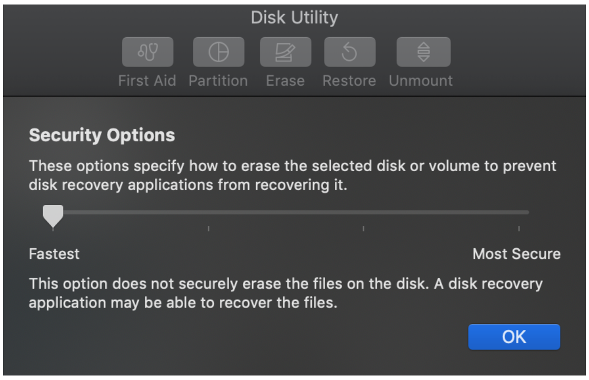
Erasing "CRONTO" will delete all data stored on it and cannot be undone. Provide a name and format and click Erase to proceed.

Name: CRONTO

Format: MS-DOS (FAT)

Figure 3.2: macOS 10.14 dialogue when erasing the USB stick with *Disk Utility*.

In a variation to the above procedure, the user can select the *"Security Options"* before selecting the *"Erase"* button. This opens a new dialogue window (see fig. 3.3) where the user can select a range of security options. On the default option, the dialogue window informs the user that this will not securely erase the files and disk recovery applications may recover them. For the other three options, the dialogue window informs the user that the function will erase the data. The CW concludes when the user makes a selection and confirms the operation when selecting the *"OK"* button followed by the *"Erase"* button (see fig. 3.2). Our forensic analysis was able to recover the test file when using the default security option but unable to recover the file when using any of the other three secure erase options.



Security Options

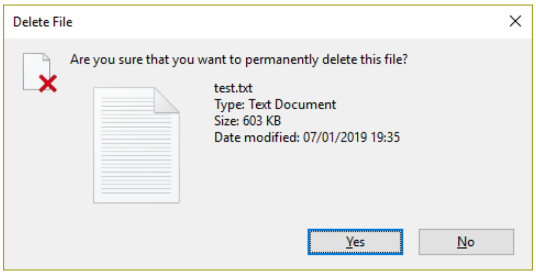
These options specify how to erase the selected disk or volume to prevent disk recovery applications from recovering it.

This option does not securely erase the files on the disk. A disk recovery application may be able to recover the files.

Figure 3.3: macOS 10.14 dialogue to select *Security Options* when erasing the USB stick with *Disk Utility*. The lower description changes as different options are selected on the horizontal slider.

3.4.2 Windows 10

Goal: Erase a single file. To remove the test file from the USB stick, the user can locate the USB stick in the *Explorer* application and physically press the keyboard delete button whilst the file is selected. This opens a new dialogue window[6], which informs the user that this action will permanently delete the file (see fig. 3.4). The CW concludes when the user confirms this operation by selecting the "*Yes*" button. Our subsequent forensic analysis was able to recover the test file.



Delete File

Are you sure that you want to permanently delete this file?

test.txt
Type: Text Document
Size: 603 KB
Date modified: 07/01/2019 19:35

Figure 3.4: Windows 10 dialogue when deleting the test file from the USB stick.

Goal: Erase all files on a USB stick. To remove all files on the USB stick, the user has three options: they can (1) proceed similarly to the removal of a single file[7] (see above), (2) access the "*Format*" dialogue from the *Explorer*, or (3) access the application "*Disk Management*".

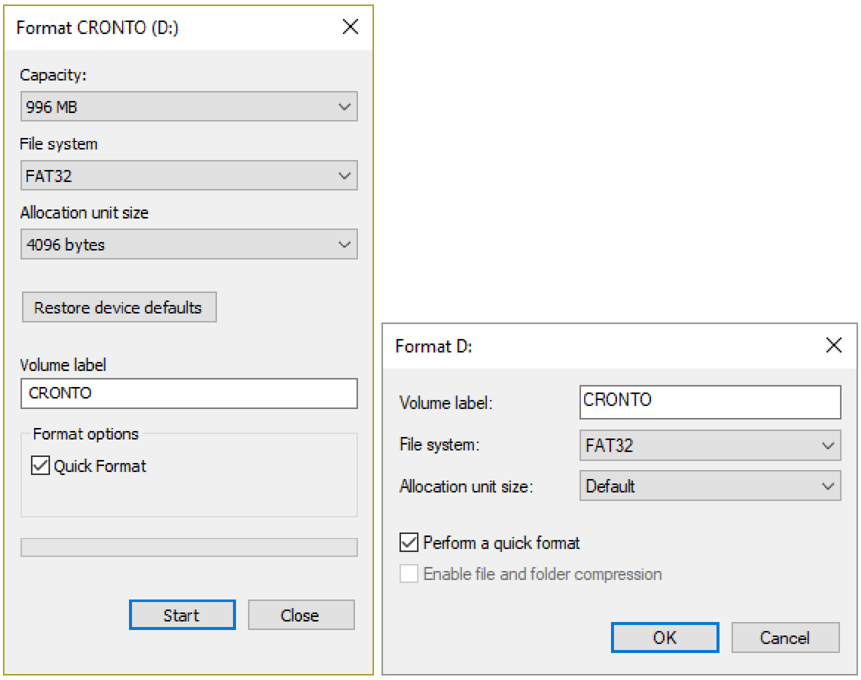
If the user chooses to access the *Format* dialogue in *Explorer*, a new dialogue window opens (see fig. 3.5a), where the user can confirm the operation by selecting "*Start*". A second dialogue window informs the user that this will erase all data (see fig. 3.6a). The CW concludes when the user confirms this operation by selecting "*OK*". After performing this quick format operation, our forensic analysis was able to recover the test file. In a variation to the above, the user deselects "*Quick Format*" (which is selected by default) before selecting "*Start*". Our forensic analysis was unable to recover the test file after this operation.

If the user chooses to access the *Disk Management* application, they can select

[6] Windows 10 treated our USB stick as *removable media*, which is why files were not placed in the *Recycle Bin* first. This might differ under other circumstances but is unlikely to affect the overall result of this CW.

[7] This option would entail that the files are deleted and likely to be recoverable under a forensic examination.

"*Format...*" from the context menu of the USB stick. This opens a new dialogue window (see fig. 3.5b), where the user can confirm the operation by selecting the "*OK*" button. A second dialogue window informs the user that this will erase all data and suggests making a backup before formatting the USB stick (see fig. 3.6b). The CW concludes when the user confirms the operation by selecting "*OK*". After performing this quick format, our forensic analysis was able to recover the test file. In a variation of the above, the user can deselect "*Perform a quick format*" (which is selected by default) before selecting "*OK*". Consistent with previous results our forensic analysis was unable to recover the test file.



Format CRONTO (D:)

Capacity: 996 MB

File system: FAT32

Allocation unit size: 4096 bytes

Volume label: CRONTO

Format options

☑ Quick Format

Format D:

Volume label: CRONTO

File system: FAT32

Allocation unit size: Default

☑ Perform a quick format

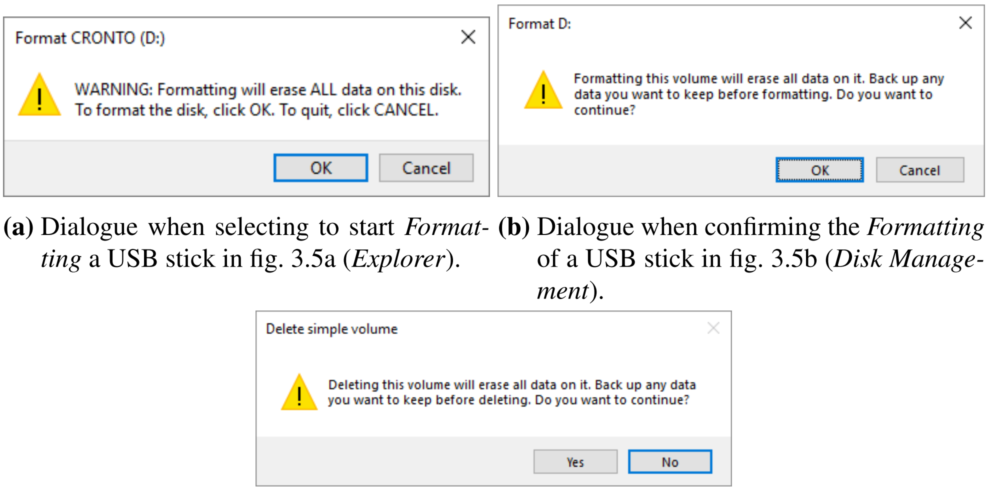
□ Enable file and folder compression

(a) Format dialogue accessed via *Explorer*.

(b) Format dialogue accessed via *Disk Management*.

Figure 3.5: Windows 10 dialogues when erasing the USB stick.

Alternatively, within the *Disk Management* application, the user can select "*Delete Volume...*" from the context menu of the USB stick. This opens a new dialogue window, which informs the user that the action will erase all data and suggests making a backup before proceeding (see fig. 3.6c). The CW concludes when the user confirms the action by selecting the "*OK*" button. After performing this delete volume operation, our forensic analysis was able to recover the test file.



Format CRONTO (D:)

WARNING: Formatting will erase ALL data on this disk.
To format the disk, click OK. To quit, click CANCEL.

Format D:

Formatting this volume will erase all data on it. Back up any data you want to keep before formatting. Do you want to continue?

Delete simple volume

Deleting this volume will erase all data on it. Back up any data you want to keep before deleting. Do you want to continue?

(a) Dialogue when selecting to start *Formatting* a USB stick in fig. 3.5a (*Explorer*).

(b) Dialogue when confirming the *Formatting* of a USB stick in fig. 3.5b (*Disk Management*).

(c) Dialogue when confirming to *Delete Volume* of USB stick. (*Disk Management*)

Figure 3.6: Windows 10 confirmation dialogues for formatting and deleting a volume.

3.4.3 Results of forensic analysis

Our CW identified three methods for removing a file from a test USB stick when using macOS 10.14 and six methods when using Windows 10. However, after completing a forensic examination of our test USB stick after performing each method, the test file was fully recoverable after two of the file removal methods in macOS and after four of the file removal methods in Windows 10. (see table 3.1).

3.5 Discussion

Modern OS for computers commonly provide accessible data delete functionality to users. Yet, data erasure functions for entire drives are typically located at deeper levels of administrative tools, whilst functionality to erasure individual files is not provided without expert knowledge or the use of third-party software.

Restricting these functions can protect users from accidental data loss. However, omitting information, guidance, and functionality can place lay users – especially those in the role of data controller – at risk of causing accidental data breaches. This could result in data subjects having their data exposed and organisations being non-compliant with data protection legislation.

In the following section, we discuss the results from our investigation of delete and erase functions in macOS and Windows, and suggest alternative UI design ap-

Table 3.1: Summary of our forensic analysis for various methods to remove data from USB sticks. Data removed with a delete function was successfully recovered, data removed with an erase function was not recoverable.

System	Function		Forensic evaluation	
			Deletion	Erasure
macOS 10.14	**Goal**	Erase single file		
	➤	Finder	✓	
	Goal	Erase all files		
	➤	Disk Utility (default options)	✓	
	➤	Disk Utility (changed options)		✓
Windows 10	**Goal**	Erase single file		
	➤	Explorer	✓	
	Goal	Erase all files		
	➤	Explorer Format (default options)	✓	
	➤	Explorer Format (changed options)		✓
	➤	Disk Management Delete Volume	✓	
	➤	Disk Management Format (default options)	✓	
	➤	Disk Management Format (changed options)		✓

proaches. We focus on default options and the terminology used to label and describe these functions in the UI, and then discuss the relevance of sufficient guidance for users. Entwined into these discussions, we argue for OS-dependent changes to the UI and highlight OS-independent implications from our findings.

3.5.1 Default options

macOS 10.14 and Windows 10 provide the functionality to securely erase all data from a USB stick. Yet, both OS use default options that reduce the effectiveness of these functions. We suspect that these default options are designed to increase the speed in which these operations are executed, with delete operations being much faster then erase operations to execute. Under macOS 10.14, the *Disk Utility* application contains security options (see fig. 3.3) to "specify how to erase the selected disk". Its default option contains a description that the files may be recoverable using certain data recovery applications. Figure 3.5 shows two UI screens for formatting a drive in Windows 10, with options "Quick Format" and "Perform a quick format" preselected. These options do not provide the user with any form of description. In both OS we were able to recover the test file when these default options were set.

Recommendation 1: Defaulting an option is commonly understood by users as a recommendation, reducing the likelihood of other options being selected by the user [104]. In the context of this research, default options discourage users from securely erasing files. Yet, those users might have significant interests in a secure erasure. We recommend an active selection process which encourages users to make an informed decision. In Windows 10, for example, the single confirmation button in fig. 3.5 could be replaced with two confirmation buttons to actively select between “Quick Format” and “Full Format”.

3.5.2 Incorrect terminology

Inconsistent and incorrect terminology was used to label delete and erase functions across both OS. For example, deleting a file (or multiple files) from the *Trash* in macOS is labelled as both *delete* and *erase*, depending on whether individual files or all files are deleted (see fig. 3.1). However, our forensic analysis found that both of these functions perform a delete operation, as in both cases the test file was fully recoverable.

Recommendation 2: Incorrect use of the terms delete and erase in OS UI might reinforce colloquial use and foster the misunderstanding that they denote the same technical function. This interferes with users’ ability to make informed decisions. We argue that the terms erase and delete should be used exclusively in relation to their technical meaning. In some cases the outcome of an operation (i.e. whether the OS will execute an erase or delete function) depends on future input from the user, e.g. in fig. 3.2 the outcome of pressing the *Erase* button depends on possible changes to the default security option. Under such circumstances, we recommend labelling the confirmation button with a neutral term, e.g. *Proceed*, and customising the description text depending on the selected security options.

3.5.3 Insufficient guidance and cues

During our CW we encountered multiple dialogue screens with insufficient or inadequate descriptions of underlying technical operation. For instance, the descriptive text in fig. 3.1 provides macOS users with a warning that they “can’t undo this action”. Whilst it may not be possible for users to undo this action using native functions within the OS, forensic software can fully recover these files. This can, therefore, create a false sense of security that these files are no longer recoverable. In Windows 10, when a file is deleted from the system, the final description of the function (see fig. 3.4) informs users that the file will be “permanently deleted” but lacks detail on what “permanent” means and whether the file could, under certain conditions, still be recovered.

Recommendation 3: Informative and accessible descriptions are required for informed decision making. Information related to a user task should not be exclusively accessible through optional UI screens. On each screen, where a user can make a selection, the relevant consequences of this decision should be explained. We suggest adding informative text to describe the difference between delete and erase functions where it is contextually relevant within an OS UI. Furthermore, a note about the existence of file recovery applications should be added to all delete function confirmation screens.

3.5.4 OS-independent implications

Recommendation 4: Designers of UIs rely on metaphors to make complex and abstract functions more intuitive and comprehensible for users [105]. For instance, placing an unwanted *file* into the *recycle bin* uses multiple metaphors from an office environment, allowing users to relate these complex computing artefacts and processes to everyday physical items and actions. Yet, the *delete* and *erase* metaphors are problematic, as they denote different meaning in the UI, whilst relating to the same constructs in the physical world. Designers should, therefore, consider integrating new metaphors that better distinguish between these two functions to reduce the risk of confusion for users.

Recommendation 5: As well as being well-positioned to provide users with guidance on the appropriate use of delete and erase functions, OS can also provide appropriate cues and prompts towards more secure outcomes. In section 3.2.1 we discussed past research showing how people intend to erase data from decommissioned drives but fail to do so securely, with researchers being able to recover data using digital forensic techniques. We propose OS should detect when a user deletes all (visible) files from a memory storage device, e.g. a USB stick. Upon detection of this event, the OS could remind the user about the difference between delete and erase functions, nudging the user to take an informed decision before potentially decommissioning said device.

Recommendation 6: In addition, we suggest official guidelines and recommended practice be developed on GDPR compliant erasure of data. This would be informative to users and provide OS with a single source for developing consistent UI functionality across platforms. The European Data Protection Board[8] may be best positioned to develop these as they are already tasked with issuing guidelines, recommendations, and best practice on other GDPR-related topics (Article 70 GDPR) and consist of representatives from each national data protection authority (including EEA countries).

Recommendation 7: Lastly, national data protection authorities could make recommendations to carry out a data protection impact assessment (DPIA) for the process of decommissioning data storage devices, since this activity can be "likely to result in a high risk to the rights and freedoms of natural persons" (Article 35 GDPR).

3.6 Limitations

Cognitive walkthroughs are limited in that they do not involve users (other than the researchers themselves), the results are solely based on the skills and expertise of the researchers, and the frequency of identified problems cannot be estimated. This means that cognitive walkthroughs commonly only identify a subset of usability issues of the evaluation system. However, we do not believe this limitation reduced the validity of the issues identified in our analysis. The file system of the USB stick used in our study was set to FAT32 as it is the most commonly used file system for this type of device. We do not anticipate different file systems would have affected our findings but further work would be needed to confirm this.

[8] See `https://edpb.europa.eu`.

3.7 Summary

We investigated possible causes for unreliable data input around delete and erase functions, which was identified as a privacy and security threat in the context of decommissioned USB sticks. In two of the most commonly used OS in today's market, we identified inconsistencies in the UI, insecure default options, and confusing and occasionally incorrect guidance. We propose design changes that could alleviate these issues and motivate a "call for action" for official guidelines and best practice on data erasure to be developed. Finally, our research evidences the need to evaluate those tools used for tasks that rely on accurate data input by non-expert users.

Chapter 4

A Tale of Typos and Fraudsters: The Challenge of Securing Online Banking Transactions

4.1 Introduction

Online banking is a fixture in modern society and economy. The majority of residents in the EU rely on it, with some countries reporting a usage-rate of above 90% [17]. More than 20% of electronic, single payment credit transfers in the EU in 2018 were initiated through online banking, totalling more than 4.3 trillion Euro [106]. The degree to which the European banking sector relies on the digitisation of its processes is apparent when considering the decline in bank branches by more than a quarter over the past decade [107].

But online banking is also an increasingly attractive target for cybercriminals and, thus, exposes banks and their customers to financial risks. The Kaspersky Security Network, for example, detected over 10 million attacks aimed at stealing funds and financial data (from users of Kaspersky's security products) in the first half of 2019, a 93% year-on-year increase [108]. The UK's National Cyber Security Centre reported that one group of cybercriminals alone caused financial losses of hundreds of millions of pounds in the UK [109] and warns users about fraudulent bank transfer activity as a result of an infection with financial malware [110].

To control these risks, some banks in the European Union integrated transaction authentication technologies (see section 2.4.2) in their processes. Later it became a recommended practise by European institutions [111, 112] and was eventually made mandatory throughout the European Union [113]. Today, transaction authentication is considered an essential technology to secure online banking against malware attacks.

By integrating transaction authentication technologies in online banking, transactions are conducted in two essential steps: sending transaction instructions to the payment service provider (transaction initiation) and verification that the instructions received by the payment service provider are correct (transaction authentication), whereby the latter comprises of state-of-the-art solutions–according to the classification methodology published by the European Union Agency for Cybersecurity [24] – to mitigate risks from infections of financial malware on users' devices.

The effectiveness of state-of-the-art transaction authentication solutions has been the focus of several studies. Most recently, Haupert and Gabert [114] found that only 18% of their participants were able to detect malware-like manipulation of transaction instructions. Yet, due to the quantitative nature of these studies, little is known about factors that seemingly affected this low incident detection rate, how to address these issues, and how to improve state-of-the-art transaction authentication mechanisms. In this work, we present the first qualitative investigation into people's behaviour when authenticating online banking transactions. We present a rich corpus of data gathered through a multi-method user study. Our work presents an important step towards better understanding the strengths and limitations of current state-of-the-art transaction authentication technologies. By uncovering a broad spectrum of factors that can impact people's behaviour when using these technologies, we provide suggestions on possible improvements to the state-of-the-art and how it is regulated. Our main findings are:

1. Concerns about undetected typos can overshadow concerns about consequences of undetected fraud, causing insecure behaviour that could be exploited by fraudsters.
2. Participants reported difficulties when initially learning how to use transaction authentication technologies. Eventually, some participants developed an automatism for these activities that resulted in insecure behaviour.
3. Some participants misunderstood the purpose of related security codes as being for user authentication or were generally confused about what was expected of them for the security of their online banking accounts.

4.2 Background: Online Banking Consumer Obligations for Fraud Reimbursement in Europe

Online banking security is predominantly supplied through a mixture of technological and regulatory means. Its effectiveness depends on not only the correct use of state-of-the-art technologies but also on the adherence to other obligations by all relevant parties. In the EU, such obligations are regulated via the Payment Service Directive 2 (PSD2, Directive (EU) 2015/2366) [113], which ensures harmonisation of the payment service market throughout the European Union. It is the second generation of this regulation (cf. Directive 2007/64/EC) in only eight years to reflect the rapid growth of the electronic and mobile payment market, including a large number of technical innovations and the emergence of novel types of payment services [113, recital 3].

The PSD2 describes not only mandatory features of and requirements on technologies to secure online banking transactions but also regulates the related conduct of online banking users as a contributing factor of "utmost importance" [113, recital 69]. Failure to adhere to these obligations by the payment service provider or customer can cause a shift of the entire liability for online banking fraud on the respective party. Consequently, payment service providers are mostly encouraged to invest in technologies that enabled them to comply with their obligations and focus on the prevention of fraud which they would otherwise have to reimburse. Thus, the regulatory context of online banking security and fraud reimbursement is an essential component in understanding the goals, requirements, and context-of-use of transaction authentication technologies.

When considering the regulatory framework created by a directive of the European Union, it is important to note that a directive in the context of European Union law is a legal act that requires member states to transpose its content into separate legislation to achieve a particular outcome without stipulating the precise implementation. This distinguishes directives from regulations, which are self-executing and become immediately enforceable as law in all member states simultaneously. Thus, the national implementations of PSD2, and the regulatory environment in which they are implemented, are of relevance to understanding the detailed conditions for fraud reimbursement in each member state. In the following subsections, we provide background information on relevant European and national legislation of online banking transactions and fraud reimbursement.

4.2.1 Different types of transactions

A transaction account is created through a contract between a user[1] and a bank. The user can loan money from or deposit it in the account, thus creating a debt in either direction. Users can access (the money on) their accounts through a visit to a branch of their bank, cheques, ATMs, online banking (including mobile banking), telephone banking, mail banking, and banking cards (credit and debit).

A transaction from such a bank account is the on-demand reduction of a deposit (or increase of a loan) with the corresponding value either added to the debt-balance of another account or paid-out. Transactions between two distinct bank accounts are commonly called payments and further distinguished between *push* and *pull* payments. Push payments are where the user has been provided with the relevant account details of another entity and initiates the transaction from their account to that of the other entity, e.g. via online banking. Pull payments are where the user has provided another entity with the relevant account details and authorised them to pull funds from their accounts, e.g. direct debits.

While the various types of transactions are similar at their basic functionality of moving money, and might thus not differ significantly from a user's perspective, they are treated very differently by legal systems. Consumer Protection organisations from the private and public sector, such as the UK's consumers' association *Which?* [115, 116] and Germany's *Ministry for Rural Areas and Consumer Protection Baden-Württemberg* [117], have long warned that fraud victims in online banking and in particular those of so-called authorised push payment fraud — a term to describe a transfer of funds initiated and authorised by the account holder but manipulated through technical means or social engineering, which was intended for a different recipient or purpose, independent of whether the fraud victim could have reasonably become aware of that discrepancy [118] — are not subject to the same legal rights and protection as if they were to use other payment methods, e.g. regarding the right to obtain a refund of fraudulent transactions.

[1] Without loss of generality, we define a user in this context as a natural rather than a legal person.

4.2.2 Differences in protection for different types of transactions

4.2.2.1 Pull payments

Pull payments ordinarily do not require transaction authentication technologies. Thus, we only briefly describe the context of consumer protection for pull payments to lay the context for differences in consumer obligations.

When transferring funds through other means than online banking, customers in the EU can make use of various rights to revoke the transaction. For example, European Union law guarantees the rights to withdraw from credit agreements within 14 days [119] and to obtain unconditional refunds of direct debits within a minimum of 8 weeks after the transaction date [113]. Beyond that, credit card companies such as Visa, MasterCard, and American Express voluntarily offer a compensation system called charge-back within 120 days of a purchase. In some cases, such as under the specific conditions of Section 75 of the UK's Consumer Credit Act 2006 [120], credit card transaction can be annulled years after they had been authorised. Online banking transactions, on the other side, are only subject to conditional refunds depending on the user's behaviour and the context of the transaction, as explained in the next subsection.

4.2.2.2 Push payments

In the following, we take a closer look at the obligations for users of online banking services with respect to conditional reimbursement of fraudulent push transactions. We primarily refer to the UK's implementation of PSD2 and provide a reference to the respective German laws in parenthesis.[2] In the UK, PSD2 was implemented in the Payment Services Regulations 2017 (PSR) [121]. In Germany, it was primarily implemented in the Payment Services Supervision Act (Zahlungsdiensteaufsichtsgesetz, ZAG) [122] and the civil law provisions in the German Civil Code (Bürgerliches Gesetzbuch, BGB) [123] with further adjustments to other laws, e.g. in the German Banking Act (Kreditwesengesetz, KWG) [124]. In table 4.1, for the ease of overview, we provide a list of the specific laws from both countries refer-

[2] We consider the regulatory frameworks in the UK and Germany since these are the countries in which we conducted the user study presented later in this chapter.

enced in remainder of this section.

UK law	German law	UK law	German law
Reg 72 PSR	§675l BGB	Reg 75(4) PSR	§675w BGB
Reg 74(1) PSR	§676b BGB	Reg 76(1) PSR	§675u BGB & §676b BGB
Reg 75(1) PSR	§675w BGB	Reg 77(4) PSR	§675v BGB
Reg 75(3) PSR	§675w BGB	Reg 65 PSR	§675i BGB

Table 4.1: Overview of laws referenced in section 4.2.2.2 on consumers' obligations for fraud reimbursement of push payments in the UK and Germany (split into two-by-two columns to preserve space). Laws with similar effect are shown in juxtaposition. PSR refers to the UK's Payment Services Regulations 2017 and BGB refers to Germany's Bürgerliches Gesetzbuch.

Regulation 72 of the PSR (cf. §675l BGB) obliges online banking users to adhere to their bank's Terms and Conditions for any personalised device or agreed set of procedures capable of initiating a payment order, as long as these requirements are objective, non-discriminatory, and proportionate. Users must take all reasonable steps to protect their corresponding personalised credentials and notify their bank without undue delay on becoming aware of the loss, theft, misappropriation or unauthorised use of a personalised device.

Regulation 74(1) (cf. §676b BGB) requires users to notify their bank without undue delay after becoming aware of an unauthorised transaction and no later than 13 months after they were debited. Regulation 75(1) (cf. §675w BGB) places the burden of proof that a disputed transaction had been authorised by the user on the bank. Regulation 75(3) (cf. §675w BGB) clarifies that any proof of a disputed transaction having been authorised is not necessarily sufficient to prove that it had been authorised by the user, that the user acted fraudulently, or that the user failed with intent or gross negligence to comply with regulation 72. This means that, for example, it is not sufficient for the bank to attest that the transaction had been authenticated with valid credentials and therefore to effectively require the user to prove that it was not them and that they were not grossly negligent or fraudulent in their conduct. Instead, each case will have to be treated on its own merits. If a bank claimed that their user acted fraudulently or failed with intent or gross negligence to comply with regulation 72, regulation 75(4) (cf. §675w BGB) requires that they provide evidence to the user in support of this claim.

Regulation 76(1) (cf. §675u BGB in conjunction with §676b BGB) stipulates that banks must reimburse users for unauthorised transactions reported within the time limits set in regulation 74(1). An exception to this regulation is if the user acted fraudulently or failed compliance with regulation 72 through intent or gross negligence, in which case the user is liable for such transactions according to regulation 77(4) (cf. §675v BGB). Regulation 65 (cf. §675i BGB) outlines specific and limited cases in which a bank and their customer may agree that the aforementioned regulations 76 and 77 do not apply for low-value transactions.

4.2.3 Differences in consumer protection with push payments

In the following, we take a closer look at the meaning of some of the conditions for reimbursement of unauthorised, fraudulent push transactions that were referenced in but not defined or mandated through PSD2.

4.2.3.1 Protecting credentials

The financial regulatory body in the UK, the Financial Conduct Authority (FCA), stated in paragraph 8.178 of their PSR guidelines [125] that users need to protect any information known only to the user and their bank, such as the PIN or password for their online banking account. In paragraph 8.179, the FCA stated that the expectations on the protection of credentials need to be disclosed by the bank to the user in pre-contract communications and that it will take a case-by-case approach when deciding whether these expectations would be considered reasonable.

In German law, §1 ZAG [122] defines the relevant credentials as any personalised features issued by the bank to the user for the purpose of authentication. Reasonable measures, stipulated by the bank to protect these credentials, must not hinder or unreasonably restrict everyday use of the service[3] [126]. For example, the German Federal Court of Justice (Bundesgerichtshof, BGH) has previously ruled that users must be allowed to write down credentials and store them separately from

[3] Cf. BeckOK BGB/Schmalenbach, 52. Edition, BGB §675l Randnummern 4, 4a. Available at: `https://beck-online.beck.de/Bcid/Y-400-W-BECKOKBGB-G-BGB-P-675L-Gl-II-2` (Accessed: 17 January 2020)

other knowledge or objects that would need to be used jointly to authorise a transaction (e.g. at two separate locations within the same dwelling[4]) but shall not store them at the same location (e.g. by carrying the components necessary to authorise a transaction in the same purse[5]).

4.2.3.2 Gross negligence

Negligence is defined through UK tort law as a breach of one's duty of care, with a causal relationship to a harm that a reasonable person would guard against. But, crucially, UK law does not define the term *gross* negligence. While recital 72 of PSD2 [113] stipulates that gross negligence should mean more than mere negligence by a significant degree, it stopped short of defining the term but referred back to the respective national laws instead. The FCA expressed in their guidelines that they will adhere to the description given by PSD2 while still assessing each case on its own merits [125, paragraph 8.221]. This does affect rulings of the UK's Financial Ombudsman Service for dispute resolution between financial service providers and their customers but does not provide the degree of certainty a legal definition would have provided. As a consequence, the Financial Ombudsman Service noted their concern over an increasing trend by banks to falsely conclude that users would have acted with gross negligence and, thus, refuse reimbursement [127].

German law on negligence is comparable to UK tort law described above. *Gross* negligence has been defined through relevant case law by the German Federal Court of Justice (Bundesgerichtshof, BGH), which laid out requirements for gross negligence in the context of an unauthorised online banking transaction as an objectively severe and subjectively inexcusable breach of duty, whereby the former does not necessarily imply the latter[6].

[4] BGH, Urteil vom 17. 10. 2000 - XI ZR 42/00 `https://beck-online.beck.de/Bcid/Y-300-Z-NJW-B-2001-S-286-N-1` (Accessed: 17 January 2020)

[5] BGH, Urteil vom 5. 10. 2004 - XI ZR 210/03 available at `https://beck-online.beck.de/Bcid/Y-300-Z-NJW-B-2004-S-3623-N-1` (Accessed: 17 January 2020)

[6] BGH, Urteil vom 26.1.2016 – XI ZR 91/14. Available at: `https://beck-online.beck.de/Bcid/Y-300-Z-NJW-B-2016-S-2024-N-1` (Accessed: 17 January 2020)

4.2.3.3 Validity of Terms and Conditions (T&C)

In the UK, the FCA's guidelines provide examples for contractual conditions that would not be permitted [125, paragraphs 8.174 and 8.175]: Requiring users to open and destroy letters containing their PIN immediately, prohibiting users from writing down or recording their PIN in any form, or restricting users' ability to interact with authorised open banking service providers.

In Germany, the use and content of T&C are regulated in the German Civil Code [123, paragraphs 305 to 310] with general requirements on any valid conditions, e.g. that they must not be ambiguous, surprising, cause disproportionate disadvantage for the user, or cause an effective reversal of the burden of proof.

4.2.4 Summary of consumer protection with push payments

Thus, online banking users in the UK and Germany who became victims of fraud through unauthorised transactions, and notified their bank under these regulations, should get reimbursed unless they acted fraudulently, failed to take reasonable steps to protect their credentials, or were grossly negligent in their failure to comply with the bank contract's T&C – unless these T&C were discriminatory, disproportionate, or not objective.

But on the other side, no legally mandated protection extends to cases in which an online banking user authorised a transaction to another bank account that turns out to be fraudulent, e.g. manipulated by financial malware on their devices. This is independent of whether they had taken all sensible steps to protect themselves and not behaved negligently or fraudulently, but were deceived by sophisticated means and could not have reasonably been expected to become aware of the nature of the fraud.

4.3 Related Work

In this section, we cover domain-specific related work on robust study design for investigations of sensitive topics such as online banking, past work exploring the effectiveness of transaction authentication mechanisms, more general related work on investigations into people's understandings of security processes, as well as related work on the specific methods we used in our study to investigate people's understanding of their use of these technologies.

4.3.1 Designing user studies for sensitive topics

For most people, their personal finance and online banking-related activities are sensitive topics, and researchers should pay special attention when investigating them. De Laine [128] explained that sensitive behaviour is that which takes place in "back regions", a personal or private space only available to people of high trust. Conduct done in back regions can be emotionally charged [129] and put the observed at risk by abuse or exposure from an observer [128]. Online banking commonly takes place in private spaces, hidden from prying eyes, and is generally considered one of the most sensitive tasks on the Internet [130]. Conversations related to money or financial values are commonly considered sensitive [131], even with close friends or family [132, 133], and can be a cultural taboo if done in public [134].

When data from comparable investigations of similar topics to that of inquiry is sparse, researchers should put particular focus on data validity in their study design. Krol *et al.* [135] recommended that human subject computer security research should give participants a primary task other than security, ensure participants experienced something as close as possible to realistic risk, avoid priming of participants, perform double-blind experiments where possible, and assign meaning to the terms threat model, security, privacy, and usability. On the same topic, Pfleeger and Caputo [136] outlined that resource-constrained human subject research should place particular focus on controlling confounding variables and minimising biases.

4.3.2 Effectiveness of transaction authentication mechanisms

Few user studies on online banking security mechanisms have been published. Al Zomai *et al.* [137] used a remote, email-based online banking simulation when studying whether participants would notice a Man-in-the-Browser attack, noting that "it would be impossible to create an environment with real risk" for their study. They manipulated the transaction context data in two out of ten emails and observed that this setup was prone to such attacks with a success chance of 21%.

Hartl and Schmuntzsch [138] evaluated transaction security with a more realistic simulation of online banking interactions by incorporating security codes delivered via SMS and an authenticator in the design of their simulated online banking platform. Participants conducted several transactions, among which one had the destination account number and transaction value manipulated, which was detected by 29% of participants. Qualitative data indicated that participants might have been unaware of the risks which potential malware infections of their devices could add to their online banking interactions. The authors described that the background of the study was disclosed to the participants but it remains unclear whether they were aware of the particular security-focus on that study.

Haupert and Gabert [114] asked employees of a software company to compare the usability of a simulated online banking website with two different security technologies by submitting two transactions. Participants had a primary task other than security and were purposefully unaware of the security focus of the study. The second of two transactions used the technology participants were most familiar with and a simulated malware attack manipulated the destination account number. This modification was detected by 18% of participants. The researchers found that the detection rate was statistically significantly better for participants who compared the transaction details with a printed invoice rather than the computer screen, the technology they used in the second transaction, their technical affinity, as well as their knowledge about and amount of experience with online banking.

Krol *et al.* [139] conducted a longitudinal user study of 2FA in online banking. They observed confusion about the "excessive" security measures taken online, compared to using only a PIN at ATMs and for offline payments. Participants were unable to memorise a successful sequence of actions and create a corresponding automatism for the use of hardware tokens over the study duration of eleven days.

4.3.3 Mental models in security

A mental model is a simplified, psychological representation of real-world situations that describes how a system works in the user's mind. It can be technically inaccurate, borrow elements from other systems the user believes are conceptually similar, and assists in decision making through derived expectation on how possible actions would influence the system state. Mental models are naturally evolving through interaction with the system (and sometimes other systems).

Research has shown that people are more likely to adhere to security behaviour recommended by experts if it aligns with their mental models. L. Jean Camp [140] discussed how five types of generalised mental models could be sufficient to communicate the basic principles of computer security to lay-people. Rick Wash [141] reasoned that study participants might have had no intention to comply with experts' computer security advice because it was in contradiction to their mental models about hackers and viruses. Similarly, in a study by Renaud and Volkamer [142], participants rejected the offer to learn about a new PIN memorisation technique because it was not aligned with their existing mental model for PIN management.

Mental models change over time and can become more detailed as people gain new knowledge about the system. Fulton *et al.* [143] found that people developed more complex mental models of computer security threat models based on information available to them. They further found that inaccurate depictions in hacking-related TV shows and movies lead to an inaccurate understanding of risks and possible countermeasures. Murillo *et al.* [144] found that novice users' understanding of online data deletion covered only a subset of what security and privacy experts thought they should know. When comparing the complexity of participants' views, the researchers found two distinct categories of novice user's: *backend-aware* and *UI-based*. Similarly, Abu-Salma *et al.* [145] found that novice participants had a user-centric view of communications networks, while security experts' had a network-centric perception. Kang *et al.* [146] found that the complexity of participants' mental models of the Internet correlated with the depth of their technical knowledge about it.

Mental models inform decision making and a more accurate understanding can lead to more secure behaviour. Vaniea *et al.* [147] found a correlation between an accurate mental model of software updates to operating systems and users' secure behaviour. Raja *et al.* [148] found that incomplete understanding of how certain

OS features interact with each other can lead to users selecting insecure firewall settings. Renaud *et al.* [149] found that incomplete understanding of threat models and countermeasures hampered adoption of end-to-end encryption in email. Zhang-Kennedy *et al.* [150] found that improving participant's understanding of password guessing attacks and related threat models lead to a more secure password choice one week after the intervention.

Understanding user's mental models can be used as a basis to design systems that better communicate relevant risks to users and, thus, lead to more secure behaviour. Raja *et al.* [151], building upon mental models elicited in their previous work [148], designed security warnings for the functionality of a personal firewall. Their designs improved the users' understanding of the functionality of a personal firewall and the consequences of their actions and were preferred by the majority of participants.

Clustering of mental models can be feasible and a population can be segmented accordingly, but not if sampled from domain experts. Wash and Rader [152] found that different groups of users in a representative sample of the Internet-using US population had different beliefs about computer security and corresponding self-reported behaviour. Bravo-Lillo *et al.* [153] found that participants with a background in computer privacy or security had vastly different mental models of security warnings compared to those without such background.

4.3.3.1 Elicitation of Mental Models

Researchers can draw from a large number of available data collection methods to elicit users' mental models, such as thinking-aloud demonstrations, drawing tasks, participatory modelling, interviews, focus groups, and free listing tasks [154, 155], which can be categorised into verbal, graphical, and hybrid [156]. Royer *et al.* [157] recommends combining multiple elicitation methods, as any single method likely yields an incomplete understanding of people's mental model.

Thinking-aloud demonstrations have been found particularly effective in identifying concepts accessed by participants during a task [158] and the sequence in which people access them [159]. Norman [160] points out that such data directly generated by participants may be incomplete or contain erroneous elements, and recommends supplementing such data with descriptions of user activities, e.g. video recordings, as they suffer from fewer biases.

Drawing tasks have been used to supplement thinking-aloud demonstration as a vehicle for participants to express thoughts they may find difficult to articulate [161]. This combination has also been described as effective to avoid narrow, incomplete understanding of people's mental model [162].

Semi-structured interviews have been described as an effective method to elicit the rationale behind participants' behaviour [163] and decision making process [164]. Conducting them after a drawing task was found to be particularly effective in revealing discrepancies between different participants' mental models [165].

Morgan *et al.* [166] recommend between 20 and 30 participants in mental model research to likely reveal at least once any belief held by 10 per cent or more of the population. Participants should be sampled using a wide range of methods, such as intercept surveys [167, 168] and through engagement with social organisations (e.g. youth clubs and civic groups). To evidence the seriousness of researchers and research topic, they suggest compensation for each participant's time in the range of $10 to $20.

4.4 User Study

This study aimed to collect rich qualitative data from current online banking users on their use and understanding of transaction authentication technologies. Motivated by recommended practise outlined in section 4.3.1 on the design of user studies on sensitive topics, we placed particular focus on aspects of data validity and, thus, took the following high-level measures:

- Since our study aimed to investigate online banking behaviour, we requested participants to access their online banking account and demonstrate how to make an online banking transaction. We assume that accessing their online banking account and handling their own money has introduced a feeling of realistic risk, as recommended for this type of study by Krol *et al.* [135].

- Although the focus of our research was on transaction authentication mechanisms in online banking, we framed the research study during any communication with study participants as being more generally on the topic of online banking to avoid unnecessary priming. Following this narrative, online banking was the primary task and securing their online banking interactions the natural, secondary task for our participants. These measures follow recommendations by Krol *et al.* [135].

- To improve internal validity, as recommended for this type of study by Pfleeger and Caputo [136], we decided to triangulate our findings through a multi-method research design [169, 170]. Following best practise in mental model elicitation research (see section 4.3.3.1), we included verbal, graphical, and hybrid methods [156] through a combination of a questionnaire, demonstration, drawing task, and semi-structured interview, as recommended by Royer *et al.* [157].

4.4.1 Procedure

We collected data through a set of four measures: Questionnaire, thinking-aloud demonstration, drawing task, and semi-structured interview. The latter three were video recorded.

During the questionnaire task, participants provided us with their demographic data, self-reported statistics on their payment and online banking-related behaviour, the Security Behavior Intentions Scale (SeBIS) [171], and the International Personality Item Pool-Interpersonal Circumplex (IPIP-IPC) [172]. All questions were optional. No participant omitted any questions.

Participants had been asked during the recruitment phase to bring with them any devices they need to make transactions from one of their online banking accounts. Here they were asked to demonstrate making a transaction to a bank account provided by the researcher while thinking-aloud, i.e. explaining their actions, their thought process behind them, and what they believe the system was concurrently doing. The researcher interrupted the participants before they would have selected the final submit button.

Subsequently, participants were asked to make and explain a drawing that visualises how making a transaction in online/mobile banking works. The order of these tasks was chosen such that participants may benefit from their short term memory and, thus, supported more thorough drawings. This approach can also help participants' to overcome articulation problems they might have had during the demonstration [161].

Finally, a semi-structured interview was conducted, traversing questions about participants' perceptions of usability and security in online banking, incorrect or fraudulent transactions, attack vectors against online banking, and reimbursement in cases of fraud. During the interview, participants were also asked to augment their previous drawing with explanations of how someone could attack their online banking. Again, the order of tasks was chosen to take advantage of participants' short term memory and maximise benefits to the study's internal validity.

The study design and material were piloted with three researchers from the department of economics. Study participants were compensated for their time with a choice of shopping vouchers with a value of £20 / 20€ and the majority completed the study within 60 to 90 minutes. Thus, participants were compensated above the national living wage of the respective countries.

The study was conducted in one city in Germany and one city in the United Kingdom to gather data from a wider range of online banking technologies and implementations. Data saturation, as a subjective measure of redundancy in the data at the time of data collection [173], was reached. Since the study was designed in accordance to best practise for participant sampling in mental model research [174, 166], it is likely to yield data that is representative for the majority of data subjects in the sampled population and, thus, can yield generalisable design recommendations.

4.4.2 Analysis

The questionnaires were digitised and time-coded verbatim transcripts with discourse analysis of the video recordings were made by Andreas Gutmann. He then re-read the transcripts and pre-coded significant quotes or events as "codable moments" [175] before applying descriptive codes. Mark Warner independently coded 2 participants ($\approx$10% of the data). The resulting codes were discussed and both agreed on a coding scheme. Andreas Gutmann applied the agreed coding scheme and themed the data using an inductive approach. Emerging themes were augmented with nested magnitude codes [176] in 3-quantiles: *few, a small number, rarely* $<$ *some, several, common* $<$ *many, frequent, most*. The findings were discussed and agreed upon by Andreas Gutmann, Mark Warner, and Steven J. Murdoch.

The participants' drawings had been completed in two phases and were analysed accordingly. Andreas Gutmann and Mark Warner coded the drawings deductively based on the themes *UI-focused* and *backend-aware* described for drawings of computer privacy and security concepts by Murillo *et al.* [144]. The results were discussed and agreed upon. Participants had augmented their initial drawings in a different colour to explain during the interview how someone could attack their online banking. These modifications, and the transcripts of their explanations, were coded deductively by Andreas Gutmann based on expert knowledge about threat models.

4.4.3 Recruitment

Leaflets to advertise the study were distributed with a street intercept approach in pedestrianised shopping areas of the respective cities, at online bulletin boards dedicated to the respective geographic areas, and on the websites of local community organisations for senior citizens. Appointments were scheduled via email.

4.4.4 Ethics

A realistic online banking experience, as necessary for our robust study design, introduces challenges for an ethical study design as it requires to either use a real system or the use of deception. The *APA Ethical Principles of Psychologists and Code of Conduct* [177, principle 8.07] requires for the use of any deceptive techniques in research that any effective non-deceptive alternative procedures have been excluded as not feasible. We found that subject to appropriate and stringent control measures, it was feasible for participants to use their real online banking accounts to prepare an online banking transaction. The main control measure was that, in order to minimise the actual risk, the researcher had to interrupt participants at the end of the demonstration task *just-in-time* before submitting their transaction. Our study design was approved by our university's Research Ethics Committee (ref: 10817/002).

4.4.5 Limitations

Online banking is a highly sensitive topic. More security wary people might have hesitated to participate, especially since our participants were asked to demonstrate use with their own accounts. We tried to compensate for this by adhering to best practise for participant recruitment in mental model research to use a wide range of recruitment methods, highlighting our university affiliations and the study location in a university building, and emphasising that participants may use their own laptops during the demonstration.

Due to risk compensation bias, participants could have behaved more or less secure during the study, compared to their usual behaviour, as a response to the university environment and the presence of the researcher. Yet, since the effects of risk compensation are usually rather small or do not occur at all [178], any such bias would likely have had a negligible impact on our qualitative findings since it does not aim to deliver accurate measurements of behaviour.

4.5 Findings

We recruited 21 participants, eleven from Frankfurt, Hesse in Germany and 10 from Cambridge, Cambridgeshire in the UK. Participants are labelled as P1 to P21 in chronological order of recruitment, beginning with those recruited in Frankfurt.

An anonymised version of the data we collected during this study has been made available[7] through the UCL Research Data Repository under Creative Commons CC BY-SA 4.0 (Attribution-ShareAlike) license with digital object identifier

[7] Temporarily embargoed until publication of thesis or peer-reviewed paper at journal or conference.

(DOI) 10.5522/04/12860834. This approach follows the best practice to preserve research data of value long-term and make it available to others.

4.5.1 Participants

We recruited 10 male and 11 female participants, aged 19 to 73 (median 26, average 30). 57% of participants were students and 62% had completed some academic education. Participants' average and median SeBIS score, a scale to measure their awareness of and intention to adhere to common computer security recommendations, were 3.22 points and 3.14 points (out of a possible maximum of 5), respectively. Table 4.2 provides a detailed list of these items for all participants.

	Age	Gender	Profession	Completed education	SeBIS score
P1	26	male	research assistant	M.Sc.	3.2125
P2	26	male	student	M.Sc.	4.0625
P3	30	female	manager	Ph.D.	4.0375
P4	22	male	student	A level	3.7875
P5	27	male	student	B.Sc.	3.3833
P6	22	male	student	B.A.	2.7458
P7	33	male	technician	A level	3.3208
P8	26	female	unemployed	A level	3.4167
P9	19	female	student	A level	2.9250
P10	21	female	student	A level	2.7042
P11	36	male	mechanic	high school	3.5250
P12	69	female	retired	B.Sc.	3.0917
P13	26	male	student	M.Sc.	2.5250
P14	21	female	student	B.Sc.	3.2458
P15	20	female	student	A level	2.7833
P16	20	male	student	A level	2.8667
P17	33	female	medicine	B.A.	3.1375
P18	28	male	student	M.A.	2.2250
P19	33	female	student	M.Sc.	3.1375
P20	24	female	administrator	M.Sc.	2.9958
P21	73	female	retired	Ph.D.	4.5250

Table 4.2: Demographic data of participants. Study locations, i.e. Frankfurt or Cambridge, are not included in this table since we did not require participants to be residential in the corresponding geographic areas.

Participants were asked five Likert scale questions about their preferences for and trust in online banking. The majority of participants preferred online banking over physical interactions, placed trust into their bank to secure their transactions, and said to adhere to instructions read from the bank's website. Participants were split over whether to value personal interactions with bank employees and being worried about the security of their transactions. Figure 4.1 provides detailed results for each Likert scale question.

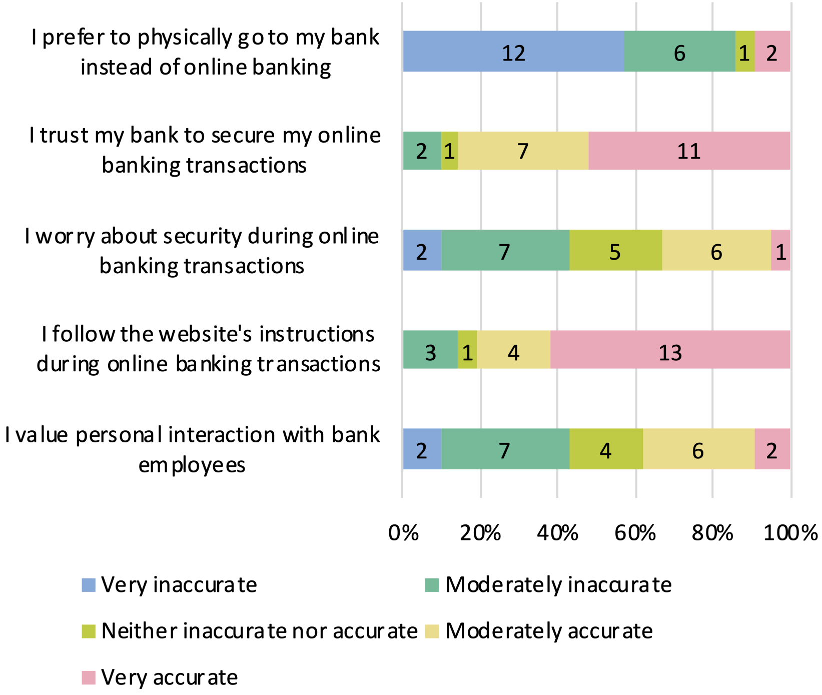
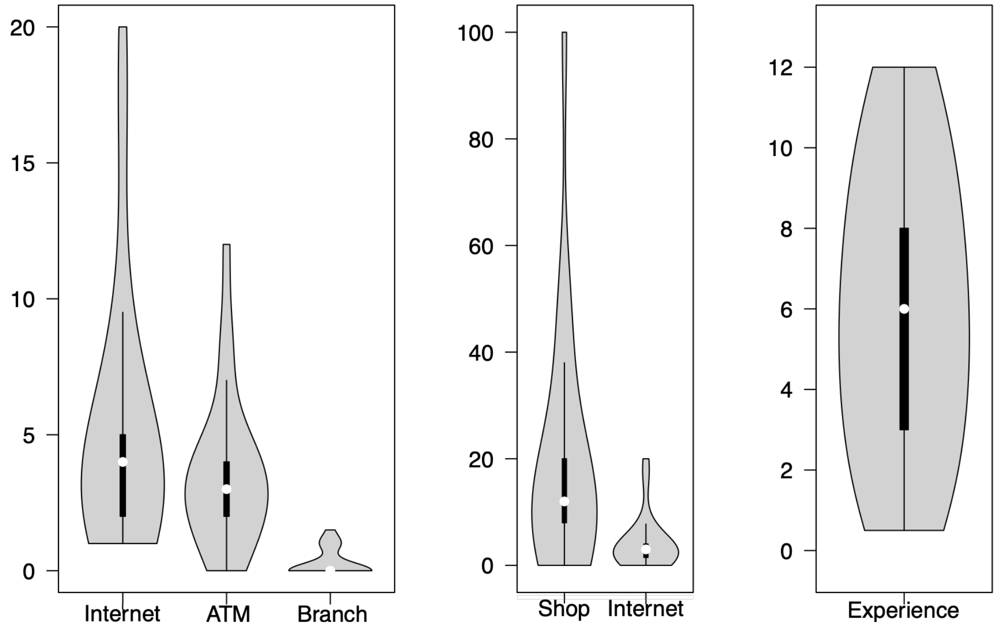


Figure 4.1: Responses in the questionnaire to statements with Likert-scale answers to assess participants' preferences for and trust in online banking. Digits denote number of participants who gave the corresponding response to each statement.

We asked participants to best-guess statistics about their usual payment and transaction related behaviour. Their answers indicate that they are familiar with the system of online banking by virtue of frequent use and multi-year experience. Violin plots for detailed statistics are provided in fig. 4.2.

4.5.2 Demonstration

Participants' demonstrations of online banking transactions involved the services and products of nine different banking institutions. Thirteen participants used a laptop to access their account, while the other eight used apps on their phones. Details are provided in fig. 4.3.



(a) Self-reported statistics of how many transactions participants usually conduct through Internet banking, withdraw money at ATMs, or initiate transactions inside their bank's branch per month.

(b) Self-reported statistics of how many card-based payments participants usually make in person or over the Internet per month.

(c) Self-reported statistics of participants' years of experience with Internet banking (not limited to their current bank).

Figure 4.2: Violin plots of self-reported statistics around participants' transaction related behaviour and experience. White circles show the medians, box limits (thick black lines) indicate the 25th and 75th percentiles, whiskers (thin black lines) extend 1.5 times the interquartile range from the 25th and 75th percentiles, polygons (grey areas) represent density estimates of the data and extend to the extreme values.

Fifteen of the participants demonstrated a transaction with a technology that required a security code. Eight of them had an authenticator to receive or generate the security code, while six participants received a security code via SMS. The remaining participant used two apps on their phone, the main banking app to initiate their transaction and a separate app dedicated to the purpose of transaction authentication to receive a security code.

Another six participants used one app on their phone to initiate the transaction without additional use of transaction authentication technologies to secure their transactions. These participants presumably (and possibly unwittingly) relied on the integrity of their mobile phone's Trusted Execution Environment (see section 2.3.2) and potentially additional security measures such as run-time application self-protection.

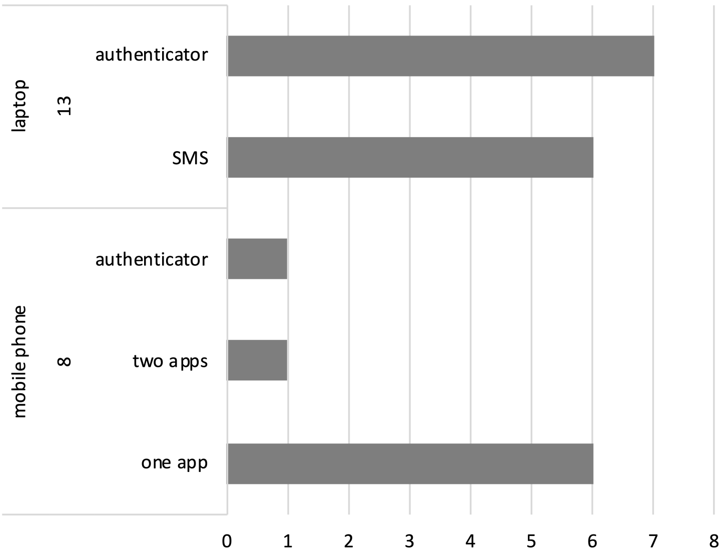


Figure 4.3: Combinations of devices used by participants to initiate and authenticate transactions. Thirteen participants used a laptop, seven in combination with an authenticator and six with SMS. Eight participants used their mobile phone to initiate transactions, whereby one of them authenticated their transaction with an authenticator and one with a second app on their phone. The remaining six participants who used their mobile phone to initiate a transaction were not required by their bank to authenticate their transaction using another device or app.

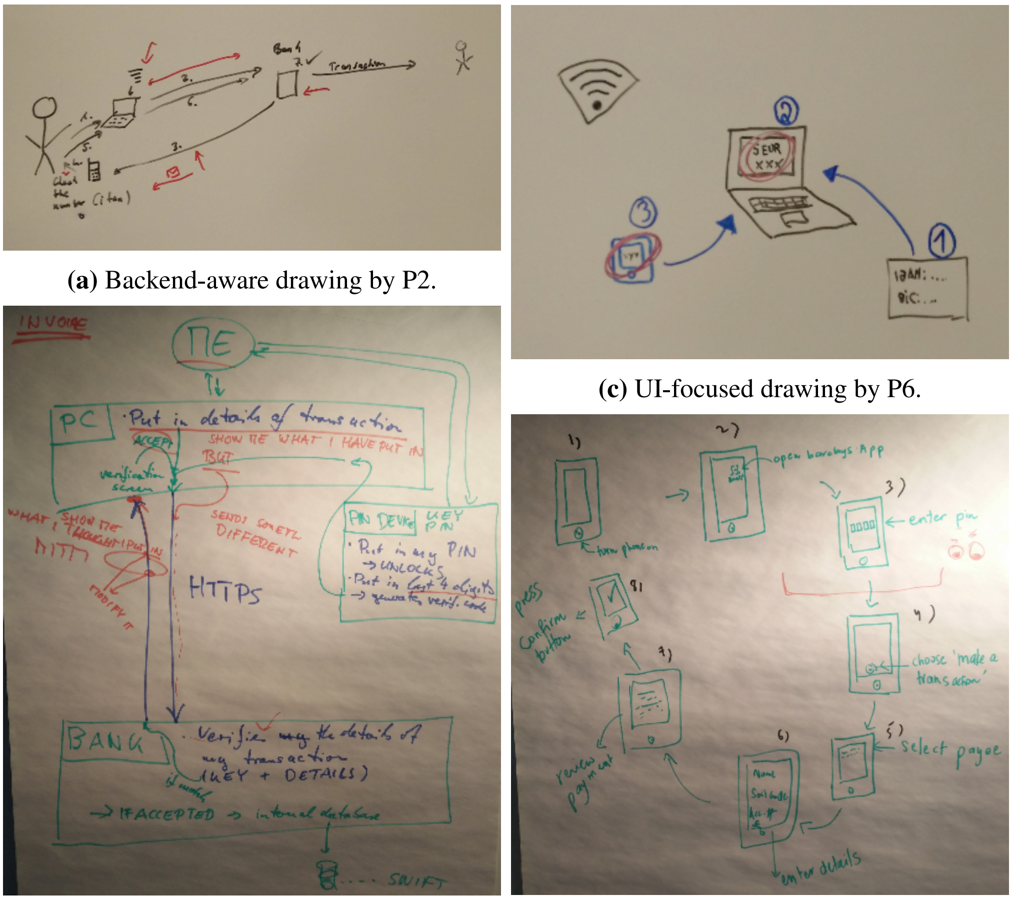
Two participants experienced substantial difficulties with the completion of the online banking demonstration. One participant first initiated a transaction on a laptop but expressed confusion when the system requested a security code. They eventually gave up on the attempt and demonstrated a payment using an app on their phone, which did not require the use of a security code. This participant is thus classified as using a mobile phone without transaction authentication. The other participant was equally unaware that their bank would request a security code for a transaction to a new recipient and expressed their understanding that if they would press the next button, then their transaction would be completed and the money transmitted. The researcher chose not to correct this falsehood at that moment in order to avoid introducing an unnecessary authority bias into the remainder of the study, thus ending the demonstration task in line with the study protocol. This participant used an authenticator but is classified as *incomplete* where appropriate, since his use of this technology could not be determined.

Of the 14 participants who used security codes to actively secure their transactions, eight did so in an insecure way by not correctly verifying the integrity of the transaction data before quoting the security code. This insecure behaviour includes counts of participants not verifying the correctness of the transaction context data or comparing it with the screen of their laptop instead of the printed instructions they had received, either implicitly relying on the integrity of their device's screen or on the accuracy of their short-term memory, respectively. The former would have made them immediately vulnerable to a Man-in-the-Browser attack which tampers with the transaction context data as well as their screen output (see section 2.2.2.4) and the latter could still make them vulnerable to a Man-in-the-Browser attack if it coincides with automation bias, i.e. the user's implicit or explicit over-reliance on the correctness of the output of an automated process [179, 180]. The six participants who conducted transaction authentication in a secure way by verifying that it matched with the printed instructions did so using a laptop and an authenticator.

4.5.3 Drawings

Participants' drawings were either focused on the electronic communication between different devices (e.g. laptops, mobile phones, authenticators, bank servers, etc.) or on how the user had to operate those devices (e.g. filling out forms, pressing buttons, copy security code, etc.), and coded those drawings as *backend-aware* or *UI-focused*, respectively. This is in line with findings by Murillo *et al.* [144], who found that *backend-aware* and *UI-focused* appear to be natural groupings for non-expert user's understanding of complex IT processes. We classified nine drawings in the former category and 12 in the latter. Four example drawings are depicted in fig. 4.4 and the coding results are listed in table 4.3.

Participants' understanding of how their online banking could be attacked, based on their drawings and corresponding explanations during the interview, was coded according to threat models for *malware* (see section 2.2.2.4, as example), *Man-in-the-Middle* (see section 2.2.2.3), and *illegitimate account access* (see sec-



(a) Backend-aware drawing by P2.

(b) Backend-aware drawing by P13.

(c) UI-focused drawing by P6.

(d) UI-focused drawing by P15.

Figure 4.4: Examples of participants' drawings in blue, green, and black colour of how making a transaction in online/mobile banking works. The red colour was subsequently used during the interview to describe how someone could attack their online banking.

tion 2.2.2.2, as example). Nine participants identified hackers manipulating their devices, e.g. via malware, as a threat. Seven participants explained that someone could hack the Internet connection between them and the bank. 13 participants expressed concerns about someone else logging into their online banking account. The detailed coding is available in table 4.3.

Of the six participants who securely verified the transaction context data during the demonstration (as described in section 4.5.2), three drawings were backend-aware and three were UI-focused, five identified malware on their devices as a threat to their online banking, two described a Man-in-the-Middle attack, and one expressed concerns about when an attacker would be able to login to their account.

	Verified transaction context data	Malware	Man-in-the-Middle	Illegitimate account access	Drawing type
P1	×			✓	UI
P2	×	✓	✓		Backend
P3	✓	✓			UI
P4	×	✓	✓		Backend
P5	✓	✓			Backend
P6	×			✓	UI
P7	×	✓		✓	Backend
P8	×		✓	✓	Backend
P9	✓		✓	✓	UI
P10	✓	✓			UI
P11	×			✓	UI
P12	N/A			✓	UI
P13	✓	✓			Backend
P14	N/A			✓	UI
P15	N/A			✓	UI
P16	N/A	✓	✓		Backend
P17	N/A			✓	UI
P18	N/A		✓	✓	UI
P19	Incomplete			✓	UI
P20	×			✓	Backend
P21	✓	✓	✓		Backend

Table 4.3: Overview of whether participants verified the transaction context data, the threats they considered relevant for online banking transactions, and their drawing type. *N/A* indicates participants who did not interact with security codes during the demonstration task, i.e. six participants who used a single banking app on their phone. *Incomplete* denotes a participant who assumed the transaction would have been completed before the step in which the bank would have required them to quote the correct security code for transaction authentication.

Of the eight participants who did not correctly verify their transaction data before using the security code, five were backend-aware and three were UI-focused, one identified malware as a threat, three described a Man-in-the-Middle attack, and six expressed concerns about when an attacker would able to login to their account. Of the six participants who were not required to use security codes to complete the transaction, one was backend-aware and five were UI-focused, one identified malware as a threat, two described a Man-in-the-Middle attack, and five expressed concerns about when an attack would be able to login to their account. A summary of this information is given in table 4.4.

Verified transaction context data	Malware	Man-in-the-Middle	Illegitimate account access	UI	Backend
✓	5	2	1	3	3
×	1	3	3	3	5
N/A	1	2	5	5	1
Incomplete	0	0	1	1	0

Table 4.4: Summary of statistics on which threats participants considered relevant for online banking transactions and their drawing type, grouped by whether participants verified the transaction context data. *N/A* indicates participants who did not interact with security codes during the demonstration task, i.e. six participants who used a single banking app on their phone. *Incomplete* denotes a participant who assumed the transaction would have been completed before the step in which the bank would have requested a security code for transaction authentication.

4.5.4 Thematic Analysis

We conducted a thematic analysis on the transcripts of the participants' thinking-aloud demonstrations, explanation of their drawings, and the interviews. This revealed four main themes about their perceptions of online banking transaction security: *feelings of personal responsibility for one's actions*, *importance of being familiar with the system*, *the meaning of online banking security*, and *concerns about fraud*.

4.5.4.1 Feelings of personal responsibility

Online banking is generally considered a sensitive task for which participants expressed feelings of personal responsibility for their actions and omissions. Most participants were concerned with typos in their transaction data. Comments about feelings of security were commonly made. A few participants expressed feelings of responsibility for their beneficiary's privacy.

Correctness of transaction data. Many participants voiced their concerns about typos when entering the transaction data. These concerns commonly surfaced multiple times during the demonstration, drawing task, and interview. For some participants, the concern about typos in their transaction data overshadowed being wary about malicious transaction manipulation. These participants verified the transaction data a number of times when entering it but abstained from verifying them again during transaction authentication, e.g. *"I could again check if this is what I have entered, but I am usually too lazy to do that and I rely on the computer"* (P1). As explained in sections 2.2.2.4 and 4.5.2, such behaviour is likely to result in a vulnerability against transaction manipulating malware. A small number of participants also found and corrected typos during the live demonstration, e.g. *"every time I type it incorrectly, like I did this time"* (P10).

Some participants expressed negative emotions and feelings of stress when entering the transaction data, e.g. *"the account number, oh, this one I hate entering"* (P21), while others relied on their ability to detect eventual typos at a later stage, e.g. *"I guess the app guards against that because it asks you to review the transaction about two or three times before you send it"* (P16).

A few participants described methods to mitigate these risks beyond careful entry and immediate verification of the data. One possibility is to utilise the transaction summary accompanying the TAN security code as the last opportunity to detect typos, e.g. *"this, so to say, is the last way to re-check your information"* (P7). Other participants described relying on a supposed system that would automatically verify that the name they entered matched the destination account's owner, apparently unaware that such a system did not exist, e.g. *"I think the bank checks if the account exists and the name matches"* (P13). Lastly, participants verified the correctness of the Bank Identifier Code (BIC) shown by their bank after entry of an International Bank Account Number (IBAN), e.g. *"if it shows the BIC, then I know the IBAN that I entered is correct"* (P8). Yet, the BIC only identifies a bank's branch and is independent of the recipient's account number. For example, the two IBAN DE73-5001-0517-0123-4569-89 and DE93-5001-0517-1111-1112-11 (both chosen at random and for illustrative purposes) refer to the German bank code 5001-0517 and account numbers 0123-4569-89 and 1111-1112-11, respectively. Yet, both IBAN refer to the same bank code and share the same BIC code: INGDDEFFXXX. Furthermore, although the IBAN features technical measures to minimise the risk of typos through a two-digit checksum in positions three and four, no participant voiced awareness of this feature (yet, this does not allow to conclude that no participant would be aware of this feature or implicitly make use of it).

Security as a feeling. Many participants described their rationale of using online banking despite its intrinsic security risks. Some explained that they expect their bank to protect them from malicious activity, e.g. *"I would always demand my bank to be prepared for it so it doesn't harm me"* (P8). A few disregarded security risks entirely under the assumption that fraudsters would only target more wealthy people, e.g. *"I'm so small, how would you pick me out of the crowd. I think you should look for people who have money"* (P14), while others voiced the rationale that banking could never be risk-free, e.g. *"I think, for me personally, I've got the balance between pragmatism and paranoia at a point where I can live with it, and I'm happy to live with it even if that does potentially open me to risks"* (P21).

Several participants described specific aspects of their online banking experience as trust anchors. Authenticators were commonly mentioned as guarantors of secure online banking experience due to being tangible and not directly connected to the Internet, e.g. *"that's something that is made not on the Internet, but like in the real world. So, you still have this contact to something real"* (P5). A few participants assumed positively correlated causation between friction in the user experience and an effective security mechanism, e.g. *"if there would be an additional, even another additional, step in the process, it might be nicer [...] don't ask me if I have a profound argument why I would just maybe feel even more secure"* (P1). A few participants described relying on communication from their bank to stay informed about relevant threats, e.g. *"you go into the bank there and they put very useful posters and that, about what to look out for. [...] I mean they do try to warn you about these things"* (P12).

A few participants expressed their desire for more security. One reason was for participants to offload some of the perceived responsibilities, e.g. *"I don't want to go to the branch, but maybe still have this human factor in it [...] so that I can conveniently say that, hah, it's done"* (P1). Others clarified that they would not mind more friction if that would make their interaction more secure, e.g. *"what I like about the process is that there are several checks [...] I'm okay with two, could be one more [...] you could say that you can only be online in your account for two or three minutes and after that you either have to login again or you can't do the transaction"* (P3). Lastly, a few participants expressed their belief that securing their online banking is of paramount importance, e.g. *"I would always like it to be more secure, no matter how secure it actually is. I think we're talking about money here. And with money you can not be careful enough. So, more security would always be welcome"* (P4).

Privacy. A few participants mentioned a responsibility to not expose their beneficiaries' bank account details to any potential adversaries. These feelings were manifested in participants deleting text messages from their phones and not saving trusted beneficiary account details in their online banking, e.g. *"I don't tend to store existing payee for security reasons, because I'm worried that sometimes I may put too much information of other people's account and that, if there's any security leak of the bank website, then I will probably expose other people's information"* (P19).

4.5.4.2 Being familiar with the system

Online banking platforms are complex technical systems which participants considered rather complicated to use. They described relying on consistent user experience for efficient and effective use and reported how they usually react to unexpected events.

Complicated to use. Some participants said that they found online banking difficult to use in the beginning, until they had memorised particular sequences of actions leading to the desired outcomes, e.g. *"for the first time it might be not easy. Second time it also might be not that easy. But once you did it a few times, it gets quite easy to use"* (P5). A few participants reported being unable to remember explanations or instructions previously received from their bank after not utilising it for an extended period, e.g. *"I don't know how it works. I have no idea. I think my support tried to explain it to me when I got it, but I don't remember"* (P10), and others voiced frustration over high cognitive load when they had difficulties retrieving a successful sequence of actions from their memory, e.g. *"oh, for days when you just went into the bank and talked to people. [...] All that was exceedingly stressful"* (P12). Only participants who demonstrated a technology with security codes are represented with this theme, i.e. no mobile banking.

Memorised sequence of actions. Some participants described relying on consistent interfaces and memorised sequences of actions to achieve their goals, e.g. *"the truth is that people very quickly, I mean myself included, you know, you get used and you do it, it works. Next time, and next time after that"* (P12). A few participants voiced their surprise if an interaction with the system was perceived as inconsistent, e.g. *"Oh, I did get a text, I don't usually get one. So, it's saying I got a text from my bank, saying you've added a new payee to your account"* (P20).

Reacting to the unexpected. Several participants reported on their usual response to unexpected events when transferring money. The majority explained they would repeat their last action or sequence of actions, e.g. *"first of all I would try it a second time. Starting all over, maybe refreshing the page. Doing every single step again. If it's not working, I would call my support"* (P10), while others expressed their immediate security concerns and would abort any interactions, e.g. *"I would immediately take a screenshot of my laptop, logout, and either write an email or call my bank"* (P7). A few participants expressed their belief that deviations from the system's expected behaviour would likely cause a strong emotional reaction, e.g. *"probably, first thing I would do would be to panic"* (P9).

4.5.4.3 What is online banking security?

Online banking is an abstract and intangible system. Participants explained it by drawing comparisons to more tangible offline communication systems. Participants also confused the purpose of security codes for transaction authentication in online banking with an OTP for user authentication, e.g. as part of a two-factor authentication system. Participants also described various insecure practises that could be exploited by an attacker.

Comparison to offline systems. A few participants compared entering the beneficiary's account details to writing an address on a postcard and described their understanding that transactions are manually processed by employees of the bank, e.g. *"if you have already submitted the transaction, and the people of the bank are not working on it, and it just doesn't get transferred because someone has to confirm it or do stuff behind the screen."* (P4). Others compared authenticators to keys required to unlock access to the security code, e.g. *"in order to open the lock [...] I use my card together with the TAN generator. [...] So, I have like the online part and an offline part coming together to open the lock."* (P5).

Purpose of security codes. Many participants explained the purpose of security codes while making an online banking transaction. Some participants falsely assumed the purpose was to confirm their identity and verify they are authorised to make a transaction, e.g. *"in this second app I'm supposed to log in again for my bank to be sure that it is me that is doing a transaction, because here I need a second password. [...] if I enter this password, my bank will be sure that is me who is doing that"* (P11). Others correctly understood the purpose to verify the correctness of the transaction details, e.g. *"if you're trying to use a TAN for a transaction that doesn't fit exactly the information that was sent to you via SMS it doesn't work. That's the good thing about it"* (P7).

Insecure behaviour. Many participants demonstrated practises that could be described as insecure. Some failed to verify the transaction data before entering the security code for transaction authentication. A few participants described problems reading the transaction data sent via SMS when doing mobile banking because their banking app would automatically log out, e.g. *"This is sometimes a bit hard, because as soon as you're leaving the app, it automatically logs you out, which, of course, is a good thing, in terms of security, but this way it's, it makes it really hard to conduct a transaction [...] I will have to catch the SMS when it comes up on the top of the screen. And then I will have to hold it down with my thumb, so it doesn't disappear again. And it makes it really hard, because then I have to remember the six-digit TAN and type it in"* (P8, explaining why they prefer to initiate transactions on a laptop rather than on their mobile phone). A few participants were typing the transaction data from the laptop screen into their authenticator. This could make them vulnerable to malware manipulating the screen of the device. A small number of participants used an authenticator with separate functions for login and transaction authentication but were unable to remember the purpose of the latter. Not understanding the semantics of data input to or the purpose of data output from transaction authentication technologies could make people more vulnerable to divulging such security codes during social engineering attacks [181].

4.5.4.4 Fraud

Online banking is inherently linked with concerns about losing money due to user error or fraud. Participants were either very optimistic or very pessimistic about getting reimbursed if they lost money. Those assumptions were predominantly based on hearsay, personal experience or logical deductions, rather than knowledge about legally binding guarantees and consumer protection laws. Several misconceptions about the ability to recall transaction and fraud insurance were unveiled.

Reimbursement. Participants frequently stated their beliefs about eventual protection if they lost money via online banking. Many participants argued that they should be reimbursed in any case unless they acted grossly negligent, e.g. *"I'm not worried about the transaction going wrong. And even if it does go wrong, the bank is going to respond to it surely"* (P14). Some participants were certain the burden of proof for gross negligence would be on the bank, e.g. *"unless the bank could make clear that it was my personal fault and I acted careless"* (P5). A small number of participants were uncertain about the conditions under which their bank would reimburse them, e.g. *"perhaps the bank should make it a little more transparent, you know, what they will do and what they won't"* (P12). A few participants explained that typos in their transaction details would not be reimbursed, e.g. *"anything that is me putting in wrong details, I think that's my own responsibility"* (P13), but fraudulent transactions would be, e.g. *"I'm guessing that whenever there is some kind of manipulation, I would get my money back, because there was a crime"* (P8).

Some participants expressed their doubts over being reimbursed by their bank for cases of fraud, e.g. *"I don't think they would do that. They like their money too much. No, I mean, if I have a virus on my computer, they would say, hey, it's* *not our fault if you're not secure enough"* (P9). A few participants argued against reimbursing victims of fraud generally, e.g. *"you can only confirm your transaction under your authorisation. Then it should be the customer's responsibility"* (P19). A small number of participants argued that although banks should reimburse them morally, they would not be compelled to do so, e.g. *"I would expect it and I would hope for it, but I wouldn't be very confident that they would do it. [...] If they didn't do it, I would probably understand"* (P16). A few participants stated that they would not trust their bank to reimburse them unless they would not be the only victim, e.g. *"if they had similar cases earlier, then they would probably give it back. Because with one customer they can do it, and say no, but if it's many that would be pretty dumb"* (P9).

A small number of participants said they would expect insurance to cover their losses, e.g. *"financial institutions have insurances which cover this normally to a certain amount. And I think I would get my money back."* (P2). A few participants thought they could recall transactions within a certain time frame, e.g. *"but I could also, as I said, just get the money back if I say I want it back in this six weeks span"* (P4).

4.5.5 Expected standard of care

Participants mentioned a wide range of general computer security practises that they considered relevant for online banking security. Most such mentions were linked to something they currently do, something they think they should do, or something they assume they could do to improve the protection of their online banking activities. As previously mentioned, our methodology is likely to have revealed at least once any belief held by 10 per cent or more of the population from which we sampled [166]. Thus, we can expect that our participants' understanding of their online banking security due diligence, i.e. the requirements for not gross negligent behaviour to avoid liability in case of unauthorised online banking fraud, is a subset of the collection of these practises. The items in this list were grouped by Andreas Gutmann based on the context in which they were mentioned during the study.

Social engineering. Participants mentioned that they should not give their credentials or access to their devices to untrusted individuals under any circumstances. In particular, regarding unsolicited phone calls, participants mentioned the importance of choosing a trustworthy phone company and reporting any suspicious phone calls to their bank.

Shoulder surfer, theft, etc. Participants described locking the screens of their devices, logging out after use, and the importance of reporting lost devices to their bank. They also mentioned their intention to use good passwords and the relevance of not saving them on their devices. Concerning physical proximity, they described that one should not do online banking in front of untrusted individuals.

Phishing. Participants described checking the URL of the online banking website and whether the interface showed differences to its normal appearance.

Malware/MitB. To protect their own devices from digital threats, participants mentioned the use of antivirus software, firewalls, and a separate browser dedicated to their online banking activities. They also described the benefits of using physically separate devices and the relevance of frequently updating the software on their devices.

Hacker/MitM. Participants advised against the use of public WiFi and to check that any connection to banking services is HTTPS. They also described not using their banking credentials for other services and frequently changing them.

4.6 Discussion

Online banking is a fixture in modern society and its security properties are an essential component. In this study, we investigated people's behaviour during and understanding of the most sensitive task in online banking: transferring money.

As is the nature in exploratory research, we aim at a better understanding of the causes and effect for secure and insecure behaviour, including observations about in the context of such behaviour. Therefore, we discuss insights from the study data where quantification is irrelevant and propose hypotheses on causal relations that can be derived from the study data where quantification is relevant.

4.6.1 Difference between mechanisms

We observed several counts of insecure behaviour during the demonstration task. This includes counts of participants not verifying the correctness of the transaction context data or comparing it with the screen of their laptop instead of the printed instructions they had received.

We noted insecure behaviour for all six participants who received security codes via SMS and the one participant who used two applications on a mobile phone but only for one out of seven participants with authenticators. This difference is significant (Fisher's Exact Test (FET), p=.005) and indicates that using authenticators for transaction authentication correlates with significantly more secure interactions than SMS-based mechanisms. It remains an open question for future work to assess the extent to which this correlation entails causation and the direction of such causation, e.g. whether authenticator-based mechanisms lead to more secure behaviour or whether people with more secure behaviour prefer to use (banks which offer) authenticators as a mechanism.

4.6.2 Knowledge

Relevant related work on the use of mental models in security research (cf. section 4.3.3) found correlations between the user's understanding of threat models and their corresponding behaviour [149, 150]. More general work on information security awareness [182] also suggests that better understanding of information security topics correlates with more secure and less insecure actions, but recommends situation-specific assessments of such correlations. In this subsection, we discuss the situation-specific assessment of this proposed correlation for transaction authentication security in online banking.

Participant's awareness of relevant threats was derived from their explicit drawings and explanations of threats against the security of their transactions, and categorised as malware, Man-in-the-Middle, or illegitimate account access (see table 4.3). We could not find significant correlations between their awareness of such threats and secure behaviour (FET *Secure behaviour* $*$ *Malware*, p=.138; *Secure behaviour* $*$ *Man-in-the-Middle*, p>.999; *Secure behaviour* $*$ *Illegitimate account access*, p=.103). Thus, awareness of threats (and presumably intention to guard against risks from those threats) seems to be only weakly correlated or uncorrelated with actual secure behaviour in online banking. This suggests that while it could be a contributing factor, knowledge about threat models alone is insufficient to explain the insecure behaviour we observed and that simply informing people about the existence of certain adversarial risks is insufficient to improve their practices.

One reason why such knowledge is not strongly correlated with actual behaviour could be that some participants had learned to do the right things without knowing the reasons for it or for the wrong reasons. P9, for example, assumed that security codes would be to authenticate oneself rather than one's transaction and was mostly concerned about someone else being able to login to their online banking account. Yet, P9 securely verified the transaction context data on their authenticator. It is unclear to what degree this could be related to the individual characteristics (of P9) and/or the specific online banking implementation (used by P9).

Another reason could be that participants with correct knowledge about relevant threats had additional knowledge which deterred them from securing their transaction. For example, P2 assumed that financial institutions have insurances which cover fraud up to a certain amount while P4 assumed that they could recall any transaction within six weeks and get their money back. In these cases and others that surfaced throughout the study, we believe that participants remembered protection schemes that apply only under very specific conditions but were unaware of those limitations, e.g. the EU's Directive 2014/49/EU [183] to protect an individual's deposit in case of a bank's insolvency and Directive (EU) 2015/2366 [113] to allow for the unconditional refund of direct debits within a minimum of 8 weeks after the transaction date.

It is not unreasonable if people overestimate the level of security and protection they could rely on during online banking. Financial institutions (like most businesses) focus on brand appeal and their reputation through public messaging and advertisements. Chandy *et al.* [184] showed that emotion-based appeals and positively framed messages are more effective in established and mature markets than argument-based appeals, expert sources, and negatively framed messages. Indeed, Mogaji and Danbury [185] analysed 1274 print advertisements by UK banks for emotion-based appeals and found that feelings of "relief and relaxation" and "secure" were likely to be induced by 43.5% and 21.8% of the advertisements, respectively. Emotion-based advertisements have also been shown to particularly boost long term memorability [186], which affects people's perception of the likeliness of a future event [187, 188]. Thus, it is reasonable to assume that such public messaging and advertisement behaviour understates the associated security risks of those services and, thereby, could bias users to overestimate the level of security and protection they could rely on.

Lastly, it could have also been that participants erroneously thought their actions would have guarded them against the relevant risks or assumed there would not be anything they could do against them – both leading to inaction despite awareness of these risks. We could not find evidence in our data to support these hypotheses but could not exclude them either.

4.6.3 Distractions

Many participants in this study were concerned about the risk of typos in their transaction data. Indeed, a few of the 21 transactions demonstrated in this study contained typos in the transaction destination account number or IBAN. It is not surprising that people are concerned about typos in their transaction details, as they can entail similar financial risks [189]. Furthermore, people evaluate the likeliness of a future event based on the ease of recalling corresponding examples from their memory [187, 188]. Risk management attributes risk severity to a combination of likeliness and impact. As a consequence, it is reasonable when online banking users intuitively allocate more mental resources to the avoidance of typos as the more severe risk.

Since attention capacity is limited [190], a large amount of attention devoted to typo avoidance likely comes at a cost of attention towards the detection of fraudulent manipulation. This would explain how and why some people forget to verify the transaction context data. For others, an effect called task-set inertia (which is an indirect consequence of our limited attention capacity) can temporarily inhibit performance when switching from one task to another [191]. This would explain how and why some people fail to detect a discrepancy in the transaction context data. In summary, the indirect consequences from our limited attention capacity combined with reasonably large concerns about typos in transaction data could be the cause for human error (see section 2.1) in online banking: people forgetting to validate the transaction details before entering a security code (mistake) or not noticing a discrepancy when validating the transaction details (slip).

Brumby *et al.* [192] showed that people intuitively add a short pause when switching tasks if error minimisation is a priority and that a 10-second delay significantly reduces the error rate. Where the design of transaction authentication mechanisms does not encourage such a long delay, the consequences of task-set inertia will likely reduce the effectiveness of the mechanism. This could explain why we observed people using authenticators having more secure interactions than those using SMS (see section 4.6.1), as the former arguably requires more time to handle than the latter while both arguably do not entail large cognitive loads (for the information retrieval task of locating the security code and the transaction context information). We argue that, unless users' concerns about typos are alleviated, delays of up to 10 seconds are a likely natural barrier on the efficiency of effective transaction authentication mechanisms.

4.6.4 Friction

A number of times topics around friction surfaced during the interviews. This had been expected since transaction authentication by itself could be considered friction – similar to confirmation screens for data deletion or erasure in most operating systems (cf. section 3.4, e.g. fig. 3.1). Indeed, no participant with a single-app mobile banking experience (i.e. no handling of security codes) reported any issues related to friction. Many of those who did handle security codes, on the other side, described online banking as a rather complex and complicated task. They commonly referred to difficulties in memorising a successful sequence of actions to access and use security codes and subsequently recalling and executing this sequence for easy and automatic processing with low mental effort. Some participants further described relying on consistent interface elements as cues for assisted recalling of their memorised sequence of actions.

The dual-processing theory of decision making [193] classifies the automatic processing of a memorised sequence of actions as the first of two mental processing systems (type 1): an unconscious, rapid, and automatic execution with low demand on concentration. The second system (type 2), in comparison, is responsible for conscious, slow, and deliberative task execution with a high demand for concentration. The prevalence of typos, as well as the execution of automated control processes to detect and correct errors [27], adds further evidence that many participants in this study likely processed large parts of the demonstration task in an automatic manner. Arguably, the detection of fraudulent transaction manipulation requires conscious and deliberate task execution, i.e. type 2 mental processing. This is evidenced by the fact that a number of participants omitted to check for such manipulation, i.e. a planning mistake rather than a slip (see section 2.1).

Thus, it is evident that the automated processing of memorised sequences of actions during online banking is a direct consequence of the (perceived) high complexity of the task itself and likely an important contributing factor to insecure behaviour in online banking. We want to stress that the issue is not the type 1 (i.e. unconscious, rapid, and automatic) execution of memorised sequences of actions *per se*, but the omission of essential steps for transaction authentication as a consequence of it.

One could assume that a reduction of the overall friction in online banking to an absolute minimum would remove the need for memorised sequences of actions and, thus, allow for more deliberate and conscious processing of all tasks. Yet, systems with minimal friction and fast interfaces have also been shown to result in more error-prone strategies when validating the correctness of data input [194].

A possible solution for this dilemma, called micro-boundaries, has been described by Cox *et al.* [195]: the purposeful and targeted introduction of some friction, as an active design choice, to an otherwise low-friction system, can facilitate user's switching from an automated task processing to more mindful and conscious interactions when and where it matters. Indeed, targeted interruptions to automated cognitive processes have also been found effective at reducing mistakes by clinical professionals [196]. To be effective, micro-boundaries should be small, easy-to-overcome barriers placed right before the targeted interaction to effectively interrupt the automated task processing.

Thus, it appears that authenticators might be more suited at shifting the passive type 1 mental processing into an active type 2 processing, compared to receiving security codes via SMS, due to an adequate increase of friction when compared to the remainder of the task. This could explain why we observed people using authenticators having more secure interactions than those using SMS (see section 4.6.1).

Beautement *et al.* [197], on the other side, discussed how systems with too much friction could be detrimental to operator's intention to comply with security procedures as they try to balance the actual and anticipated costs and benefits of their actions. Thus, it appears that a sweet spot in the amount of friction would be able to shift automated type 1 processing into more conscious type 2 processing without overloading the user with more (anticipated) effort than worth the (anticipated) benefits. Indeed, a few participants shared their understanding that friction was evidence for effective security mechanisms and clarified that they would not mind more if that would make their interactions more secure. In an analogy, we could compare friction in security with salt: too little or too much can be detrimental, but the right amount also depends on the circumstances.

4.6.5 Implications for practice

In this study, we observed that transaction authentication mechanisms in online banking do not always work as intended and gained insights into circumstances of when this is the case. We also know from the seminal work on usable security by Adams and Sasse [198], that most cases in which security mechanisms are not correctly operated by their users are best addressed through changes to the mechanism or to the communication with those users. In the following, we discuss possible improvements to mechanism and communication based on insights gained from our study and make further recommendations for regulators and legislators to address fundamental issues.

4.6.5.1 Implemenations

We observed participants' desire to give meaning to elements which they encountered during an online banking session that are otherwise meaningless to them. We also observed misconstrued conclusions from such unassisted meaning-making, e.g. that a correct BIC would indicate a correct IBAN. Existing features which could ease the mental workload of online banking, such as the two-digit checksum in the IBAN, were unknown. The mental workload due to the system's complexity can be high. Reducing unnecessary friction in online banking could make micro-boundaries more effective in shifting users' mental processing from the passive type 1 into the more active type 2 mode, thus improving user consciousness and affect overall online banking security.

Recommendations 1 & 2: We argue for the implementation of two measures to reduce the system's complexity: just-in-time reminders to assist users when recalling their memorised sequence of actions (e.g. "it is important for your security to now check the correctness of the transaction details one last time before you enter the security code, even if you have already checked them before" during transaction authentication) and the removal of unnecessary UI elements (e.g. the BIC for transactions within the EU).

4.6.5.2 Communication of banks with their customers

Banks commonly communicate with their customers about a plethora of topics to maintain a good relationship and for regulatory compliance. During this study, we noted two salient opportunities to improve the security and usability of online banking. Both cases are forms of security education, which according to Bada *et al.* [199] needs to be targeted, actionable, doable, and provide feedback on the user's actual behaviour. Wash and Rader [152] argue that it is particularly important to communicate the cause and effect of security measures. Yet, one has to be careful when introducing additional corporate communication as this can be counter-productive due to information overload [200], which can confuse the receiver and affect their ability to recall and act upon information [201].

Recommendation 3: Our first recommendation is related to participants' reported difficulties in learning how to make an online banking transaction. We also observed participants who had developed an automatism that led to insecure behaviour. We believe that the development of routines which include undesirable or omit essential elements is an indicator for insufficient guidance when users attempt to memorise and retain a successful sequence of actions for a complex task. We recommend that banks increase their efforts to support customers who are new to a transaction authentication mechanism and occasionally provide actionable reminders to all users about the essential elements of this interaction. Such a spaced repetition approach has previously been shown beneficial for retention of a wide range of knowledge and skill [202], including complex motor skills [203, 204] and historical [205] or obscure facts [206].

Recommendation 4: Our second recommendation is linked to participants' reliance on hearsay, personal experience, and logical deductions to determine whether, and under what conditions, they'd likely get reimbursed in case of fraud. Some participants recalled specific regulations that applied to other payment methods, such as direct debit or credit card payments, and wrongfully believed this would apply to all payment methods. We believe that clarifying the differences in legal rights and obligations between the various types of payment would help users to adopt correct behaviour for each context. We argue that banks should promote more specific information about the differences in rights and obligations between the various types of payments.

4.6.5.3 Regulatory and legislative environments

Recommendation 5: One of the main findings in this research was that some participants' concerns over typos in their transaction data overshadowed their concerns about malicious transaction manipulation, and such behaviour is reasonable since the former is arguable more likely than the latter, yet both can have similarly dire consequences [189]. We discussed in section 4.6.3 how this likely increases security risks for online banking users. Reducing such reasonable concerns about typos in transaction data would, thus, likely reduce these negative effects and consequently improve online banking security. One approach for this would be if banks were to ensure that the recipient name entered by a user also matches that of the destination account before executing a transaction. In that case, it could suffice to ease users' concerns about typos by verifying the correctness of the name of the recipient, a task that is certainly easier than the *status quo* of comparing long numbers without inherent meaning but would provide comparable protection against typos. Such functionality was announced by several banks in the UK under the name *Confirmation of Payee*. We recommend regulators and legislators everywhere to mandate the implementation of such functionality.

We also found inconsistencies and falsehoods in participants' understanding of what they are obliged to do to avoid being grossly negligent in their conduct related to online banking security. As outlined in section 4.2.3.2, there is notable concern by some authorities over an increasing trend by banks to falsely conclude user would have acted with *gross negligence* and refuse reimbursement on that ground. The Treasury Select Committee of the UK Parliament noted that different banks could interpret *gross negligence* in different ways and instructed the financial regulatory body to "require financial firms to produce an easy to read lists of 'dos and don'ts' for customers" [207]. Despite being a wicked problem [208], we aim to provide the first step towards a common understanding of what constitutes *gross negligence* in online banking for consumers in Germany and the UK from a consumer perspective.

We assume that *gross negligence* implies that the corresponding person (should have) had the knowledge and ability required to avoid an objectively severe and subjectively inexcusable breach of duty. Then it is important to understand what could be considered common knowledge about the expected standard of care for security in online banking. Any requirements on people's behaviour beyond such common knowledge would require banks to prove that an effective intervention had happened since they have the burden of proof for gross negligence. Our methodology allows us to conclude that we have likely revealed any belief held by at least 10 per cent of the population from which we sampled [166]. In section 4.5.5 we summarised what our participants had mentioned they would or could do to improve the protection of their online banking account. This list is, thus, likely a superset of computer security practises in online banking for which the omission could be considered gross negligence.

Future work would be required to narrow this collection of behaviours down to those held by a sufficient majority of online banking users, and derive from it a set of actionable, goal-referenced, and "easy to read lists of 'dos and don'ts' for customers" (as requested by the UK Parliament [207]). A similar approach to our research could also be used in future work to create similar lists for other relevant areas of financial services, e.g. card usage at Points of Sale (PoS) and Automated Teller Machines (ATM). We want to stress the importance of items on such lists being actionable, feasible, and goal-referenced as key requirements on effective behaviour-change interventions [209, 210].

4.7 Summary

We studied how online banking users conduct transactions, using their accounts and money. We gathered rich data through a multi-method study design on their use of transaction authentication, and their understanding of these technologies and risks of fraud in online banking. Our findings confirm that some people fail to secure their online banking transactions and we uncovered a complex interplay of factors surrounding this phenomenon. Finally, we discussed and recommended measures that banks, regulators, and legislators could pursue to improve online banking security.

Chapter 5

Of Personas and a Novel Mechanism

5.1 Introduction

Drawing on the exploratory and foundational nature of the study presented in chapter 4, this chapter is a diversion from our investigation of state-of-the-art mechanisms in support of reliable data input. Instead, in this chapter, we aim to collate the newly gained knowledge about transaction authentication and derive further insights relevant to address some of the identified issues for reliable data input. To this end, we present two analytical studies in this chapter.

The first study presented in this chapter is a statistical analysis of the data gathered on people's behaviour and perceptions while doing online banking transactions, as presented in section 4.5 of the previous chapter. Based on the statistical analysis, we then derive and design three personas as representations of the studied user population. These personas offer an alternative perspective on the user study data and, thereby, complement it in a significant and meaningful way.

In the second study of this chapter, we propose a novel transaction authentication mechanism to address specific issues discovered during the online baking user study in the previous chapter. We present a low-fidelity prototype of this mechanism, analyse its security properties against financial malware, and re-use the previously developed personas to evaluate its usability.

By doing so, we present an application of the transaction authentication personas in the evaluation of low-fidelity prototypes: First, the thematic analysis of the raw user study data unveiled specific usability issues that translated into security issues, as presented and discussed in chapter 4. We then proposed a solution for these security issues and, finally, utilised our personas to evaluate this mechanism and confirmed that it has an overall positive impact on the usability in addition to the security. This usability evaluation would not have been possible by looking at the user study data from the same perspective we used to identify the usability and security issues, which showcases the usefulness of personas as a complement to such data.

Our main contributions in this chapter are as follows:

1. We derive three representative personas of transaction authentication users

based on a statistical analysis of between-subject similarities. The description of each persona is grounded in the qualitative data of the participants this persona represents.

2. We propose a novel transaction authentication mechanism, motivated by findings in our empirical data, and present a theory-driven usability evaluation and security analysis.
3. We showcase an application of personas in the evaluation of low-fidelity prototypes by providing an alternative representation user study data.

5.2 Related Work

In this section we present relevant related work on the creation of personas from qualitative data. Personas are fictitious characters that are used to represent users in terms of their goals, behaviour, and personal characteristics. They can be used to communicate specific user issues and corresponding requirements within and outside the design team, act as stand-ins for users in (the phases of) the design process when real users are not easily available, and–if grounded in empirical user data–help to connect designers to a more tangible representation of such data [211]. Jones *et al.* [212] give particular emphasis to their use as guidance when exploring the design space of complex systems and problems.

There is not a single persona elicitation method, but a plethora of variants and adaptions exist, each with their strengths and limitations [213, 214]. One commonly recommended method is to create personas with a strong foundation in, and rigorously derived from, empirical data [215, 216, 211], rather than based on expert knowledge and assumptions. Matthews *et al.* [217] found that personas better communicate critical design constraints if they clearly distinguish between content based on empirical findings and supplemental data included to make the results more tangible and facilitate the communication.

Faily and Flechais [218] describe a methodology to directly connect personas with the coded user data from which they were created, which provides them with more credibility: In the first phase, researchers elicit propositions related to the most salient themes discovered in the underlying empirical data. In the next phase, researchers create succinct descriptions of potential persona characteristics based on these propositions and note their confidence in the strength of this relationship. Lastly, supporting narratives consistent with the elicited persona characteristics are written. In cases where this is not possible, the researchers need to evaluate whether some of the characteristics are not relevant for the context of analysis, or multiple personas are needed to reflect this variation in persona characteristics. Dotan *et al.* [219] argued about the importance to balance for the researchers' inherent subjectiveness when coding qualitative data by incorporating multiple data sources into the persona generation.

Chapman and Milham [220] argue that personas can be difficult to verify despite being rooted in empirical data without yielding an overwhelming amount of distinct entities, diminishing their usefulness. Sinha [221] and Tu *et al.* [222] reduced the number of potential personas by applying Principal Component Analysis (PCA) and Hierarchical Agglomerative Cluster analysis (HAC) to find natural groupings in the data and derived one persona from the data for each grouping.

5.3 Persona Creation

In this section, we report on our creation of three personas from the empirical data collected in chapter 4. The goal of these personas is to provide a different perspective on the data gathered during the user study through a more tangible representation. As such, we aim at minimising subjectiveness in the persona generation to provide an as accurate as possible perspective on that data. We emphasise that, as all persona generation techniques simplify the representation of data and are essentially data reduction techniques, our personas are best used in combination with (knowledge of) the study findings presented in section 4.5. This is an inherent characteristic in the concept of personas and not specific to our work.

5.3.1 Methodology

We use an approach similar to Sinha [221] and Tu *et al.* [222] in preparing our study data for the persona creation: By using statistical methods to identify clusters in a low-dimensional representation of the study data, we group participants based on similarities in their characteristics. Afterwards, we derive personas for each cluster from the data of participants within that cluster.

We begin by reducing the dimensionality of the highly correlated representation of the study data through Multiple Correspondence Analysis (MCA) with principal normalisation. This method quantifies nominal data by assigning numerical values such that the distance between the objects' representation in a low-dimensional Euclidean space corresponds to the similarity of these objects in the original, high-dimensional data. It is an effective method for data reduction during the analysis of high-dimensional data while retaining the underlying structure that represents key characteristics of that data.

Subsequently, we use cluster analysis to remove subjectivity from the classification process. Cluster analysis identifies structure within the data expressed through objective similarities between data subjects and creates natural groupings that describe the data space [223]. This method is considered a standard method to improve understanding of customer data [224] and is likely to produce homogeneous subject profiles when used on a low-dimensional representation of the data [225], as done here.

Specifically, we use the object scores from the previously calculated MCA for Hierarchical Agglomerative Cluster analysis (HAC) with the Unweighted Pair Group Method of Arithmetic means (UPGMA; between-group average linkage), measured by squared Euclidean distance. This method initially allocates individual groups to each object (participant), before iteratively merging groups with minimal squared Euclidean distance between their members (i.e. minimal proximity between the geometric centroids of each group). HAC has advantages over non-hierarchical clustering in handling noise and outliers in the data [226]. The UPGMA method is likely to create clusters of similar size, which is favourable when creating personas who shall be indicative for several participants.

The clustering phase is followed by the creation of one persona representative for each cluster from the qualitative study findings in section 4.5.4. We create personas close to the centroid of the characteristics that define each cluster. This removes subjectivity from the persona creation since the cluster analysis defined objective clusters and supports the creation personas that are rooted in the data. As previously described in section 5.2, creating personas with a strong foundation in, and rigorously derived from, empirical data is one of the most common recommendations in the literature on persona creation [215, 216, 211].

As previously described in section 5.2, there is no single persona generation method. Instead, a plethora of variants and adaptions support tailoring the creation of personas to the individuality of each use case. Our persona creation process was inspired by a method proposed by Faily and Flechais [218], who placed particular emphasis on ensuring that their personas are derived from and rooted in empirical data.

This method builds upon a grounded theory methodology used in preceding data analysis. Given the output of this data analysis, their approach begins by deriving propositions from the most grounded concepts. These propositions are used to justify persona characteristic in line with the relationships of the grounded theory model. Next, the characteristics are augmented with subjective ratings of the analyst's confidence them and then aggregated into clusters through affinity diagramming. Finally, descriptions of the aggregated persona characteristics are written, whereby characteristics not relevant to the context of analysis or covering poorly grounded themes may be omitted.

Our data analysis followed the thematic analysis methodology, while Faily and Flechais [218] built upon the grounded theory methodology. Accordingly, while we are inspired by their persona generation approach, we take the following adaptations to accommodate the difference in the data gathered through those methodologies:

Analogue to the concepts and relationships used in [218], our method utilises the nominal variables previously used as input to the MCA as propositions to represent the themes developed during the thematic analysis in section 4.5.4. We use the objective clusters created by the HAC method instead of subjective affinity diagramming techniques to allocate the data of each participant to the creation of one persona. Candidate characteristics for each persona are then calculated from the nominal variables of all participants in the corresponding cluster as a majority vote of the individual variable values. Similar to Faily and Flechais, we exclude from the persona design those characteristics not relevant or poorly grounded, in our case defined as characteristics that are identical in the majority vote across all clusters. Finally, each persona is enriched with the average values of demographic data and Likert scale questions collected from the corresponding participants.

5.3.2 Cluster analysis

We prepared the study data for Multiple Correspondence Analysis by deriving 16 nominal variables from the qualitative findings. Each sub-theme discovered in the thematic analysis (as reported in section 4.5.4) yielded one variable, whereby binary values were used to denote whether individual participants contributed to the themes. The sub-theme of *Insecure behaviour* was combined with our observations during the demonstration of whether people verified the transaction context data reported in section 4.5.2. Finally, we incorporated three variables for participants' threat model perceptions, as summarised in table 4.3. Overall, three variables were defined with ternary values: Technology Used (SMS, app, authenticator), Correct Transaction Authentication (secure, insecure, N/A), Reaction to Unexpected (repeat, abort, N/A).

MCA with equally weighted ranked variables and variable principle normalisation was calculated in SPSS 26. The SPSS syntax is given in listing 5.1. The first three dimensions represented variance of 25.967%, 17.993%, and 14.786% with Cronbach's Alpha of α=.810, α=.696, and α=.616, respectively. Although the commonly accepted rule of thumb is to require Cronbach's Alpha values of at least .7, this requirement should be relaxed for small sample sizes of around 20 or fewer data objects [227] and for exploratory research [228]. Thus, the first two dimensions were kept as reliable representations of the data, while the third was discarded as not sufficiently reliable. The resulting, two-dimensional model accounts for 43.96% of the variance. Table 5.1 lists the discrimination measures for both dimensions.

Listing 5.1: SPSS syntax for MCA calculations (adjusted for line breaks).

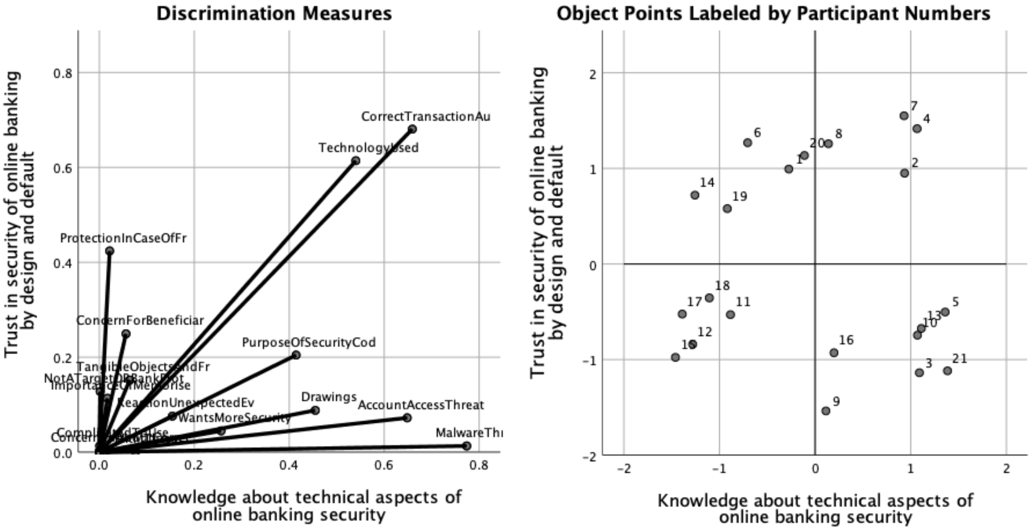
```
MULTIPLE CORRES VARIABLES=CorrectTransactionAuthentication
    TechnologyUsed MalwareThreat MitMThreat AccountAccessThreat
    ConcernedAboutCorrectTransactionData NotATargetORBankProtectsMe
      PurposeOfSecurityCodes ProtectionInCaseOfFraud Drawings
    TangibleObjectsAndFrictionSignOfSecurity WantsMoreSecurity
    ConcernForBeneficiaryPrivacy ComplicatedToUse
    ImportanceOfMemorisedSequenceOfActions ReactionUnexpectedEvent
     /ANALYSIS=CorrectTransactionAuthentication(WEIGHT=1)
        TechnologyUsed(WEIGHT=1) MalwareThreat(WEIGHT=1) MitMThreat
        (WEIGHT=1) AccountAccessThreat(WEIGHT=1)
        ConcernedAboutCorrectTransactionData(WEIGHT=1)
        NotATargetORBankProtectsMe(WEIGHT=1) PurposeOfSecurityCodes
        (WEIGHT=1) ProtectionInCaseOfFraud(WEIGHT=1) Drawings(
        WEIGHT=1) TangibleObjectsAndFrictionSignOfSecurity(WEIGHT
        =1) WantsMoreSecurity(WEIGHT=1)
        ConcernForBeneficiaryPrivacy(WEIGHT=1) ComplicatedToUse(
        WEIGHT=1) ImportanceOfMemorisedSequenceOfActions(WEIGHT=1)
        ReactionUnexpectedEvent(WEIGHT=1)
     /DISCRETIZATION=CorrectTransactionAuthentication(RANKING)
        TechnologyUsed(RANKING) MalwareThreat(RANKING) MitMThreat(
        RANKING) AccountAccessThreat(RANKING)
        ConcernedAboutCorrectTransactionData(RANKING)
        NotATargetORBankProtectsMe(RANKING) PurposeOfSecurityCodes(
        RANKING) ProtectionInCaseOfFraud(RANKING) Drawings(RANKING)
          TangibleObjectsAndFrictionSignOfSecurity(RANKING)
        WantsMoreSecurity(RANKING) ConcernForBeneficiaryPrivacy(
        RANKING) ComplicatedToUse(RANKING)
        ImportanceOfMemorisedSequenceOfActions(RANKING)
        ReactionUnexpectedEvent(RANKING)
     /MISSING=CorrectTransactionAuthentication(LISTWISE)
        TechnologyUsed(LISTWISE) MalwareThreat(LISTWISE) MitMThreat
        (LISTWISE) AccountAccessThreat(LISTWISE)
        ConcernedAboutCorrectTransactionData(LISTWISE)
        NotATargetORBankProtectsMe(LISTWISE) PurposeOfSecurityCodes
        (LISTWISE) ProtectionInCaseOfFraud(LISTWISE) Drawings(
        LISTWISE) TangibleObjectsAndFrictionSignOfSecurity(LISTWISE
        ) WantsMoreSecurity(LISTWISE) ConcernForBeneficiaryPrivacy(
        LISTWISE) ComplicatedToUse(LISTWISE)
        ImportanceOfMemorisedSequenceOfActions(LISTWISE)
        ReactionUnexpectedEvent(LISTWISE)
/DIMENSION=2
/NORMALIZATION=VPRINCIPAL
/MAXITER=100
/CRITITER=.00001
/PRINT=DISCRIM
/PLOT=OBJECT(20) DISCRIM(20).
```

	Dimension	
	1	2
Correct Transaction Authentication	.659	.681
Technology Used	.540	.614
Malware Threat	.771	.013
Man-in-the-Middle Threat	.076	.001
Illegitimate Login Threat	.648	.072
Concerned About Correct Transaction Data	.017	.005
Not A Target OR Bank Protects Customer	.002	.128
Purpose of Security Codes	.414	.205
Protection in Case of Fraud	.022	.424
Type of Drawing	.455	.088
Trust Through Tangible Object OR Friction	.064	.152
Would Like to Have More Security	.256	.045
Concerned for Beneficiary's Privacy	.056	.249
Online Banking is Complicated	.001	.013
Importance of Memorised Sequence of Actions	.017	.114
Reaction to Unexpected Event	.154	.076

Table 5.1: Discrimination measures for two dimensions to define low-dimensional representation of participants' characteristics.

Considering the individual contributions of nominal variables to both dimensions allows for a better understanding of the meaning of the resulting two-dimensional space. Two variables, denoting the technology which participants used and whether they securely handled codes for transaction authentication, contribute to both dimensions to a similar, large degree. Thus, these variables are inadequate to understand the difference between these dimensions. Considering the other 14 variables, it becomes apparent that the top three contributors to each dimension weight more than the remaining 11 variables. For Dimension 1 these are *Malware Threat*, *Illegitimate Login Threat*, and *Type of Drawing*. For Dimension 2 these are *Protection in Case of Fraud*, *Concerned for Beneficiary's Privacy*, and *Purpose of Security Codes*. Considering these factors we label Dimension 1 as *Knowledge about technical aspects of online banking security* and Dimension 2 as *Trust in security of online banking by design and default*.

Figure 5.1a provides a graphical representation of the discrimination measures calculated through MCA and fig. 5.1b depicts the placement of the 21 study participants in the corresponding two-dimensional space.



(a) Graphical representation of discrimination measures for two dimensions to define a low-dimensional representation of participants' characteristics.

(b) Placement of study participants in two dimensional space defined by measures given in table 5.1.

Figure 5.1: Graphical representations of the output of the MCA.

Listing 5.2: SPSS syntax for HAC calculations (adjusted for line breaks). Variables are output of MCA with TRAX_1_1 and TRAX_2_1 referring to input variable X in MCA code (see listing 5.1).

```
CLUSTER   TRA1_1_1  TRA1_2_1  TRA2_1_1  TRA2_2_1  TRA3_1_1  TRA3_2_1
    TRA4_1_1  TRA4_2_1  TRA5_1_1  TRA5_2_1  TRA6_1_1  TRA6_2_1  TRA7_1_1
    TRA7_2_1  TRA8_1_1  TRA8_2_1  TRA9_1_1  TRA9_2_1  TRA10_1_1
    TRA10_2_1  TRA11_1_1  TRA11_2_1  TRA12_1_1  TRA12_2_1  TRA13_1_1
    TRA13_2_1  TRA14_1_1  TRA14_2_1  TRA15_1_1  TRA15_2_1  TRA16_1_1
    TRA16_2_1
  /METHOD BAVERAGE
  /MEASURE=SEUCLID
  /PRINT SCHEDULE
  /PLOT DENDROGRAM.
```

Next, we used to object scores of the MCA for Hierarchical Agglomerative Clustering with UPGMA settings, as described in section 5.3.1. The calculations were done in SPSS 26, using the syntax in listing 5.2, and generated the dendrogram shown in fig. 5.2. The dendrogram illustrates the arrangement of clusters and suggests a separation of study participants into three clusters.

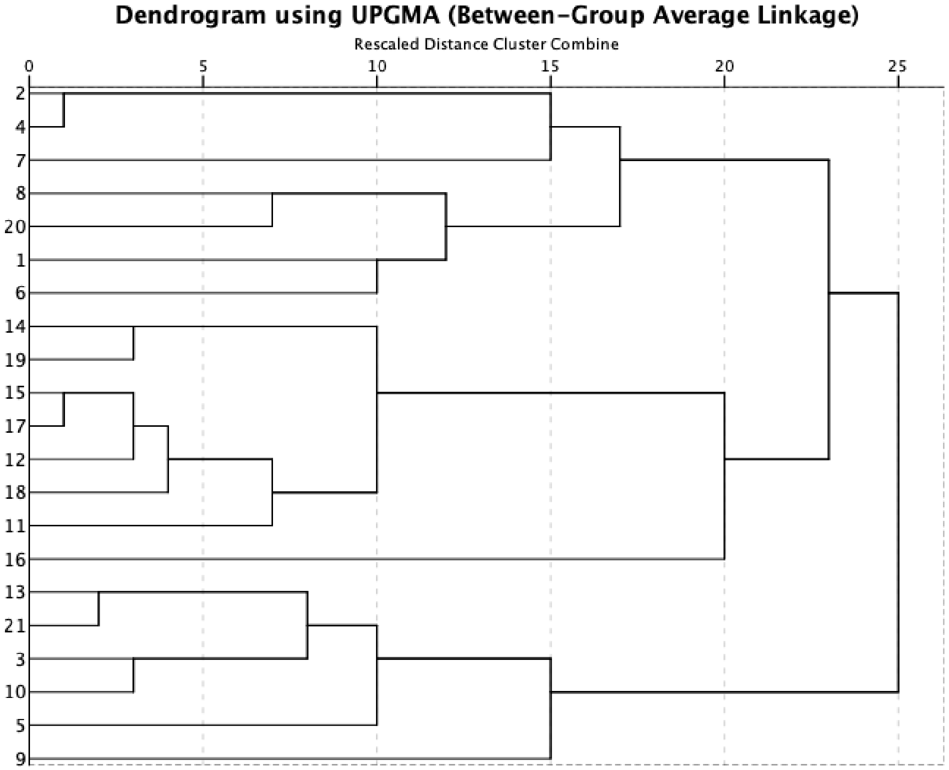


Figure 5.2: Dendrogram showing the hierarchical relationship between participants based on HAC analysis with UPGMA method and between-group average linkage, measured by squared Euclidean distance in two-dimensional space described through discrimination measures from table 5.1.

5.3.3 Persona design

Following the methodology outlined in section 5.3.1, three personas were created based on the output of the HAC analysis (see dendrogram in fig. 5.2). We calculated candidate characteristics for each persona according to the majority vote of the corresponding participants' variable values, i.e. a persona is allocated a candidate characteristic if the characteristic is also represented in the data of the majority of participants upon who this persona is built. Table 5.2 lists the individual characteristics from which the majority vote was calculated. Four variables were found to have identical majority vote results for all three clusters: *Man-in-the-Middle Threat*, *Concerned About Correct Transaction Data*, *Concerned for Beneficiary's Privacy*, *Reaction to Unexpected Event*. Following the persona creation methodology, we excluded these variables from the persona design as they would not contribute to the distinction between the personas. One of three majority votes for the theme *Com-*

plicated to Use was inconclusive (three *yes* and three *no* counts) and subsequently interpreted as *medium* for the persona design. The results of the majority votes for *Malware Threat* and *Illegitimate Login Threat* were mutually exclusive for all three clusters, i.e. for each persona one variable applied and the other did not, and were thus unified into a joint characteristic.

The persona skeletons are defined by the resulting set of characteristics for each persona and were filled with brief descriptive labels and suitable graphics for each characteristic. They were further enriched with data from the participants' average age, gender, years of experience with online banking, number of online banking transactions per month, SeBIS score, and answers to the Likert-style questions in the questionnaire (see table 4.2 and figs. 4.1 and 4.2). To take further advantage of the qualitative nature of the underlying data, we created descriptive texts for the variable values of each persona and augmented them with anecdotal context information given by corresponding participants during the interviews. Finally, the personas were augmented with computer-generated profile pictures[1] and given imaginary names, professions, and brief personality descriptions. The resulting personas are shown on pages 133 to 135.

[1] Generated via the generative adversarial network StyleGAN2 (version of Dec 2019), available at `https://www.thispersondoesnotexist.com/`.

	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	c15	c16
P3	secure	authenticator	yes	no	no	yes	yes	transaction	no	UI	yes	yes	no	no	no	repeat
P5	secure	authenticator	yes	no	no	no	no	user	yes	backend	yes	no	no	yes	no	abort
P9	secure	authenticator	no	yes	yes	yes	no	transaction	no	UI	yes	no	no	yes	no	–
P10	secure	authenticator	yes	no	no	yes	no	transaction	yes	UI	no	yes	no	yes	no	repeat
P13	secure	authenticator	yes	no	no	no	yes	transaction	no	backend	no	no	no	no	yes	–
P21	secure	authenticator	yes	yes	no	yes	no	transaction	no	backend	yes	no	no	no	yes	repeat
P11	insecure	app	no	no	yes	yes	no	user	no	UI	yes	no	no	yes	no	–
P12	N/A	app	no	no	yes	yes	yes	–	no	UI	yes	no	no	yes	yes	–
P14	N/A	app	no	no	yes	no	yes	user	yes	UI	no	no	yes	no	no	repeat
P15	N/A	app	no	no	yes	yes	no	–	no	UI	no	no	no	no	no	–
P16	N/A	app	yes	yes	no	yes	yes	–	no	backend	yes	yes	no	no	no	–
P17	N/A	app	no	no	yes	yes	no	–	yes	UI	no	no	no	no	no	–
P18	N/A	app	no	yes	yes	no	yes	user	no	UI	no	no	no	no	no	–
P19	N/A	authenticator	no	no	yes	yes	yes	user	yes	UI	no	no	yes	yes	no	repeat
P1	insecure	SMS	no	no	yes	yes	yes	–	yes	UI	yes	yes	no	yes	yes	repeat
P2	insecure	SMS	yes	yes	no	yes	no	user	yes	backend	no	yes	no	no	no	–
P4	insecure	SMS	yes	yes	no	no	yes	user	yes	backend	no	yes	no	yes	yes	–
P6	insecure	SMS	no	no	yes	yes	no	user	no	UI	no	no	yes	no	yes	repeat
P7	insecure	SMS	yes	no	yes	yes	yes	transaction	yes	backend	yes	yes	yes	no	no	abort
P8	insecure	SMS	no	yes	yes	yes	yes	transaction	yes	backend	no	no	no	no	yes	–
P20	insecure	authenticator	no	no	yes	yes	yes	user	yes	backend	no	no	no	no	yes	repeat

Table 5.2: Table of participants' characteristics derived from qualitative data and grouped according to clusters affiliations calculated in section 5.3.2 (see dendrogram in fig. 5.2). Column labels *c1* to *c16* denote characteristics listed in table 5.3. *N/A* denotes non-applicability of participants who did not interact with security codes during the demonstration task. – (long dash) denotes participants who did not make statements to indicate their position on the corresponding theme. Three majority votes were calculated for each characteristic according to the participant groupings, whereby – cases were ignored. The majority vote of c14 in the upper grouping (i.e. P3, P5, P9, P10, P13, and P21) was inconclusive (three *yes* and three *no*) and subsequently as *medium* interpreted. Highlighted columns denote characteristics which were excluded from the persona design (in line with our persona creation method described in section 5.3.1) due to identical results of the majority vote in all cluster.

c1	Correct Transaction Authentication
c2	Technology Used
c3	Malware Threat
c4	MitM Threat
c5	Illegitimate Login
c6	Concerned About Correct Transaction Data
c7	Not A Target OR Bank Protects Me
c8	Purpose Of Security Codes
c9	Protection In Case Of Fraud
c10	Drawings
c11	Tangible Objects And Friction Signs Of Security
c12	Wants More Security
c13	Concern For Beneficiary Privacy
c14	Complicated To Use
c15	Importance Of Memorised Sequence Of Actions
c16	Reaction Unexpected Event

Table 5.3: List of characteristics and their corresponding labels c1 to c16, as used in table 5.2. Highlighted rows denote characteristics which were excluded from the persona design (in line with our persona creation method described in section 5.3.1) due to identical results of the majority vote in all three cluster.

Finally, the different elements of each persona were arranged such that they fit on a single page. The order of elements from top to bottom is as follows: The name, picture, and first paragraph of each persona are entirely imaginary (except for their age). The following two paragraphs are subjective narratives to describe the persona demographics and characteristics. The graphics and graphic labels in the middle section of each persona represent factual persona characteristics derived from and grounded in qualitative data. We chose graphical representations to bring each persona's characteristics into the focal point and enhance their memorability, as this section is arguably the most important and reliable data for each persona. Lastly, score meters provide additional, supplementary data at the bottom of each persona.

Leah Hardware

Leah is a 33-year-old regional sales manager at an office supplies store who enjoys jigsaw puzzles, watching television, and reading novels. She is intelligent, careful, and likes planning.

Leah started doing online banking 6.5 years ago. She uses a laptop and authenticator to do on average 5 transactions per month.

She is aware that malware on her laptop would be a risk to her online banking and that she needs to be proactive to secure her account. Having a tangible, offline device involved makes her feel more secure when sending money although it makes online banking a little bit more complicated. She always checks the transaction details with her authenticator to prevent typos and fraudulent manipulation at the same time. She understands that quoting the security code means that she confirms that those details match with her intentions. Leah is aware of how her devices communicate with the bank's servers but usually focuses mostly on the UI. She assumes that her bank would want to avoid reimbursing her in case of fraud.

Transaction context data securely verified

Laptop and authenticator

Concerned about malware

Fraud is realistic risk

Security codes authenticate transactions

No reimbursement in case of fraud expected

UI-focused and backend-aware

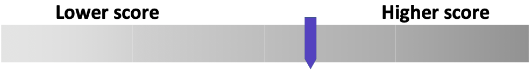
Trust in security linked to tangible object

Current level of security satisfactory

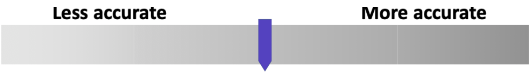
Somewhat complicated to use

No automatism of online banking

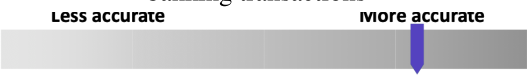
SeBIS score



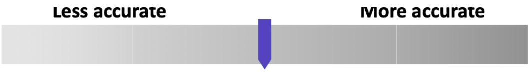
"I value personal interaction with bank employees"



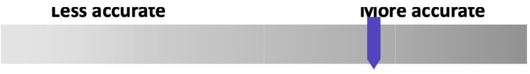
"I follow the website's instructions during online banking transactions"



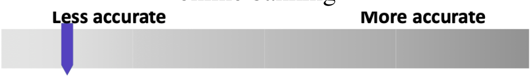
"I worry about security during online banking transactions"



"I trust my bank to secure my online banking transactions"



"I prefer to physically go to my bank instead of online banking"



Shelly Mobile

Shelly is a 33-year-old local activist who enjoys eating out, going to the movies, and blogging. She is intelligent, creative, and spontaneous. Due to her work, she spends a lot of time at social events and travelling.

Shelly started doing mobile banking 4 years ago. She uses her mobile phone to do on average 5 transactions per month. Previously, Shelly did online banking on her laptop with security codes via her mobile phone and authenticator for one year but switched to mobile-only because it requires one device less. She still has an authenticator at home but is not sure she remembers how it works.

Shelly is wary of the risks if someone else could log in to her app. Thus, she makes sure nobody watches her when she logs in and always logs out when she is finished. Yet, she assumes that criminals would likely target people who are more wealthy than her and is, therefore, more focused on preventing typos when she pays someone. In either case, she would not expect her bank to reimburse her if she lost money. For Shelly, the online banking experience begins and ends with the app on her phone. Security codes, as she remembers from her past, were required to prove her identity to the bank.

Transaction verification does not apply

Mobile banking

Concerned about illegitimate login

Fraud is not a realistic risk

Security codes authenticate the user

No reimbursement in case of fraud expected

UI-focused

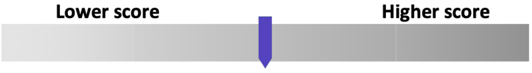
No specific trust anchor

Current level of security satisfactory

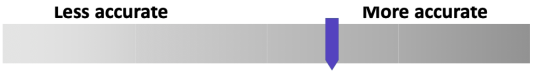
Not particularly complicated to use

No automatism of online banking

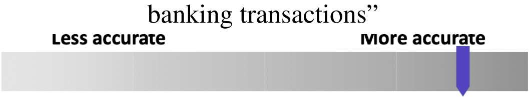
SeBIS score



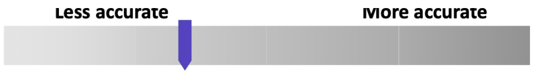
"I value personal interaction with bank employees"



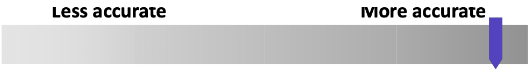
"I follow the website's instructions during online banking transactions"



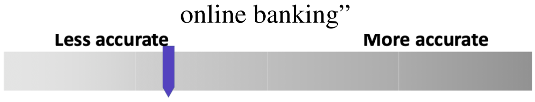
"I worry about security during online banking transactions"



"I trust my bank to secure my online banking transactions"



"I prefer to physically go to my bank instead of online banking"



Jun Message

Jun is a 26-year-old finance and accounting manager at his family's restaurant chain, who enjoys working on his car, travelling, and playing hockey. He is charismatic, caring, and creative.

Jun started doing online banking 6 years ago. He uses a laptop and receives security codes via SMS on his mobile phone. On average, Jun does 6 online banking transactions per month.

Securing access to his online banking account is most important for Jun. Security codes, as he understands, are used for Two Factor Authentication to proof his identity when he makes a transaction. He is wary that the SMS is delivered over a separate communication channel and would not access his bank account from the same mobile phone. Jun memorised in detail how to make a transaction, minimising the cognitive load through automation. This includes utilising that his mobile phone highlights the security codes in his SMS, enabling him to copy the code without paying much attention to the other content. Jun has high trust in his bank's online security but would not mind if they introduced additional security measures as long as he can integrate them into his routine. Should he still become a victim of fraud, he is certain the bank will be able to get the money back from the criminal's account.

Transaction context data not verified

Laptop and text message

Concerned about illegitimate login

Fraud is not a realistic risk

Security codes authenticate the user

Reimbursement in case of fraud expected

Backend-aware

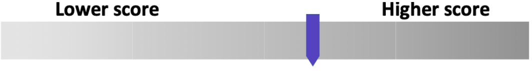
No specific trust anchor

Would appreciate more security

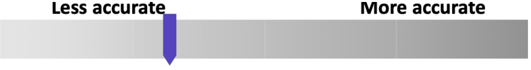
Not particularly complicated to use

Online banking becomes automatism

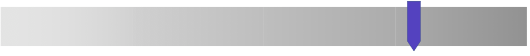
SeBIS score



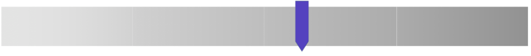
"I value personal interaction with bank employees"



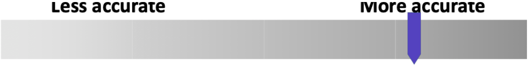
"I follow the website's instructions during online banking transactions"



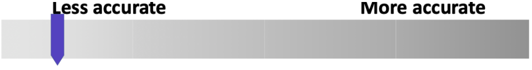
"I worry about security during online banking transactions"



"I trust my bank to secure my online banking transactions"



"I prefer to physically go to my bank instead of online banking"



5.4 Novel Transaction Authentication Mechanism

In this section, we describe the design of a novel transaction authentication mechanism and demonstrate how our transaction authentication personas can be utilised to evaluate the usability and security of transaction authentication mechanisms. First, in section 5.4.1, we examine specific findings from our user study as motivation for the novel mechanism and as evidence for its expected effectiveness. Next, we present the novel mechanism in section 5.4.2 based on a description of how it could operate on top of a Visual Cryptogram mechanism (see section 2.4.2). Subsequently, we conduct a security analysis of this mechanism against financial malware threats in section 5.4.3 and evaluate its usable security properties through the transaction authentication personas in section 5.4.4. Finally, we discuss variations of the novel mechanism through an analysis of how it would interact with mechanisms other than Visual Cryptograms and its unique applicability to authenticate bulk transactions in section 5.4.5.

5.4.1 Motivation

The main motivation behind our novel mechanism is based on our findings of online banking users' *feelings of personal responsibility* for the correctness of transaction data, as described in section 4.5.4.1: Most of the participants in this study expressed concerns about typos in their transaction data. As a consequence, some users focused more of their attention on the correctness of the data visible on their primary device or app rather than on their trusted, secondary device. Yet, most transaction authentication technologies, such as Visual Cryptograms and those based on text messages or push notifications, require users to ensure that the transaction context data shown by their trusted, secondary device or app matches their intentions.

We know from several decades of research in psychology that attention capacity is limited [190]. A cognitive process called Central Executive [229] directs one's thoughts and is responsible for maintaining the focus on task goals. Research has shown that not only is the switching between two tasks cognitively demanding due to between-task competition but also causes prolonged distraction due to task-set inertia and can, thus, inhibit performance on the new task [191]. We predict that removing the need to switch between the detection of typos and the detection of malicious content manipulation will improve transaction security. In the absence of a method that relives users from their concerns about eventual typos, we focus the latter instead.

The goal for the novel transaction authentication mechanism is to ensure that checking a primary device for typos *at the right time* will essentially achieve concurrent checks for malicious content manipulation on the secondary device. Yet, the system shall also be usable by those who already verify the correctness of the transaction context data on the secondary device. Furthermore, no significant behaviour change shall be required from either user.

5.4.2 The basic idea

Recall from section 2.4.2 that users of Visual Cryptogram mechanisms shall scan the encrypted data with their trusted device and then shift their focus from the webpage to the screen on that device. Next, they shall verify that the transaction context data shown on the device matches their intention, e.g. by comparing it with a printout. If the user does not validate the transaction context data they would not be able to detect if that data had been manipulated. Finally, the user shall quote the corresponding TAN only if they had successfully verified that the transaction context data is correct.

The process for the novel mechanism is to automate the comparison of the data received by a trusted, secondary device and that shown to a user by the corresponding primary device or app. This would make it indifferent on which of those the user verifies that this data matches with their intentions. The novel mechanisms would operate as follows:

1. When a user enters transaction instructions on their primary device, the browser displays the corresponding Visual Cryptogram (i.e. encrypted transaction data as received by the bank) on the same page. After entering the instructions, the user operates their trusted device to scan the Visual Cryptogram and the input fields with the transaction data they had entered at the same time. The user shall either concurrently focus their attention on detect-

ing typos on either device or verify the correctness of the transaction data shown by their trusted, secondary device.

2. We envision that the trusted device uses optical character recognition (OCR) technology to digitise the visible transaction context data scanned from the primary device (e.g. webpage). We assume that this process will produce accurate results with negligible error rate. We believe this assumption is reasonable since the structure of that data (e.g. position, font, size, contrast, etc.) can be tailored to this purpose. Note that this assumption is critical for the security guarantees of the mechanism and will be discussed in detail in section 5.4.3.

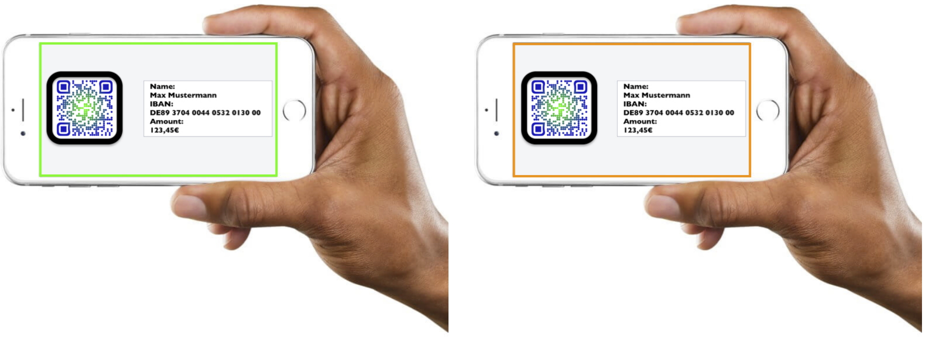
3. The device shall decrypt the encrypted data in the Visual Cryptogram scanned from the webpage and compare it with the unencrypted data scanned at the same time form the same webpage.

4. Finally, if those data do not match, the device would indicate this outcome to the user (e.g. an error or warning message). If the data does match, the device could display the corresponding TAN or provide an action (e.g. a button) on the second device for seamless verification of the transaction.

Figure 5.3 depicts a low-fidelity prototype as example visualisation of the novel mechanisms with a mobile phone and horizontal camera alignment. Implementations with an authenticator would be analogous.

5.4.3 Security analysis

In this section, we present a security analysis of the novel mechanism against financial malware. The attacker conducts at Man-in-the-Browser (MitB) attack, as described in section 2.2.2.4, with the goal to manipulate the destination or value of an authorised online banking transaction. The attacker is successful if the user authorises a transaction despite such manipulation.



(a) Encrypted data in Visual Cryptogram matches the human-readable data, verifying that the user sees the same data on their screen as received by the bank.

(b) Encrypted data in Visual Cryptogram does not match the human-readable data, indicating that the user does not see the same data on their screen as received by the bank.

Figure 5.3: A low-fidelity prototype of the novel mechanism to concurrently scan a Visual Cryptogram and human-readable transaction data from a webpage.

5.4.3.1 Resistance against adversarial samples

Here, we discuss an assumption that is critical for the security of the proposed transaction authentication mechanism. We assume that the novel mechanism will not recognise characters different than what the user would recognise when capturing and analysing a video or a set of pictures taken of the screen of the user's primary device.

Different OCR models. Recognising characters from an image is a task called optical character recognition (OCR) and related technologies can be distinguished between machine-printed character recognition and handwriting recognition. Character recognition models for machine-printed characters commonly localise individual characters in an image by breaking it down into subsections and then classifying each character individually. They can be further divided into single-font and multi-font classifiers, whereby the former refers to algorithms optimised for a particular typeface to achieve far higher accuracy. When using a font that is particularly optimised for the OCR task, recognition rates of 99.99% (for genuine samples) are commonly achieved [230].

More advanced machine learning models such as Deep Neural Networks are commonly used to avoid the segmentation of text – which can be more error-prone for some tasks than others, in particular for handwriting recognition [231] – and instead classify entire sequences of characters at once.

Adversarial samples. The misclassification of characters in textual input can be an adversarial goal. Szegedy *et al.* [232] first explored this topic and found that classifiers based on deep neural networks, which have shown strengths in classifying a wide variety of natural occurrences, can be exploited with carefully crafted input signals to achieve misclassification. Paperknot *et al.* [6] formalised the notion of attacks against deep neural networks and showed that adversarial samples can be produced reliably. Figure 5.4 depicts a reproduction of Figure 10 in [6], showing cases of misclassification for handwritten digits. Song and Shmatikov [231] showed similar results for systems based on deep learning to classify printed text. They explained that their adversarial samples do not transfer to traditional OCR models without deep learning or those that significantly pre-process and simplify input signals such as interpreting each input pixel as binary, e.g. black and white.

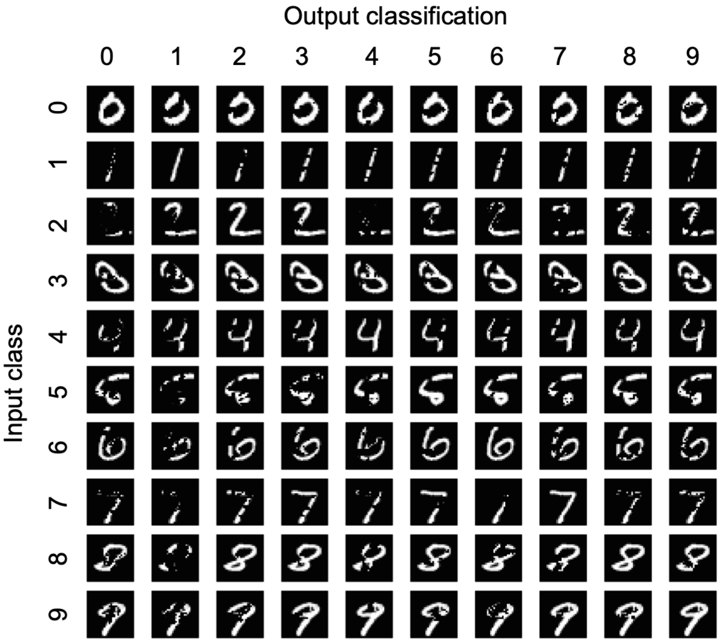


Figure 5.4: Adversarial samples against a deep neural network classifier for the classification of handwritten digits, reproduced from [6, Fig. 10].

In general, such misclassification attacks on optical character recognition exploited inaccurate decision boundaries in the classifier models by adding artefacts into pictures of text that most humans would reasonably ignore as minor disturbances (if they are perceived at all) and removing parts of individuals characters that most humans would intuitively repopulate. In both cases, these attacks effectively also exploit most humans' superior ability (compared to computers) to recognise and interpret structure expressed through the principles of Gestalt psychology [233], e.g. closure, proximity, continuity, and similarity.

OCR in the novel mechanism. Existing literature on adversarial samples is almost exclusively on Support Vector Machines and Deep/Convolutional Neural Networks [234], which are not necessary for our application of machine-printed character recognition. Yet, the existence of adversarial samples *per se* exposes general weaknesses for all machine learning models since they are required to provide proper outputs for previously unseen input. Thus, similar threats might eventually extend to machine learning techniques relevant to our use case. As a consequence, we use the remainder of this section to discuss the robustness of our use case against adversarial samples.

Formal security proofs and impossibility results against the security threats posed by adversarial samples are currently not feasible due to the absence of a theoretical model underpinning their crafting process. Instead, we will discuss several characteristics of our specific use case which we argue provide additional security against the emergence of applicable adversarial samples.

Firstly, our use case does not require the classification of a wide variety of inputs. Clear-text transaction data in an image would be fixed-length and fixed-pattern. Fixed-location, fixed-font, and fixed-background are trivial extensions. In particular, fonts optimised for OCR applications are likely to be beneficial, e.g. the widely used OCR-A/OCR-B fonts [235, 236] or the E13B/CMC7 fonts [237, 238] used by the banking industry on checks. These aspects significantly reduce the available search space for the generation of adversarial samples.

Secondly, a conservative approach to character recognition, i.e. rejection of low-confidence classifications, is possible and indeed preferable. For benign situations, we can assume that disturbances on the screen are likely caused by issues such as dust or screen glare. In most situations, users should be able to correct such issues on the spot. Thus, if the scanned transaction data does not provide a high confidence match with the encrypted data, the trusted device could prompt the user to clean their screen and the device's camera, avoid screen glare, and hold the camera sufficiently close to the screen of their primary device.

Thirdly, a technique to defend against adversarial samples called *feature squeezing* [239] is highly applicable to our use case. Feature squeezing entails pre-processing of input data to coalesce input variables and reduce feature complexity, such that adversarial perturbations disappear due to low sensitivity. Feature reduction through modification of colour depth, e.g. grey-scale or black-and-white, or spatial smoothing, e.g. local blurring, significantly reduce the search space available for the generation of adversarial samples. Meanwhile, the effects of colour depth reduction on benign samples are likely reduced through high-contrast backgrounds, while the monospaced fonts for OCR applications are less affected by spatial smoothing.

As a consequence, we assume that the application of OCR in the proposed novel transaction authentication mechanism is resistant against adversarial samples and will ensure high accuracy with negligible error rate.

5.4.3.2 Security against malware on the primary device

Here, we consider the following attack scenario: A MitB attack ensures that the bank receives transaction instructions that do not match with the user's intent. It also allows the attacker to arbitrarily manipulate the clear-text transaction data shown by the browser. We consider the following cases:

1. The user checks whether the transaction context data on their trusted, secondary device matches their intention. Under the assumption that this device has not been tampered with, the user would detect that this data does not match their intention and not authorise the transaction.

2. The user checks whether the transaction context data visible on their primary device matches their intention. We consider the following sub-cases:

(a) The attacker might manipulate the transaction context data visible on the primary device to differ from the instructions sent to the bank. Then the novel mechanism would detect a data mismatch with the Visual Cryptogram and not provide the user with means to authorise the transaction.

(b) The attacker might not manipulate the transaction context data visible on the primary device to differ from the instructions sent to the bank. Then the user would detect that this data does not match their intention and not authorise the transaction.

(c) The attacker might manipulate the transaction context data visible on the primary device to differ from the instructions sent to the bank while the user checks that data but not manipulate it while the user scans it with their trusted, secondary device. This would require the attacker to know or guess the timing for both events – an assumption we deem unlikely (and further discussed in the succeeding section 5.4.3.3).

Cases 1, 2(a), and 2(b) result in the detection of the attacker's manipulation of the transaction data, while case 2(c) is deemed unlikely. Thus, we conclude that the success chance of the attacker is negligible and the novel mechanism is secure against risks from financial malware on the primary device.

5.4.3.3 Security against malware on both user devices

Here, we take a closer look at the case in which the attacker has infected the user's secondary device – in addition to the primary device – with malware and knows that both devices belong to the same user. Recall from section 2.2.2.1 that risks from the execution of untrusted code such as malware mostly affect multi-purpose IT devices, e.g. mobile phones. As a consequence, we assume that the single purpose, trusted hardware authenticators are immune to the risks described in this section.

Recall furthermore from section 2.3.2 that Trusted Execution Environments are a sandboxing approach to create a separate execution environment, dedicated to the secure processing of sensitive tasks in isolation from the main execution environment of a system. Recall also that TEEs cannot guarantee a secure and bug-free system [62]. Yet, TEEs are a commonly used and effective security measure for banking applications on mobile phones and are frequently supplemented with additional measures such as run-time application self-protection, application cloning countermeasures, and mechanisms to detected if the device has been rooted.

We consider the following cases, assuming that the attacker has infected the user's primary device and mobile phone, and–crucially– established that these devices belong to the same user:

1. In this scenario, the attacker has successfully circumvented the mobile phone's security measures to take control of the banking application. In this case, the attacker could initiate and verify an arbitrary transaction without the users being able to detect it, as neither device necessarily displays to the user the manipulated instructions received by the bank.

2. In this scenario, the attacker has not circumvented the mobile phone's security measures to take control over the banking application but established a side-channel attack to deduce information about the state of the banking application. In this case, the attacker could try to utilise such information to improve its timing during attack 2(c) described in section 5.4.3.2. For example, if the attacker could determine when the camera on the mobile phone is activated (to scan the transaction data from the primary device), it could display the tampered transaction data on the primary device while the camera is active but display the transaction data intended by the user while the camera is not active. The user could still detect the attack if they were to verify the correctness of the transaction data on either device while they are scanning it with their mobile phone.

3. In this scenario, the attacker has neither circumvented the mobile phone's security measures nor established a side-channel attack that would leak relevant information about the state of the banking apps on the mobile phone. In this case, the attacker does not gain any advantage from having malware on the secondary, trusted device and we refer back to the previous section 5.4.3.2.

In summary, the attacker would be required to exploit one or more major vulnerabilities on the mobile phone, e.g. a privilege escalation on the mobile phone's OS, to succeed with the attack. Besides, it would need to have established which infected mobile phones and which infected primary devices (e.g. laptops and desktop computers) belong to the same user. We deem the likeliness of a combination of both events as negligible and the risk as acceptable. The alternative to this risk is the use of authenticators, which are considered immune to malware.

5.4.4 Usable security evaluation

In this section, we assess the performance of the novel mechanism in line with our goal that it shall be usable and secure for users who currently either check for typos or verify the correctness of the transaction context data without requiring significant behaviour change from either. To this end, we evaluate how it would likely impact operations of the three transaction authentication personas. By doing so, we also demonstrate how the transaction personas can be beneficial to the design and evaluation of future transaction authentication mechanisms.

Recall that these personas, developed in section 5.3.3, were derived from empirical data as an approximation of authentic user behaviour and, as we previously described in section 5.2, can thus be deemed helpful to ...

> *... communicate specific user issues and corresponding requirements within and outside the design team, act as stand-ins for users in (the phases of) the design process when real users are not easily available, and [...] help to connect designers to a more tangible representation of [empirical user data] [211]. Jones* et al. *[212] in particular emphasise their use as guidance when exploring the design space of complex systems.*

As a consequence, our assessment of how the novel mechanisms would likely impact the transaction authentication personas can be considered a substitution for real users in an early stage evaluation of a low-fidelity prototype. Table 5.4 provides an overview of the results of this evaluation. However, since personas are always only an approximation of some users, this should not be confused with a representative study. Such a study would require a large number of real users and a realistic high-fidelity prototype or real implementation of the mechanism. Conducting such a study is suggested for future work.

The first persona we described on page 133, Leah Hardware, uses a laptop and authenticator. At current, she is verifying the transaction context data on the authenticator. If this persona were to use an authenticator with the novel mechanism, she could continue her current usage without impact on the usability or security of her online banking experience. If Leah were to embrace the new functionalities of the novel mechanism, she could focus on validating the correctness of the data shown on her laptop instead of the authenticator. This could result in minor usability improvements for some of the users represented by this persona but have no impact otherwise.

Shelly Mobile, our second persona described on page 134, uses an app on her phone for mobile banking without (the need for) active verification of the transaction context data. We do not assume a relevant implementation of the novel mechanism for this use case. Thus, Shelly would continue being protected through her mobile phone's Trusted Execution Environment and additional security measures such as run-time application self-protection (see section 2.3.2).

Lastly, our third persona Jun Message, as described on page 135, uses a laptop and text message. At the *status quo*, he is focused on the avoidance of typos in the transaction context data shown on his laptop and copies security codes from text messages without verifying the correctness of the salient context data in those SMS. The novel mechanism would require him to start using an authenticator or an app on his phone – something Jun might be unwilling or unable to do. If Jun were to comply with such requirements, he could benefit from security improvements without further behaviour change. Crucially, he could continue to focus his attention on avoiding typos on his laptop and would not need to adapt his current automatism of online banking. As a consequence, he could benefit from major security improvements as long as he were able and willing to use an authenticator or mobile phone application – as apparent from our security analysis in section 5.4.3.

	Leah Hardware	Shelly Mobile	Jun Message
Usability	👍	N/A	👎
Security	–	N/A	👍👍

Table 5.4: The novel mechanism would be expected to cause minor usability improvements for Leah Hardware while causing a minor reduction in usability alongside major improvements to security for Jun Message.

5.4.5 Variations of the basic idea

In the description of the basic idea behind our novel mechanism in section 5.4.2, a Visual Cryptogram is used for secure communication of transaction context data from the bank to the secondary, trusted device, which then compares this data with that visible on the primary device. In a more general description of our novel mechanism, the transaction context data could be made available to this device over other communication channels. The basic requirement is that this communication is either on a separate channel or encrypted, e.g. Bluetooth connection with the primary device to transmit data encrypted by the bank or a re-keying mechanism (see section 2.4.2). If the data is transmitted over a non-encrypted separate communication channel, additional measures to ensure the authenticity of that message would be advised. For example, transaction context data delivered via SMS could include a keyed-hash (with a secret key only known to the bank's server and the app on the user's phone) of that data to prove that the SMS was sent by the bank and that its content was not tampered with.

The novel mechanism could also be used to authenticate bulk transactions, i.e. single transactions comprised of multiple payments. At the *status quo*, bulk transactions are typically authenticated with a TAN dynamically linked to the sum of all payments and, depending on the implementation, to the number of payments. Thus, malicious modification of the transaction data could not tamper with the total value of the bulk transaction but could tamper with the value and destination of individual payments (as long as the number and sum of all payments remains the same). The novel mechanism could address this shortcoming in the following way:

1. Acquire data about intended individual transactions in the bulk transaction and the instructions about individual transactions received by the transaction service provider.
   - Take one high-resolution picture of a table showing all intended transactions, a set of pictures, or a video recording with synchronised screen scrolling (e.g. the authenticator or mobile phone could be connected as an input device to the primary device for synchronicity). This requires the existence of a trusted source for the intended instructions in suitable formatting, e.g. a database on an offline/read-only device or printed transaction instructions.
   - Receive relevant about the transaction instructions received by the service provider over a secure channel, e.g. a (large) visual cryptogram or pushed to an app on the user's mobile phone.
2. Compare the intended individual transactions with those received by the service provider.
3. Finally, if those data do not match, the device would indicate this outcome to the user (e.g. an error or warning message). If the data does match, the device could display the corresponding TAN or provide an action (e.g. a button) on the second device for seamless verification of the transaction.

This is the first usable and secure method to authorise bulk transactions in the presence of financial malware.

5.5 Discussion

A priority for user-centred design is to create functional mechanisms while minimising expectations on users to change their current behaviour. While many participants in our online banking study in chapter 4 did not securely verify the transaction context data on their trusted device, we found that participants had intrinsic motivation to avoid or detect typos in the transaction data, and most kept a vigilant eye on the respective data input fields. The novel transaction authentication mechanism proposed by us takes advantage of that desire: We showed its security against Man-in-the-Browser attacks for users who verify the correctness of the transaction context data either on the primary or secondary device when they scan it with the novel mechanism. Based on our personas, we further showed that our mechanism would accommodate many users' current behaviour with little to no changes.

Previous research had described the integration of personas into the design process as a challenge but considered them useful to communicate usability issues between development teams and to non-experts through a more tangible representation [240, 241, 242, 217]. We derived and designed three personas of transaction authentication users from our user study data through a mix-method approach. These personas were then utilised to evaluate the usability of a low-fidelity prototype, which we argue would not have been possible by only looking at the data upon which they are based. This application of personas demonstrates how they can be an effective component in the design process of novel applications and solutions. We argue that a similar approach might be useful for other researchers and practitioners, in particular for the first evaluation of low-fidelity prototypes that might not warrant the effort of a full user study.

5.5.1 Summary

We derived the first transaction authentication personas through a mixed-method approach and showed how personas can be integrated into the design process to become an effective element in the evaluation of novel applications and solutions. We also presented a novel transaction authentication mechanism, analysed its security against Man-in-the-Middle attacks, and evaluated its usability. By doing so, we proposed a suitable and feasible mitigation for one of the main security risks previously identified in chapter 4, which concludes our investigation of mechanisms for transaction authentication to secure data input in online banking.

Chapter 6

Taken Out of Context: Security Risks with Security Code AutoFill in iOS & macOS

A modified version of this chapter has previously been published as:

> Andreas Gutmann and Steven J. Murdoch. Taken Out of Context: Security Risks with Security Code AutoFill in iOS & macOS. 2019. USENIX.

and presented at the *Who Are You?! Adventures in Authentication Workshop 2019*, co-located with the *Symposium on Usable Privacy and Security 2019*.

6.1 Introduction

In June 2018, Apple announced at its Apple Worldwide Developers Conference (WWDC) the introduction of a new convenience feature to their operating systems. This new feature, called Security Code AutoFill, scans incoming SMS messages for relevant numeric codes and suggests them to the user for autofill directly on-screen. This improves user experience and convenience as the user is no longer required to open the messaging application, memorise the security code, and re-enter it on another app or website.

Security codes are sent via SMS for a variety of authentication and authorisation purposes. Each requires the user's mobile phone number to be registered with the corresponding service, while the legitimate user is expected to receive and quote security codes to proceed with certain actions – something an impersonator should be unable to do. A malicious exploit of the Security Code AutoFill feature could expose these services and their users to increased risks.

We analysed the interaction between the Security Code AutoFill feature and security procedures of online services that rely on the transmission of security codes via SMS for authentication and authorisation. Our Cognitive Walkthrough analysis found security risks stemming from a design aspect of Security Code AutoFill: reducing the information provided to users – including salient context information about the SMS message – while relying on them to make cautious decisions about security.

Our findings show that adversaries could exploit this decontextualisation. We describe three attack scenarios in which an adversary could leverage this feature to gain unauthorised access to users' online accounts, impersonating them through their instant messengers, and defraud them during online card payments. Next, we discuss the results and suggest possible measures for affected online services to reduce the attack surface by altering the phrasing of their SMS or using alphanumeric security codes.

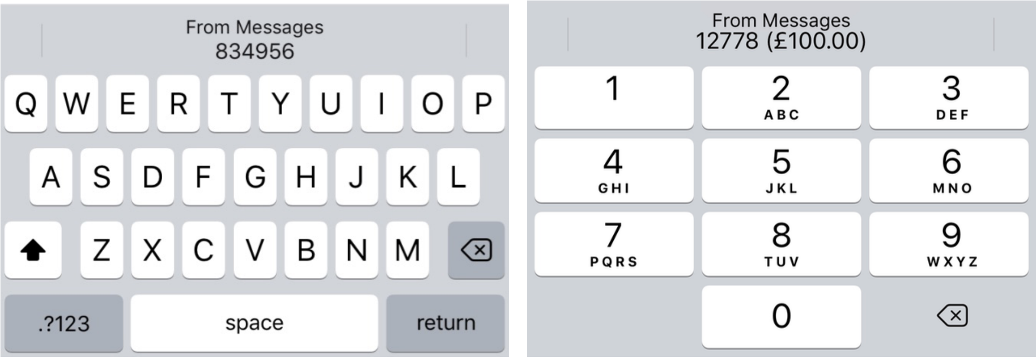
We continue with an exploration of the design space for the Security Code AutoFill feature and sketch two alternative prototype designs. These designs would alleviate our security concerns while aiming at retaining the improved convenience and empowering users and online services to safeguard their interactions. Next, we briefly present an overview of developments subsequent to the disclosure of our findings, in which Apple and Google proposed a new specification for the distribution of security codes via SMS akin to one of our proposals, and Apple updated its OS accordingly. Lastly, we briefly discuss security risks with this feature that remain after Apple's most recent update and recommend amendments.

6.2 Background

In this section, we provide background information on how Security Code AutoFill works and relevant related work on the design of security messages. For additional background information on the relevant security technologies we refer the reader back to section 2.4.

6.2.1 Security Code AutoFill

Security Code AutoFill works by scanning incoming SMS for numeric codes and keywords, e.g. *code* or *codeword*. It also monitors the Safari browser and active apps for places to fill-in security codes. Developers can use `autocomplete="one-time-code"` as HTML code to tag input fields of their forms. This informs iOS and macOS that the developer would like the system to suggest entering OTPs in that field. If a suitable SMS and input field were found, the feature extracts the security code from the SMS and suggests it for autofill (see fig. 6.1a). If the SMS contains an amount of money, e.g. *£100*, this information is displayed in brackets (see fig. 6.1b). The user needs to tap on this suggestion to autofill the code. This feature operates on iOS and macOS but requires SMS synchronisation to be activated for the latter. iOS and macOS suggest security codes for autofill into suitable form fields, e.g. those self-identifying as suitable location with a corresponding HTML tag, within apps and browser for up to three minutes after being received.



(a) iOS AutoFill suggestion for a security code.

(b) iOS AutoFill suggestion for a security code from an SMS that also contained the text "(£100)".

Figure 6.1: Screenshots: Security Code AutoFill on iOS 12 suggesting autofill of security codes.

6.2.2 Design of security messages

The design of messages with which security protocols should communicate critical information was first discussed by Abadi and Needham [243], who proposed the principle of 'Explicit Communication': "Every message should say what it means: the interpretation of the message should depend only on its content." Laughery and Wogalter [244] were concerned with the more broad topic of designing general warning messages and recommended to be concise but clearly convey the message, using concrete rather than abstract wording, and avoid unfamiliar abbreviations or ambiguous statements. Short sentences with short, familiar words should be used preferentially. Messages should be explicit in what the reader should do or not do.

6.3 Threats

We investigated whether the introduction of Security Code AutoFill has affected general threats against the affected services, i.e. whether it ameliorates or aggravates protection against known attacks. We refer the reader back to section 2.2 for details on those general threat models.

The specific adversary considered in our security analysis is located remotely, e.g. without physical access to the online service, user, or their equipment. It is restricted to the usage of publicly known attacks, e.g. no zero-day vulnerabilities. Its active social engineering is limited to sending phishing emails to the user, e.g.

no cold calling. iPhones are not jailbroken, e.g. apps or malware could not access and read SMS. We assume that telecommunication providers and their networks are secure, e.g. no SIM swap scams.

6.4 Methodology

We analysed the interaction of Security Code AutoFill in iOS 12 and macOS 10.14 with security codes delivered by SMS, utilising the Cognitive Walkthrough (CW) method previously described in section 3.3 for an exploratory data collection to gain insights into how users may perceive the new user interface (UI) and functionality, and the impact this could have on security during interactions with affected services.

We began the security evaluation during Apple's *beta* of iOS 12[1]. This required us to be flexible and adapt when a new beta version changed the Security Code AutoFill feature. The choice of an expert evaluation methodology, without the direct involvement of non-expert users, minimises the effort for expected repetitions of previously completed tests due to the release of a new beta version. To avoid priming of participants during a user study, we might have been required to deceive them about the actual purpose of the study. Principle 8.07 of the APA Ethical Principles of Psychologists and Code of Conduct [177] about the use of deception in research requires that any effective non-deceptive alternative procedures have been excluded as not feasible prior to the use of any deceptive techniques. We concluded that, under consideration of the specific context of this security evaluation, a CW would be a feasible alternative.

Andreas Gutmann conducted the CW and evidenced all findings with screenshots and handwritten notes, which were consequently rewritten into detailed descriptions. These findings were then discussed with and verified by Steven J. Murdoch.

[1] Given the expected distribution of new iOS versions to millions of consumer device, we considered a timely security evaluation, prior to the features full release, the ethical choice.

6.4.1 Preparations

We defined the main context, i.e. target of the evaluation, as the iOS 12 and macOS 10.14 UI. The user's main goal during each CW is to complete the task which eventually causes the service to send a security code via SMS, e.g. to login to their remote account. The user's secondary goal is security, e.g. not allowing an adversary to login to their remote account. Their necessary sequence of actions is as follows: (1) proceed with the activity which eventually causes the server to send a security code to their mobile phone, (2) locate the correct input field for the security code, (3) retrieve the security code, (4) conduct security checks as necessary and abort if one or more checks fail, and (5) insert and submit the security code in the previously located input field. Finally, we installed and updated iOS 12 and macOS 10.14 on our devices, prepared a second mobile phone to simulate the adversary and inserted new SIM cards into the mobile phones.

Each CW consisted of a step-by-step analysis of how the UI could guide its user when attempting to execute the necessary sequence of actions for a corresponding task, and how the adversary could try to incorporate this into an attack. At each step of this process, we assessed: (1) Which visual cues are available to the user for the next action and what feedback is provided to them after each action? (2) What actions could an adversary take to get closer to their goal and how could the user foil such an attack at this step?

6.4.2 Limitations

In line with the limitations outlined in section 3.6, the reader may recall that the CW method does not involve (non-expert) users and, therefore, the results are solely based on skills and expertise of the evaluators. This method also commonly identifies only a subset of issues for the evaluated system and the frequency of such issues cannot be estimated solely based on the results of a CW. Yet, this does not reduce the validity of the identified issues.

6.5 Results

In this section, we report on the results from our CW. For each type of security code, we present one scenario, in which the adversary's attack could succeed, as being representative for multiple findings that lead to similar outcomes for both OS.

Although we maintained a detailed record of step-by-step actions during each CW, we limit our reporting to salient information about those actions by the user and the adversary which record relevant information to describe the scenario.

6.5.1 Remote login with OTP

In the scenario described here, an adversary wants to log in to an online account of the user, which is secured with 2FA. As a login portal, we chose the PayPal website[2]. The scenario refers to the iOS 12 UI and assumes that the attacker knows the victim's email address and PayPal password. Note that the user's email address equals their user name for this service.

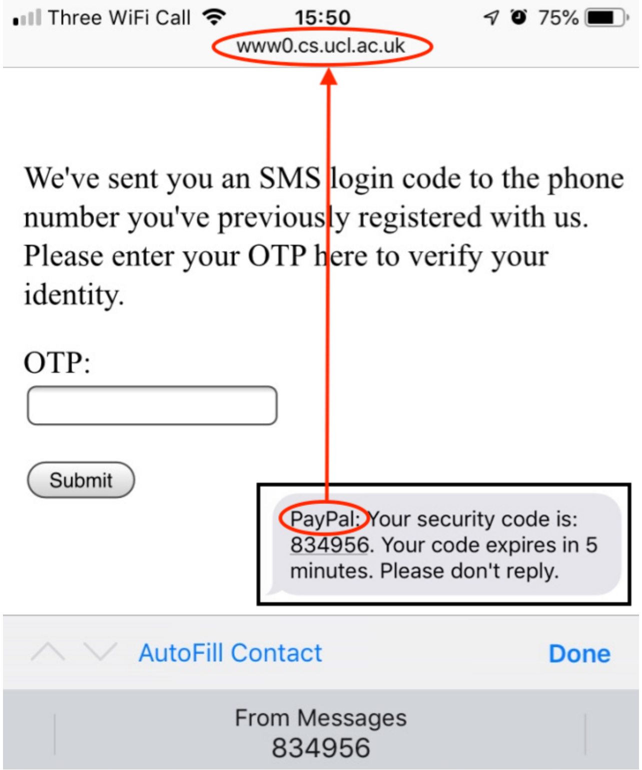
The scenario begins with the adversary sending a phishing email for an unrelated, low-risk website[3] to the user. This could be classified as targeted spear phishing or ordinary phishing. When the user clicks on a link in this email and visits the phishing website, the adversary is notified by a scripted event on that website. The adversary then attempts to log in to the user's PayPal account, which requires an OTP for user authentication and suggests to send the OTP to the user's registered phone number – which is confirmed by the adversary. This SMS also summarises the source and purpose of the code.

The user receives the SMS from PayPal on their phone. The phishing website ensures activation of the Security Code AutoFill feature, e.g. when the HTML tag `AUTOCOMPLETE="ONE-TIME-CODE"` is used on an active input field. The feature suggests filling the 2FA code sent by PayPal into the phishing website's active form

[2] Our results do not indicate any security weaknesses directly associated with this service. It was chosen for illustrative reasons and a different choice would have been possible.

[3] Herley [245] argues that users are less likely to detect phishing emails of websites for which a compromise would cause them little to no harm, compared to those where a compromise would cause them great harm, due to changes in the expected cost-benefit ratio. Following Herley's argument, most users would, for example, be less likely to confirm the authenticity of their local newspaper or a bulletin board compared to their bank's website.

field for up to three minutes after receiving the SMS. If the user follows this suggestion and submits the form, then the phishing website sends the security code to the adversary, which can use it to log in to the user's PayPal account. Without Security Code AutoFill, the user would be required to access and read the SMS in order to retrieve the security code. This would enable them to notice that the SMS was sent by a *different sender* than the website they're browsing, and thus avoid the attack (see fig. 6.2).



We've sent you an SMS login code to the phone number you've previously registered with us. Please enter your OTP here to verify your identity.

OTP:

Submit

PayPal: Your security code is: 834956. Your code expires in 5 minutes. Please don't reply.

Figure 6.2: Screenshot: Security Code AutoFill suggested to fill a security code into an unrelated website. The SMS, depicted as an overlay in this figure, exposes a discrepancy between the SMS content and website.

6.5.2 App registration with OTA

This scenario describes an opportunistic trawling attack aimed at hijacking the accounts of users of an app registered to their phone number. We chose the WhatsApp Messenger app for iOS as an example for our scenario[4]. The scenario refers to the

[4] Our results do not indicate any security weaknesses directly associated with this service. It was chosen for illustrative reasons and a different choice would have been possible.

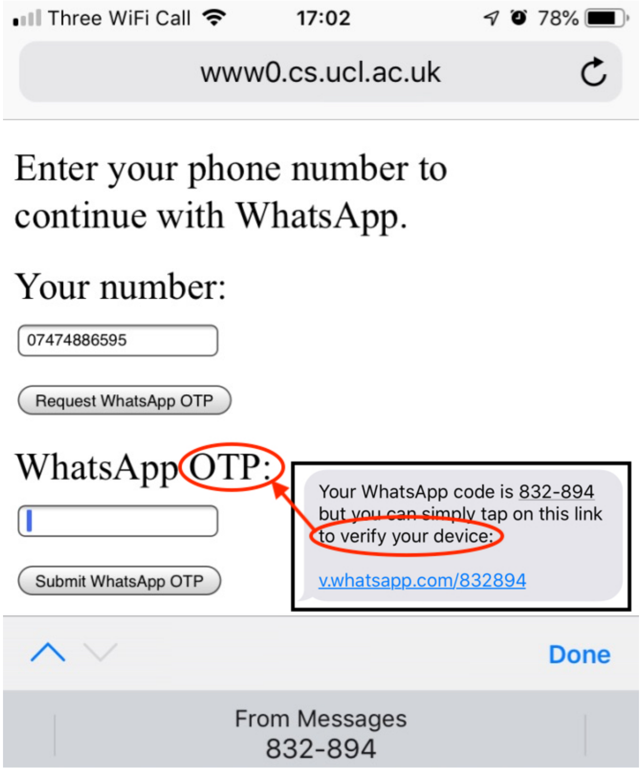
iOS 12 UI and assumes that the adversary is capable of a Man-in-the-Middle (MitM) attack on a public WiFi.

The scenario begins with the adversary executing a MitM on a public WiFi, scanning websites for social login buttons (e.g. Facebook, Gmail, etc.) and injecting a fake WhatsApp login button. The user accesses the attacked WiFi, browses the Internet, and eventually loads a website with social login buttons. They decide to try the (apparently new) WhatsApp login button, unaware of its adversarial nature. The website requests the user to enter their phone number for apparent identification, which they submit and is then transmitted to the adversary. The adversary installs the WhatsApp Messenger app on their phone and submits the user's mobile phone number during registration. WhatsApp sends an OTA to the phone number which summarises the source and purpose of the code.

The user receives the SMS on their phone. The fake WhatsApp login button ensures activation of the Security Code AutoFill feature, e.g. when the HTML tag `AUTOCOMPLETE="ONE-TIME-CODE"` is used on any active input field. The feature suggests filling the OTA into the active form field, which is part of the fake WhatsApp login mechanism, for up to three minutes after receiving the SMS. If the user follows this suggestion and submits the form, the a script on the website sends the OTA to the adversary. The adversary can use this code to register the WhatsApp app on their device to the user's phone number, thereby effectively hijacking the victim's WhatsApp account. Without Security Code AutoFill the user would be required to access and read the SMS, which would enable them to notice that the SMS was sent for a *different purpose* than that described by the website, and thus avoid the attack (see fig. 6.3).

6.5.3 Online payment with Transaction Authentication

The scenario described here is based on an adversary who wants to trick a user into paying for its purchase. We chose the implementation of 3D Secure with Transaction Authentication by Monzo Bank Ltd, and the online shops operated by Voucher



Enter your phone number to continue with WhatsApp.

Your number: 07474886595

WhatsApp OTP:

Your WhatsApp code is 832-894 but you can simply tap on this link to verify your device:

v.whatsapp.com/832894

832-894

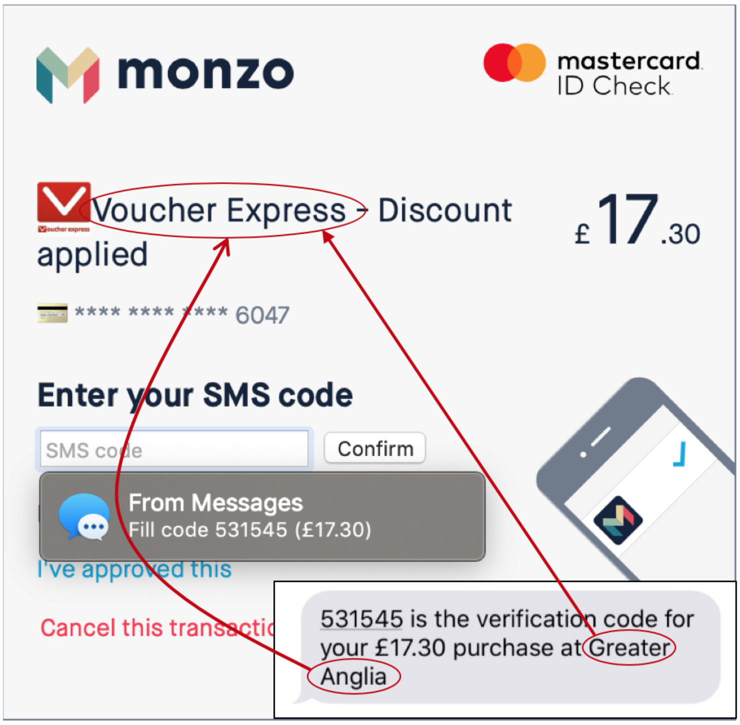
Figure 6.3: Screenshot: Security Code AutoFill suggested to fill a security code into an unrelated website. The SMS, depicted as an overlay in this figure, exposes a discrepancy between the SMS content and website.

Express and Greater Anglia, as examples for our scenario[5]. The scenario refers to the macOS UI and assumes that the adversary was able to infect the user's device with malware capable of a Man-in-the-Browser (MitB) attack.

The user wants to make a credit card payment of £21.75 at the online shop of merchant Voucher Express. The adversary wants to acquire a train ticket worth £17.30 from merchant Greater Anglia. The user selects to proceed to the payment website of Voucher Express but is redirected to a payment website for the train ticket by the malware on his device instead. The malware also tampers with the user's browser view to imitate the intended purchase, including an apparent discount to justify the difference in payment value. The user enters their credit card details on this website and requests a security code via SMS. This SMS also summarises the transaction data received by the bank.

[5] Our results do not indicate any security weaknesses directly associated with these services. They were chosen for illustrative reasons and different choices would have been possible.

The user receives the SMS on their phone. Security Code AutoFill is activated the same way as in the previous two examples in sections 6.5.1 and 6.5.2 and suggests filling the security code into the manipulated website for up to three minutes after receiving the SMS. If the user follows this suggestion and submits the security code, they confirm, and thereby pay for, the fraudulent purchase of a train ticket by the adversary. Without Security Code AutoFill the user would be required to access and read the SMS, which would enable them to notice that the SMS was sent for a purchase at a *different vendor*, and thus avoid the attack (see fig. 6.4).



From Messages
Fill code 531545 (£17.30)

531545 is the verification code for your £17.30 purchase at Greater Anglia

Figure 6.4: Screenshot: Security Code AutoFill suggested to fill a security code for the wrong purchase into the payment website. The SMS, depicted as an overlay in this figure, exposes a discrepancy between the SMS content and website / the user's intentions.

6.6 Discussion

Our security analysis found elevated security risks from the contextualization of security codes by Security Code AutoFill. The attacks we described would be unlikely to succeed if the user were to read the context information in those SMS. We know for more than twenty years that context is critical for security messages [243].

Security codes should not be presented to users without context, but Security Code AutoFill makes any user interaction with such messages optional. We also know that the required contextual information relayed to users in security messages differs based on the type of authentication and authorisation process (see section 2.4), which is why they should not be treated the same by automated systems that cannot reliably distinguish between them. Services sending security codes should be able to determine whether and how auxiliary software interacts in these processes.

Apple made a unilateral decision to introduce a feature which affects services that rely on SMS to deliver security codes. This might encourage more users to activate corresponding security options for their accounts, but affected services have limited options to influence the feature and how it elevates certain security risks for them and their users. Security Code AutoFill detects numeric security codes based on proximity to words such as *code* or *passcode* and cannot be deactivated[6]. Services could avoid having security codes being recognised by omitting such keywords or when using alphanumeric security codes. But this could be detrimental to user experience and the effectiveness would be subject to future design changes, e.g. introduction of new keywords. Modifications to the website or app code alone would only be a partial solution to prevent the feature from activating since an adversary capable of content manipulation could reactivate it.

SMS was not designed with support for security protocols in mind and its use therein has been criticised before. Some of the main risks come from the lack of endpoint authentication and cryptographic binding. Solutions that support these mechanisms, e.g. through asymmetric cryptography, have an advantage. Our findings show a further advantage for authentication and authorisation solutions that retain control over interactions with third-party software on the user's side of these processes. For example, FIDO certified security keys - which refers to the U2F protocol [246] - interact with the operating system via the human interface device protocol, similar to keyboard, and can, thus, be directly accessed by application

[6] Text Message Forwarding can be deactivated to disable SMS sharing between devices linked to the same Apple ID. Preventing macOS from receiving SMS effectively disables Security Code AutoFill on macOS.

software to circumvent interference by other applications or OS-level features.

Services that rely on SMS trade ease of deployment for control over the communication channel. Securing such interactions between independent systems can be a complex and difficult task. Services that wish to utilise benefits from interactions with systems they do not control, while minimising security risks from these interactions, could do so by borrowing techniques from the field of cryptographic proofs: a *reductio ad absurdum*, i.e. *proof by contradiction*, to demonstrate the security of a system. If the existence of a security failure would necessarily require the violation of something assumed to be true, then it implies that no security failure is possible or the assumption is false. One such assumption for systems relying on security codes must be that users can understand the context of each security code before they decide whether to disclose it to an application.

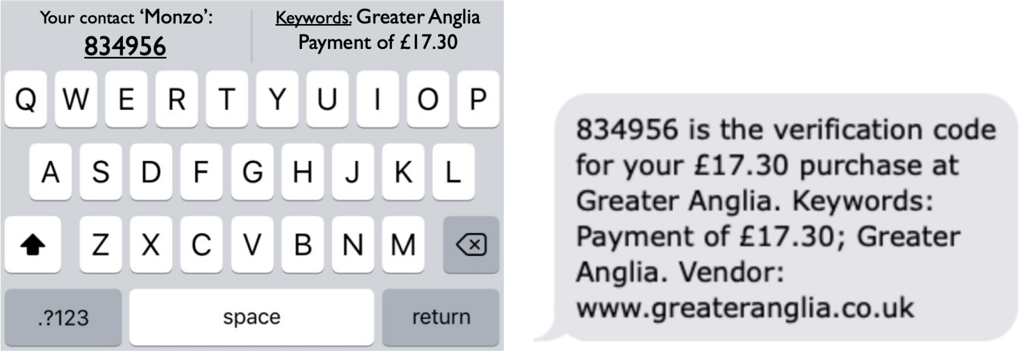
6.7 Exploring the feature's design space

In this section, we describe two opportunities in the design space of Security Code AutoFill to alleviate the risks described in this paper. We present design sketches and briefly compare their expected functionality with the current implementation of Security Code AutoFill as well as to a similar functionality in Android. Further work would be required to implement and evaluate actual, functional designs of these sketches, e.g. following a co-design or thinking-aloud methodology.

We identified two main challenges in the design space of Security Code AutoFill: (1) Salient context data shall be extracted from the SMS, yet it shall remain legible for users without the feature, (2) character and space constraints on the length of SMS and from the device's screen, respectively.

The **first opportunity** aims at displaying more context information from SMS which deliver security codes. The words "From Messages" in current autofill suggestions could be replaced with information about the sender of the SMS, e.g. phone number or contact name. Unused space to the sides of each autofill suggestion could display further context information, if available. To identify such information, additional keywords and design patterns could be specified, e.g. words "sent by" or "purchase at" followed by the desired display name. Similar patterns could apply to other context information such as transaction value and purpose of the code. Figure 6.5a is a design sketch of such an autofill suggestion.

The **second opportunity** would be to allow SMS senders to identify the targeted destination for a security code, e.g. URL or app name, with specified keywords. The feature would then only suggest codes for autofill into the designated destination (if provided). Figure 6.5b is a design sketch of such an SMS.



834956 is the verification code for your £17.30 purchase at Greater Anglia. Keywords: Payment of £17.30; Greater Anglia. Vendor: www.greateranglia.co.uk

(a) Design sketch of alternative autofill suggestion for an online payment. Sketch of corresponding SMS shown in fig. 6.5b.

(b) Design sketch of security code sent via SMS with additional keywords identifying salient context information.

Figure 6.5: Design sketches of our proposed alternative UI for autofill suggestions and corresponding SMS delivering a security code with additional keywords.

Comparison. Security Code AutoFill was described earlier in section 6.2.1. Android, on the other side, supports a method for cryptographic binding to specify in the SMS the intended recipient of a security code. Apps can be identified through their cryptographic hash, sent to the remote server when the app requests a security code. The server embeds this hash alongside a security code in an SMS sent to the user's device. When the SMS is received by the user's device, Android identifies the intended app through its cryptographic hash and makes the message text available to this app through the *SMS Retriever API*. Finally, the app needs to call the *SMS Retriever API*, receive the message text, and parse the security code from it. This provides end-to-end security but only for apps on the receiving mobile device. Security codes intended for websites or apps on other devices, as well as those containing a TAN, need to be processed by the user.

The opportunities for Security Code AutoFill we described could empower users to validate context information of all security codes while retaining the autofill functionality but would face limitations with respect to the maximum length of SMS. In comparison with Android, our proposal supports all types of security codes intended for any connected device but doesn't fully automate the interaction.

6.8 Ethical consideration

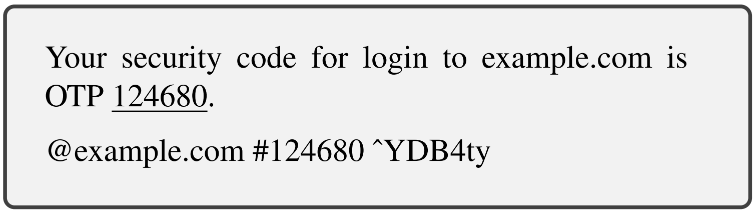
Security professionals have an ethical obligation to ensure their knowledge is shared responsibly, especially when disclosing risks for systems that have been deployed. In line with our responsible disclosure procedure, we disclosed our concerns during the beta of iOS 12 and informed Apple. This procedure is compliant with the requirements by our institutions.

6.9 Subsequent updates to Security Code AutoFill

An update during the beta of Security Code AutoFill, after our initial disclosure of identified security risks, added transaction values to the autofill suggestions (see fig. 6.1b). This change was incorporated in subsequent CWs after we had confirmed that it addressed some of our previously findings regarding the arbitrary manipulation of transaction values. We have incorporated those changes to the feature in the work presented in this chapter, i.e. we did not include the security risks which were addressed during the feature's public beta. At the time of submission of this thesis, Apple has taken further steps to address most of the remaining security risks, which we will describe in the following.

After we had concluded our research on Security Code AutoFill, and it had been published [23], Apple disclosed that they have engaged with Google to jointly develop a new, standardised format for SMS to deliver security codes, akin to the second opportunity we described in section 6.7. The proposed specification would split the SMS into essentially two parts: The first line is arbitrary, human-readable text intended for the user while the second is machine-readable formatting conventions to help operating systems extract salient information from the SMS.

Figure 6.6 provides an example of a WebOTP-conform SMS content, whereby @example.com refers to the web domain of https://example.com, #123456 is a security code intended for that domain, and ˆYDB4ty is a cryptographic hash to identify the corresponding app (e.g. browser). Note that the latter parameter is included in Google's current proposal[7] but not in the proposal by Apple[8]. In autumn 2020, Apple implemented these new functionalities of Security Code AutoFill with iOS 14 and macOS 11.0 [247] and publicly acknowledged our contributions to the security of their products [248].



Your security code for login to example.com is OTP 124680.

@example.com #124680 ˆYDB4ty

Figure 6.6: Example formatting of WebOTP-conform SMS where 124680 is the security code for domain example.com and YDB4ty is the application identifier.

6.10 Outstanding security issues

In this section, we look at security issues we found that remain after Apple updated the Security Code AutoFill feature after our disclosure and publication. We make two recommendations on how to address these issues. These recommendations are in addition to the first opportunity described in section 6.7, which would enable transaction authentication service providers to utilise this feature rather than it being a security risk to them. We have submitted our concerns about these security risks to Apple's Product Security Team.

Since the proposed WebOTP specification is akin to the second opportunity described in section 6.7, we believe that it could improve the security of user authentication and device authorisation with the Security Code AutoFill feature, but not for transaction authentication. Yet, at the time of submission, the WebOTP specification is only optional rather than mandatory, i.e. Security Code AutoFill continues to interact with security codes delivered via SMS in other formats. Thus, user authentication and device authorisation service providers, which send security

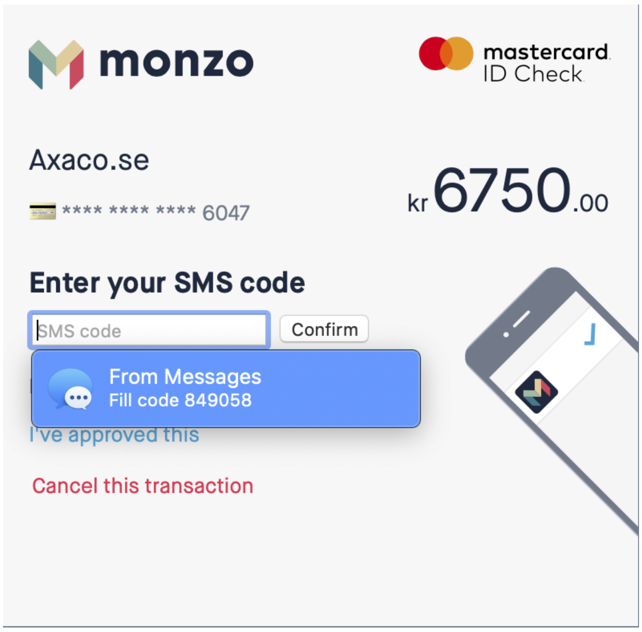
[7] `https://github.com/samuelgoto/WebOTP/blob/master/explainer.md` (Accessed: 30 March 2020)

[8] `https://github.com/WebKit/explainers/tree/master/sms-one-time-code-format#proposal` (Accessed: 30 March 2020)

codes via SMS, remain vulnerable unless they are aware of, and adopt, the WebOTP specification.

Recommendation 1: We recommend to make adherence to the WebOTP specification of incoming SMS a prerequisite for the activation of the Security Code AutoFill feature as soon as possible.

Next, we also found limitations in the feature's ability to display transaction values (see figs. 6.1b and 6.4, for examples) in certain currencies. Figure 6.7 provides an example of a transaction in Swedish crowns, locally named Krona (abbreviated *KR*), in which the Security Code AutoFill mechanism does not extract and display the corresponding transaction value. Figure 6.8 shows the respective SMS, which displays the transaction value. Thus, for transactions in certain currencies, a user who would follow an autofill suggestion could unknowingly agree to a malicious transfer only limited in value to the maximum of what his bank would allow.



monzo

mastercard ID Check

Axaco.se

**** **** **** 6047

kr 6750.00

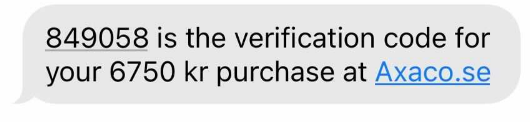
Enter your SMS code

SMS code

From Messages
Fill code 849058

Figure 6.7: Screenshot: Security Code AutoFill suggests filling a security code for an online payment, protected by 3D-Secure, without displaying the transaction value of 6750 Swedish crowns. Corresponding SMS displayed in fig. 6.8.

This security risk applies to both 3D-secure protected card-not-present transactions (i.e. online payments) and online banking transactions. It would also be possible that a user who intents to make a domestic transaction in the UK, for ex-



849058 is the verification code for your 6750 kr purchase at Axaco.se

Figure 6.8: Screenshot: SMS delivering a transaction authentication security code for an online payment of 6750 Swedish crowns to vendor Axaco.se.

ample, could unwittingly authorise a transaction of significantly larger value in a foreign currency to an account under the control of the adversary. Thus, users in every country remain at risk unless every country's currency is detected.

Recommendation 2: We recommend that the Security Code AutoFill feature should recognise all currencies in their international and domestic names, symbols, and abbreviations.

6.11 Summary

We analysed Security Code AutoFill in iOS 12 and macOS 10.14 and found security risks stemming from the contextualization of security codes, removing salient context information while requiring users to continue making security-cautious decisions. Our findings show an advantage for security messages delivered directly to users, not only to their devices. We described two opportunities in the feature's design space to retain the improved convenience while empowering users and online services to safeguard their interactions. Specifications akin to one of our two proposed improvements were adopted by Apple and Google, and Apple updated the Security Code AutoFill feature iOS 14 and macOS 11.0 accordingly. Yet, some security risks remain.

Chapter 7

Human Error Theory and Incident Investigation

In this chapter, we place our findings in the context of established models and theory about human error. Human error has been a major concern in a large number of academic disciplines, areas of the economy, as well as the society in general. In section 2.1, we described the taxonomy of human error on the level of individual operators as either mistakes and slips. Beyond this distinction, researchers studied past incidents of human error for causal influences from the environment. Theoretical models to encompass such influences were derived from the incident descriptions to not only distinguish between different types of error but also classify contextual factors that may contribute to or enable human error. These models led to methods for the investigation of human error incidents, both predictive in nature and to support investigations after an incident.

7.1 Models and theory

Bello and Colombari [249] gave one of the earliest differentiation between four key contexts of human error in the immediate actions of trained control room personnel erroneously operating a plant. They defined human error to occur when an operator's actions are sub-optimal and the consequences of their actions exceed acceptable boundaries of the plant's system state. The four principal contexts of human error are described as:

1. Operators do not understand or misinterpret information they receive.
2. Operators understand the information they receive but do not know how to appropriately respond to this knowledge.
3. Operators understand the information they receive but every appropriate response to this knowledge would be outside their possible actions.
4. Operators understand the information they receive but the (appropriate) response to this knowledge is executed wrongly.

James Reason generalised the notion of human error as all those occasions in which a planned sequence of mental or physical activities fails to achieve its intended outcome, and when these failures cannot be attributed to the intervention of some chance agency [27]. His model of organisational risk management [250, 251] – dubbed the *Swiss Cheese* model – has been instrumental towards a holistic perspective on safety and human error in accident prevention. It describes four contexts of human failure, whereby the former mentioned contexts can influence the latter: (1) *organisational influences* are causal factors at all levels of an organisation that affect most human operations within that organisation, (2) *unsafe supervision* is a failure of appropriate intervention and mitigation strategies to support and regulate the operations of supervised personnel, (3) *preconditions for unsafe acts* are the mental and physical conditions under which a human operates, and (4) *unsafe acts* are the specific action(s) or inaction(s) by a human that caused an unwanted system state described as human error.

In the Swiss Cheese model, each organisation has established imperfect barriers for each context to prevent accidents. These barriers are symbolised with slices of Swiss cheese and the imperfections in the precautions with the holes in each slice. Individual imperfections on one slice are usually compensated for with measures taken on another. But over time, the holes move around in the imaginary space of the Swiss cheese as reflection of changes in the context and personnel of the operation. An accident occurs through a combination of event trajectories if and when holes in all slices align, creating a combined trajectory that penetrates all defences.

In this model, it is insufficient to compare the state of a system immediately before and during an accident to identify the cause. Instead, it considers that some contributing factors might have been in place for a prolonged time without causing an incident because the imperfections were (temporarily) compensated elsewhere. This distinction between active and latent failures within the causal sequence of events is important as it may unveil failures that had been undetected for a extended time and only manifest themselves under specific, less common conditions. Such latent failures could be described as the underlying causes of accidents which translate into conditions that can provoke (or contribute to) human error (e.g. time pressure, understaffing, inadequate equipment, fatigue, and inexperience) and weaken existing measures implemented to prevent such accidents (e.g. untrustworthy alarms and indicators, unworkable procedures, design and construction deficiencies) [252]. According the Reason *et al.* [253] these latent failure are the more dangerous gaps in the safety of a system and are "created by the decisions of designers, builders, procedure writers, top-level managers and maintainers."

Building on top of the Swiss Cheese model, Shappell and Wiegmann [254] studied several hundred aviation accident descriptions and recordings, and derived specific categories of human error on each level of the Swiss Cheese model. The resulting system was labelled the Human Factors Analysis and Classification System (HFACS). Its reliability and content validity was subsequently confirmed during the investigation of approximately 1,000 aviation accidents [255]. Today, HFACS has become a foundation for human error incident investigation in the context of various industrial and work environments, e.g. railways [256], maritime [257], mining [258], and healthcare [259]. The following list provides a rough overview of HFACS derived from [260]:

Organisational influences

- Resource management on a corporate level is responsible for the appropriate allocation and maintenance of relevant assets under the control of the organisation.
- Climate within an organisation refers to the working atmosphere and is reflected in the chain-of-command, delegation of authority and responsi-

bility, communication channels, policies, and individuals' accountability.

- Operational processes such as the rules and procedures that govern people's conduct.

Unsafe supervision

- Supervisor deficiencies are when personnel was not provided with sufficient guidance, training opportunities, leadership, and motivation.
- Failure to correct a known problem occurs when a supervisor knew of a relevant deficiency but failed to address it and mitigate its negative effects.
- Planned inappropriate operations are conduct which introduces unacceptable risks outside of emergency situations.
- Deliberate violations of rules and regulations by supervisors can cause accidents but are not classified as human error.

Preconditions for unsafe acts

- Adverse mental and physiological states are temporary conditions that affect an individual's performance, e.g. (mental) fatigue, overconfidence, complacency, and medical conditions.
- Physical and mental limitations are permanent conditions that affect an individual's performance, e.g. sensory and motor limitations, limited attention capacity, delay in response time, and aptitude for certain behaviour.
- Substandard practices occur in groups of people as deficits in team coordination and task allocation or on a personal level as being in an appropriate condition to commence work.

Unsafe acts

- Skill based errors or slips relate to basic skills and highly automatised behaviour that occurs without significant conscious thought. As a result, these skill-based actions are particularly vulnerable to failures of attention and/or memory (e.g. task fixation, out-of-order execution of steps in a procedure, omitted items in a checklist, or forgotten intentions) and subject to operators' individual differences in task execution.

- Decision errors or mistakes are intentional behaviour with correct execution that is inadequate or inappropriate for the situation. These can be due to a misclassification of the situation and execution of a standard procedure, insufficient knowledge and consequently the wrong choice of possible actions, or inability to deduce the correct response to a problem without standard procedure.
- Perceptual errors are erroneous actions due to degraded or confused sensory input as well as misinterpretation of such information.
- Deliberate violation of rules and regulations can cause accidents but are not classified as human error. Such violations can be habitual by nature and socially accepted as minor bending of the rules or constitute behaviour that is atypical at the individual and societal level.

7.2 Our findings on human error

In chapter 4, we explored the behaviour of online banking users when authenticating their transactions. Through our multi-method approach, we were able to identify counts of human error and gained insights into correlations between people's actions as well as their reasoning behind them.

Considering the four contexts of human error by Bello and Colombari [249], we could classify the observed insecure behaviour – as described in section 4.5.2 – as of the second category: participants understood the information on their device(s) but appeared to not know how to appropriately respond to this knowledge, e.g. how to securely verify the transaction details. On the other hand, as we discussed in section 4.6.3, participants also appeared to be distracted by their concerns about typos and how this might have impacted their ability to securely verify the transaction data. One could then classify this behaviour as of the third category: information was understood but the appropriate response was outside participant's possible actions. Lastly, one could argue that participants who intended to securely verify their transactions but did so by comparing them with the laptop's screen instead of the printed instructions executed an appropriate response wrongly, i.e. the fourth category. Thus, it appears that the same behaviour could be classified in different ways, which suggests that this model of human error is of limited usefulness to classify the behaviour we observed.

In the context of Reason's model of human error, we can classify the insecure behaviour we observed during the demonstrations of online banking, as described in section 4.5.2, as *unsafe acts*. In the subsequent sections of that chapter, we uncovered factors that correlated with and likely contributed to these unsafe acts. In the following, we will discuss how such factors could be classified in the Swiss Cheese model.

Several *preconditions for unsafe acts* were unveiled during the interview and described in section 4.5.4. For example, some study participants were distracted by reasonable concerns about typos in the transaction data combined with the people's limited attention capacity (as described in sections 4.5.4.1 and 4.6.3, respectively), while others thought the transaction verification would not be an essential step as they believed they would be reimbursed in case of a fraudulent transaction or could recall it within a certain time frame (as described in section 4.5.4.4).

The category of *unsafe supervision* in the Swiss Cheese model is more difficult to apply to our context than the aforementioned *unsafe acts* and the *preconditions for unsafe acts*. Reason's model of human error was derived from and developed for workplace environments, where personnel supervision is the norm. According to Reason [250], human error in the unsafe supervision category can be found in the behaviour of "*specialists who implement the strategies of the decision makers within their particular spheres of operation: operations, training, sales, maintenance, finance, safety, engineering support, personnel, and so on.*"

Online banking users, or consumers of products and services in general, are not subject to direct or indirect supervision in the same way as employees. Yet, there are other forms of direct and indirect communication from people of (perceived) authority to guide and advise consumers on behalf of the bank and to implement the strategies of the bank's decision-makers. This includes all knowledge transfers about how to do online banking and information about the risks relevant for online banking. Under such an extended definition of *unsafe supervision*, the aforementioned specialists would include most employees at a bank who directly or indirectly interact with the bank's customers (or other consumers of the bank's products), e.g. clerks at a branch, people in charge of advertisement campaigns and information leaflets, helpline and support workers, and those responsible for the information shown on the bank's website. This would enable us to categorise some of our findings on inaccurate knowledge of online banking users – assuming that specific communications by people who acted on behalf of the bank were an unintentional but contributing factor – into the Swiss Cheese model, e.g. participants' misconceptions about fraud insurance and their alleged ability to recall transactions within a couple of weeks (as described in section 4.6.2). Furthermore, we believe that our proposed extension of the unsafe supervision category to include more general acts of unsafe guidance and advise from a position of (perceived) authority could be beneficial for other non-workplace environments.

Further explanation for some of the misconceptions we found among online banking users could be found in the category of *organisational influences* described in the Swiss Cheese model. This could, for example, be the case if communications that contributed to such misconceptions (e.g. advertisement campaigns or advice given by helpline and support workers) were implemented according to company policy or brought about by insufficient allocation of company resources – as opposed to an isolated mishap by an individual employee responsible for a specific piece of relevant communication.

Similarly, we have seen differences between the effectiveness of SMS and authenticators for transaction authentication. If such differences are known to the corporate management– or should be known to, in particular, if it is publicly available information – then it would be their responsibility to decide on possible remedies. The same perspective holds for our findings in section 3.4: We found cases in which the UI of Windows 10 and macOS 10.14 inaccurately described an operation as data erasure when it was in fact data deletion. This has to be accounted for as a contributing factor to accidental data breaches from decommissioned storage devices and can be categorised as human error in the *organisational influences* category. Similarly, in chapter 6, we identified the design of the Security Code AutoFill feature in iOS and macOS as a cause of potential human error during the usage of security codes for authentication and identification purposes.

In summary, we found that Reason's Swiss Cheese model of human error can be used to adequately categorise our findings on human error after a minor extension to the category of *unsafe supervision*. Inspired by the HFACS, we summarised a representative set of human error instances categorised in the Swiss Cheese model based on our findings, including reasonable explanations for human behaviour derived from our findings on a basis of evidence, in table 7.1.

Categories	Descriptions of examples
Unsafe acts	No attempt to authenticate online banking transactions
	Failure to securely authenticate online banking transactions
Preconditions for unsafe acts	Lacking knowledge about importance to authenticate transactions
	Lacking knowledge how to securely authenticate transactions
	Distracted by concerns about typos in transaction data
Unsafe supervision, guidance, and advise	Dissemination of knowledge that inadvertently contributes to user's lack of knowledge about the importance to authenticate transactions
Organisational influences	Corporate policies that lead to inadequate knowledge dissemination
	Deployment of inadequate mechanisms
	Deployment of inferior mechanism with insufficient remedy

Table 7.1: Classification of some of our findings and possible explanations for these findings into Reason's Swiss Cheese model of human error.

7.3 Human error incident investigation

The theory and models of human error presented in section 7.1 commonly underpin investigations into accidents and incidents of (suspected) human error. In this section we take a look at common methodologies for such investigations.

Lundberg *et al.* [261] investigated and compared eight accident investigation manuals. They cover the areas of marine traffic, aviation, work places, railway systems, road traffic, medical care, a major oil company, and nuclear power plants. While they share a common basis in Reason's Swiss Cheese model they differ in other aspects, such as whether a *safety culture* is an explicit or implicit factor. In general, the manuals focus on data analysis and documentation of findings but less on planning, deriving recommendations, follow-up actions and implementation. The typical stages and activities of investigation described in these manuals are:

1. Initiation of an investigation.
2. Planning the project and allocated resources.
3. Collection of relevant data.
4. Represent the data in a format suitable for the investigation.
5. Reconstruction of the causes and conditions that led to the incident.
6. Recommendation of remedial actions.
7. Documentation of findings and conclusions.
8. Decisions about actions and implementation of remedial actions.
9. Follow up activities.

The reliability of accident investigations depends on a number of factors, in particular what and how data are collected [262]. Yet, data collection is a highly subjective one [263]. According to Lundberg *et al.* [261], commonly recommend activities for data collection are the sighting of relevant documents, interviews of relevant personnel, and the inspection of the incident locations.

We believe that the Cognitive Walkthrough method could be a valuable addition to other data collection processes during accident and incident investigations. In chapters 3 and 6 we showed how this method can be used to collect data, how such data can be used to reconstruct causes and conditions for incidents, that such data can be sufficient to derive remedial actions, and contribute to structured reports of the findings and conclusions. Such an approach to data collection can be particularly beneficial in cases for which data collected through other means (e.g. interviews) is sparse or non-existent. We predict that this will be particularly helpful for the investigation of incidents involving people that are not in an employment relationship with the company, e.g. customers and former employees.

While the subjectiveness of the data collection during Cognitive Walkthroughs as a research methodology is a common criticism, it could be considered equally or less subjective than much of the established practice in accident investigation described by Strauch [263]. Indeed, some subjectiveness might be essential during such data collection to account for the wide variability in possible contexts of accidents. We suggest future work to explore means of integrating (a tailored variation of) the Cognitive Walkthrough method into structured approaches for accident investigation and incorporate it into relevant manuals where possible.

7.4 Summary

In this chapter, we placed the insights we gained on human error into the context of established models and theory. While we found that our insights resonate well with Reason's Swiss Cheese model, we also noted that our findings were gained in the context of a non-workplace environment whereas the existing literature focused on workplace settings. From this change of perspective, we deduced an extension to the category of *unsafe supervision* in the Swiss Cheese model for such environments. We also found that the Cognitive Walkthrough method, which is extensively used throughout this thesis, can be a useful tool for data collection during human error incident investigations.

Chapter 8

General Conclusions

In this work, we considered challenges for reliable data input and processing as part of the rapid, large-scale end-to-end digitisation of previously analogue systems and processes. We gained a better understanding these challenges and, thereby, gained insights into how to address them. While we do not want to overgeneralise form case studies, we argue that some insights on challenges faced by the Fourth Industrial Revolution can be gained from our work. That is, because it seems inevitable that multi-purpose systems of various vendors will be interconnected and highly interactive. The alternative to such multi-purpose systems is to automate analogue processes through a tailor-made transformation at an individual level, which is a labour and time-intensive process. The re-use of existing tools and technologies while digitising analogue processes, on the other side, can speed up these transformations while reducing their associated costs. Thus, while some processes receive such a tailor-made transformation, many re-use existing digitised products and services as component within a newly digitised product or service.

In this thesis, we learned about specific challenges if the re-used tools and technologies remain independent subsystems, because then they are also likely to be patched and updated on an individual level by their respective vendors. In that case, it can be that such vendors would be unaware of the various interconnected and dependent digital services that have integrated the vendor's product. If a service relies on the interplay with an independent subsystem, its functionality can be affected by how that system evolves. In chapter 6, we showed that reliance on independent subsystems as part of mechanisms that require reliable data processing can have dire consequences on the example of security processes. These security processes consequently lost part of their functionality and became vulnerable to various attacks solely because the independent subsystem (in this case, the mobile device's operating system) was updated with a new feature by its vendor.

We have also learned that as the environment of systems evolves, so do the requirements on the mechanisms that operate in the environment. In chapter 3, we found that systems which might have once been suitable for their job can become inadequate if the changes in the environment are not reflected through changes to their design. Consequently, entities introducing changes to an environment should not assume that previously suitable tools will automatically continue to enable users to meet the new or changing expectations. And while it is a common understanding that software has huge economic potential as it can be both cheap and quick to adapt the volume of production and service offerings to shifts in the market's capacity, we learned that it would be a fallacy to assume that such products and services would not require constant monitoring of and adaption to changes in their environment.

Addressing these challenges requires continuous effort. In our research, we demonstrated how simple methods such as Cognitive Walkthroughs can be efficient and effective for ongoing re-evaluations of systems that are likely to face rapid change. But if individual systems seem to be well-designed for their purpose and environment, yet there is evidence for undesirable outcomes when used in practice, more complicated methods could be required for assessments of those circumstances. We demonstrated in chapter 4 how such an approach can unveil psychological and sociotechnical factors contributing to undesirable outcomes that are not caused by the respective mechanism's design yet affect its efficacy. This can lead to design adaptations to mitigate these effects on the mechanism and serve as evidence for the respective authorities that changes to the environment of that system could be beneficial or necessary.

8.1 Work done in this thesis

We identified three specific requirements for this transformation in the ability to (a) capture accurate data of users' behaviour and intent, (b) the preservation of the integrity of the communication and (c) reliability of the processing of such data between distributed systems as well as in the presence of malicious actors. These analyses were presented as independent cases studies.

In the first case study of this thesis, presented in chapter 3, we analysed the User Interface implementations in Windows 10 and macOS 10.14 for their ability to accurately capture user's intention to delete or erase data from their storage devices. We conducted several Cognitive Walkthroughs to gather information on how users can be guided by the interfaces to find and select functions that align with their intent and the feedback they receive from those systems on their actions.

We found shortcomings in how the interfaces in both Operating Systems guide their users (e.g. fig. 3.2), how they label their functions (e.g. fig. 3.5), how they describe those functions (e.g. fig. 3.1), and the information they provide after a user has selected such functions (e.g. fig. 3.6). We discussed explicit changes to both Operating Systems to correct the identified shortcomings in sections 3.5.1 to 3.5.3, how the wider research and design community could remedy such issues by using more clearly differentiated and less conflated terminology for delete and erase functionalities in section 3.5.4, and how policymakers and supervisory authorities could contribute in line with their existing mandate by issuing guidance on law-conform data erasure in section 3.5.4.

In chapter 4, we presented the second case study on mechanisms used in online banking to protect the integrity of transaction data submitted by a bank's customer to the bank against threats posed by potential malware infections of that customer's device. We conducted a multi-method user study with 21 participants from two countries and involving the online banking platforms of nine financial institutions. During the course of this study we collected data through four primary instruments: questionnaire, thinking-aloud demonstration, drawing, and interview to gain insights into how people use these mechanisms and the context of when the mechanisms fail to provide the intended security guarantees.

We found significant differences in the effectiveness of different mechanisms to secure online transactions and were able to identify several correlated factors, such as the threats that participants were aware and concerned of in section 4.5.3, how they felt responsible for their transactions in section 4.5.4.1, the perceived complexity and corresponding impact on the memorability of the mechanisms in section 4.5.4.2, participants' understanding of the purpose the mechanism in section 4.5.4.3, and their concerns about fraud in section 4.5.4.4. In section 4.5.5, we collated a list of security practises that participants understood to be beneficial for their online banking security.

We derived three personas to represent typical online banking users from the qualitative data of section 4.5 through statistical methods in section 5.3. This uncovered two distinct dimensions as a reliable representation of the user study data: People's "trust in security of online banking by design and default" and their "knowledge about technical aspects of online banking security".

A novel transaction authentication mechanism, presented in section 5.4, was motivated by the findings presented in section 4.5. Security and usability analyses were presented in sections 5.4.3 and 5.4.4, respectively, whereby the latter utilised the personas from section 5.3.

We discussed the differences between the established transaction authentication mechanisms in section 4.6.1 and the impact of knowledge, distractions, and friction on these mechanisms in sections 4.6.2 to 4.6.4, respectively. In section 4.6.5, we discussed implications for the implementation of transaction authentication in online banking, the communication between banks and their customers, and the activities of regulators and legislators.

The last case study, presented in chapter 6, is an evaluation of the Security Code AutoFill mechanism in iOS and macOS to process security codes as they are being sent by a server via SMS to a user who is expected to quote them on a website or app to authenticate themselves, register an application on their device, or authorise a transaction. We conducted several Cognitive Walkthroughs to gather information on the possible changes to the effectiveness of these processes when users embrace the new functionality.

We found plausible degradation of the security properties for all aforementioned use cases of security codes sent via SMS in section 6.5, caused by the decontextualisation of security codes for which users need to make security-conscious decisions. In section 6.7, we proposed two improvements to this feature to alleviate the security risks for all use cases. Finally, section 6.9 outlines subsequent changes to the Security Code AutoFill feature that align with one of our two proposals and, thus, address some of the security risks.

Finally, we considered insights from our work on the theory of human error and related incident investigation methodologies in chapter 7. In section 7.2, we broadened the interpretation of the Swiss Cheese model of Human Error, and the associated Human Factors Analysis and Classification System, to account for non-workplace environments and described how our findings fit into these classification systems. In section 7.3, we argued that, based on our findings, the Cognitive Walkthrough method would be a useful component for the data collection stage in human error incident investigation methodologies.

8.2 Impact of this thesis

8.2.1 Impact for academia

We contributed to the literature on the creation and usefulness of personas from qualitative data. As we previously discussed in section 5.2, Sinha [221] and Tu *et al.* [222] used PCA on their qualitative data to reduce the number of potential personas before determining them using HAC. We, in contrast, use MCA and HAC in section 5.3.3 since our qualitative data was categorical. Indeed, according to a recent survey on quantitative persona generation by Salminen *et al.* [264], we are the first to use MCA during the persona generation. Furthermore, previous research described the integration of personas into the design process as a challenge but considered them useful as a communication tool [240, 241, 242, 217]. In section 5.4.4, we used our personas for the evaluation of a mechanism that had been designed with the raw user data in mind (see section 5.4.1). To the best of our knowledge, this use of personas is novel in the academic discourse about the persona methodology.

In section 7.2, we contributed to the understanding and classification of human error as a theoretical construct, in particular related to the Swiss Cheese model. By classifying instances of human error observed or derived from our user study in chapter 4, we were able to extend the definition of the "unsafe supervision" category to include acts of unsafe guidance and advice by people of (perceived) authority that contribute to a transfer of knowledge to device operators even if they are not employees. This is, to the best of our knowledge, the first extension of the Swiss Cheese model to non-work environments.

8.2.2 Impact for industry practitioners

In terms of direct impact for industry, we found security, usability, and privacy issues with the individual state-of-the-art systems and technologies that we analysed throughout our case studies: We described issues with Windows 10 and macOS 10.14 in section 3.4, with implementations of transaction authentication mechanisms in section 4.5, and with iOS 12 and macOS 10.14 in section 6.5. We then proposed design changes to mitigate the identified risks and, thus, contributed to iterative improvements of the respective state-of-the-art technologies: We proposed changes to Windows 10 and macOS 10.14 in section 3.5, changes to the implementations of existing transaction authentication mechanisms in section 5.4, a novel transaction authentication mechanism in section 4.6.5, and changes to iOS 12 and macOS 10.14 in section 6.7. In more general terms, we believe that these findings also call for more investigations into the suitability of other interfaces commonly used by non-experts when reliable data input and processing is required.

Specifically, concerning our findings in chapter 6, the changes to the Security Code AutoFill mechanism proposed by us and implemented by the manufacturer (as described sections 6.7 and 6.9, respectively) are likely to help secure the respective interactions of millions of users - although we suggested additional improvements to further secure these interactions. Regarding the novel transaction authentication mechanism described in section 5.4, we are working with a manufacturer of transaction authentication technologies to patent the idea and explore possible implementations.

In section 7.3, we discussed how our findings in chapters 3 and 6 evidence that the Cognitive Walkthrough method can be effective to collect data for human error incidence investigations. Related work argued that the reliability of such investigations depends on comprehensive data collection [262] and reported that investigation manuals usually lack data collection methods suitable for scenarios in which it would be difficult or impossible to interview people who were involved in an incident and comprehensive documentation of that event is unavailable or scarce [261]. We concluded that, thus, the Cognitive Walkthrough method would be particularly valuable for such human error incident investigations under such conditions.

In chapter 7, we extended the Swiss Cheese model and HFACS incident analysis framework to non-workplace environments, which are being used by incident investigators across disciplines. We further argued that the Cognitive Walkthrough method could be helpful to incident investigators to gather useful data during incident investigations, particularly when information from other sources is scarce.

8.2.3 Impact for policy practitioners

Some of our findings are also relevant for policy practitioners. We focus on pan-European institutions and those within the UK or Germany, yet, we stress that many of these findings are equally useful to comparable institutions in other countries inside and outside the EU.

Our overarching findings on the challenges for reliable data input throughout the Fourth Industrial Revolution, as described at the beginning of this chapter, are of relevance for the European Commission's European Digital Strategy [265] and in particular for Unit A.2 on *Technologies and Systems for Digitising Industry* [266] of the Commission's department Directorate-General for Communications Networks, Content and Technology (DG Connect).

Our findings in chapter 7 on investigation methodologies for human error in accidents and incidents are of relevance for authorities overseeing such incident investigations, such as the European Network of Civil Aviation Safety Investigation

Authorities [267] and the European Maritime Safety Agency [268].

Data protection issues related to the delete and erase functionalities in Windows 10 and macOS 10.14, as presented in chapter 3, are of interest to the relevant regulatory agencies and data protection authorities such as the European Data Protection Board [269] and domestic authorities, e.g. the UK's Information Commissioner's Office [270], Germany's Federal Commissioner for Data Protection and Freedom of Information [271], and Germany's 16 state-wide Data Protection Agencies.

Our findings on gross negligence in online banking, presented in chapter 4, are relevant for the UK Parliament's Treasury Select Committee [272] and the Financial Conduct Authority [273], as this lays the groundwork for and, thereby, is an important step towards an empirical definition of gross negligence that the Treasury Select Committee has requested [207] from the Financial Conduct Authority.

Our findings on insecurity with the Security Code AutoFill mechanism for financial transactions secured via mechanisms that rely on security codes delivered via SMS, presented in chapter 6, are interesting to the UK's Financial Conduct Authority [273], in particular with respect to financial fraud and the dispute resolution mechanism of the Financial Ombudsman Service [274].

8.3 Possible future research

In this section, we propose several avenues of possible future research that could be based on or motivated by the work presented in this thesis.

8.3.1 Efficacy of IT tools used by laypeople

In chapter 3, we found that the functionality to remove data from data storage devices, implemented in the two most popular Operating Systems for desktop and laptop computers, might be a major contributor to accidental data breaches when people fail to erase data before decommissioning their devices. Future work could aim at identifying and evaluating other relevant tools commonly used by laypeople for important tasks, in particular those that are a long-existing feature rather than a novel product on their own.

The first step would be to identify relevant tools. Recall that our work in chapter 3 was largely motivated by a long history of apparently accidental data breaches from decommissioned storage devices reported in the academic literature as well as the UK news media. We believe that this could serve as inspiration for a structured approach to identify other harms caused by human error for which not all latent failures seem to have been identified yet: A systematic literature review and news media research. Such an approach could be effective and efficient at identifying a subset of all relevant tools.

Alternatively, one could conduct a panel with domain experts or an online survey with the general population. This likely would identify more tools that are used by laypeople than the aforementioned literature review and media research. On the other side, it would be less likely to unveil the prevalence of any harms caused by latent failures within those tools – at least without additional research into the identified tools – and would likely contain a number of tools that are well-designed for their purpose. Thus, the alternative approach would be more likely to yield a comprehensive list of relevant tools but might be less efficient.

The next step would be the evaluation of those tools for the relevant purpose(s). Cognitive Walkthroughs, which we used in chapter 3, or other usability inspection methods such as those mentioned by Nielsen [101] could be suitable methods, in particular when the existence of the related harms has been established through literature review or new media research and when quantification of the risk of human error is not desired.

In other cases, user studies could be considered necessary to confirm the prevalence of otherwise theoretical harms from human error or to quantify the risks of human error. Quantification with small confidence intervals might be infeasible for a comprehensive study into such tools due to larger sample sizes – common rules of thumb estimate most studies will need between 20 [275, page 8] and 300 [276, page 228] participants. If the goal is solely to confirm that a theoretical risk does manifest, the requirements on the sample size can be further relaxed to 10-15 participants per tools [277], which is likely to reveal the vast majority of usability issues at least once.

We also argue that any criticism of existing tools should be accompanied by an exploration of how to remedy the identified issues. There might be cases in which identified issues are inherent in the tool's context of use and are, thus, part of the design constraints. Such issues could not be remedied by (solely) changing the design of the tool and such a case would not warrant criticism of the tool's design.

8.3.2 Extending the transaction authentication personas

In chapter 4, we presented the first transaction authentication personas for online banking but they were limited by the data set from which they were derived. Future work could extend these personas to other geographic areas with different cultures and other online banking implementations either through modifications to existing personas or the creation of new personas. See Kiljan *et al.* [278] for an overview of differences in security technologies used for online banking by 80 banks around the globe.

In addition, recent developments in the payment landscape create novel interactions that could be relevant for inclusion in the persona generation to provide a holistic perspective on the different user behaviours and needs that may affect transaction authentication technologies. For example, South African bank Absa introduced banking via WhatsApp and Facebook Messenger [279], cryptocurrencies promise anonymous payments [280], WeChat Pay and Alipay brought peer-to-peer payment systems to success as mass-market products [281], and Payment Initiation Service Providers in Europe strive to unify access to all of a user's bank accounts and activities on one platform [282].

Lastly, future research could aim at quantifying the coverage of the identified personas. At present, the three personas presented in this thesis are well-grounded in qualitative user data and can, thus, be considered validated [218]. Yet, it is unclear how much coverage they achieve. Quantitative measures could not only provide a better understanding of this coverage and, thereby, identify significant gaps that indicate characteristics of user clusters worth their own persona representation, but could also verify the extent to which these personas are transferable to other geographic and cultures.

8.3.3 Validating the transaction authentication mechanism

We proposed a novel transaction authentication mechanism in chapter 4 and presented a low-fidelity prototype, including a theoretical analysis of its usability and security properties. Future work could evaluate this mechanism in a user study to better assess its usability, either independently or in comparison with established mechanisms. Such assessments could also include a comparison between different implementation details of the proposed mechanism, e.g. horizontal vs vertical camera alignment.

The requirements for resistance against adversarial samples, as described in section 5.4.3.1, would likely require certain trade-offs for a high level of security that are sub-optimal from a usability perspective. Recall that for this purpose, specialised fonts were discussed as particularly resistant against adversarial samples. A dedicated user study (e.g. participatory design or survey) could compare applicable fonts for their suitability (e.g. readability and user acceptance). Alternatively, future research could develop new fonts that are optimised to this use case. As a side effect, such fonts might then also be helpful for other research areas such as on signage in public spaces for better orientation and navigation, e.g. for self-driving cars [283, Section 3.5.2.] or visually impaired people [284].

Furthermore, we discussed the benefit of a conservative approach to character recognition for the security of the proposed mechanism, i.e. the rejection of low-confidence classification. Yet, we also discussed that benign circumstances, such as screen glare or a dirty screen, could lower the classification confidence. A dedicated (session within a) user study could survey user satisfaction and acceptance rates for different thresholds on the required classification confidence.

8.3.4 Human error incidents in consumer-facing technologies

In chapter 7, we discussed shortcomings in the existing Swiss Cheese model on human error and the derived Human Factors Analysis and Classification System to explain some latent failures within the causal sequence of events that can lead to human error in the operation of consumer-facing devices. Based on insights from our research, we proposed to extend the category of unsafe supervision to include relevant communication between a product vendor (or other perceived authorities) and the consumer. Future work could conduct a systematic investigation into the classification of human error in consumer-facing technologies.

One approach, analogous to the development of HFACS [254], could be to source relevant incident reports for evaluation, e.g. by an expert panel. Yet, publicly available incident reports in the context of consumer-facing IT is notoriously scarce since this is a largely unregulated domain. An alternative approach could be to conducted research based on a combination of news media reports and incidents described in academic literature. While this approach might yield an incomplete picture of such human error incidents, this could be compensated for with the expert knowledge of an evaluation committee covering a wide range of relevant backgrounds.

8.4 Epilogue

In this thesis, we were concerned with challenges for cyber-physical systems to ensure reliable data input in the context of the Fourth Industrial Revolution. We conducted case study investigations into three identified sub-challenges. This led us to the evaluation of commonly used operating systems for desktop and mobile platforms as well as the online banking implementations by several European banks. We found several shortcomings in the evaluated technologies and how they were being implemented in those platforms and proposed amendments to mitigate or resolve the identified issues - some of which have been implemented by the respective vendors. We learned valuable lessons about these challenges to reliable data input for the practitioners who transform their systems and processes into digital products and services, as well as for the policymakers and enforcement agencies who oversee such transformations on an individual as well as an industry-encompassing level.

Bibliography

[1] Wenhao Li, Mingyang Ma, Jinchen Han, Yubin Xia, Binyu Zang, Cheng-Kang Chu, and Tieyan Li. Building trusted path on untrusted device drivers for mobile devices. In *Proceedings of 5th Asia-Pacific Workshop on Systems*, pages 1–7, 2014.

[2] Matthias Lange and Steffen Liebergeld. Crossover: secure and usable user interface for mobile devices with multiple isolated os personalities. In *Proceedings of the 29th Annual Computer Security Applications Conference*, pages 249–257, 2013.

[3] Janis Danisevskis, Michael Peter, Jan Nordholz, Matthias Petschick, and Julian Vetter. Graphical user interface for virtualized mobile handsets. *IEEE S&P MoST*, 2015.

[4] Norman Feske and Christian Helmuth. A nitpicker's guide to a minimal-complexity secure gui. In *21st Annual Computer Security Applications Conference (ACSAC'05)*, pages 85–94. IEEE, 2005.

[5] Saba Eskandarian, Jonathan Cogan, Sawyer Birnbaum, Peh Chang Wei Brandon, Dillon Franke, Forest Fraser, Gaspar Garcia, Eric Gong, Hung T Nguyen, Taresh K Sethi, et al. Fidelius: Protecting user secrets from compromised browsers. In *2019 IEEE Symposium on Security and Privacy (SP)*, pages 264–280. IEEE, 2019.

[6] Nicolas Papernot, Patrick McDaniel, Somesh Jha, Matt Fredrikson, Z Berkay Celik, and Ananthram Swami. The limitations of deep learning in adversar-

ial settings. In *2016 IEEE European symposium on security and privacy (EuroS&P)*, pages 372–387. IEEE, 2016.

[7] Ursula von der Leyen. Shaping Europe's digital future, 2020. `https://ec.europa.eu/commission/presscorner/detail/en/ac_20_260` (Accessed: 20 March 2020).

[8] Klaus Schwab. *The fourth industrial revolution.* Currency, 2017.

[9] Min Xu, Jeanne M David, Suk Hi Kim, et al. The fourth industrial revolution: Opportunities and challenges. *International journal of financial research*, 9(2):90–95, 2018.

[10] Directorate-General for Internal Policies. Industry 4.0, 2016. `https://www.europarl.europa.eu/RegData/etudes/STUD/2016/570007/IPOL_STU(2016)570007_EN.pdf` (Accessed: 20 March 2020).

[11] Armando W Colombo, Stamatis Karnouskos, Okyay Kaynak, Yang Shi, and Shen Yin. Industrial cyberphysical systems: A backbone of the fourth industrial revolution. *IEEE Industrial Electronics Magazine*, 11(1):6–16, 2017.

[12] Malte Brettel, Niklas Friederichsen, Michael Keller, and Marius Rosenberg. How virtualization, decentralization and network building change the manufacturing landscape: An industry 4.0 perspective. *International journal of mechanical, industrial science and engineering*, 8(1):37–44, 2014.

[13] Ontario Ministry of the Solicitor General. Investigation into the emergency alerts sent on January 12, 2020, 2020. `https://www.mcscs.jus.gov.on.ca/english/Publications/InvestigationemergencyalertssentJanuary122020.html` (Accessed: 16 November 2020).

[14] Henning Kagermann, Johannes Helbig, Ariane Hellinger, and Wolfgang Wahlster. *Recommendations for implementing the strategic initiative IN-*

DUSTRIE 4.0: Securing the future of German manufacturing industry; final report of the Industrie 4.0 Working Group. Forschungsunion, 2013.

[15] Carmen Cuesta, Macarena Ruesta, David Tuesta, Pablo Urbiola, et al. The digital transformation of the banking industry. *BBVA Research (available at https://www. bbvaresearch. com/wp-content/uploads/2015/08/EN_Observatorio_Banca_Digital_vf3. pdf)*, 2015.

[16] Raymond De Roover. New interpretations of the history of banking. *Cahiers d'Histoire Mondiale. Journal of World History. Cuadernos de Historia Mundial*, 2(1):38, 1954.

[17] Eurostat. Individuals using the internet for internet banking, 2020. `https://ec.europa.eu/eurostat/databrowser/bookmark/0f82c893-95c8-4d7c-88d4-befbc5a5f419` (Accessed: 27 February 2020).

[18] European Parliament. European Parliament resolution of 17 May 2017 on FinTech: the influence of technology on the future of the financial sector (2016/2243(INI)), 2017. `https://www.europarl.europa.eu/doceo/document/TA-8-2017-0211_EN.html` (Accessed: 23 March 2020).

[19] European Commission. FinTech Action plan: For a more competitive and innovative European financial sector (COM/2018/0109 final), 2018. `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:52018DC0109` (Accessed: 23 March 2020).

[20] Alasdair Gilchrist. *Industry 4.0: the industrial internet of things*. Springer, 2016.

[21] Nitin Naik. Choice of effective messaging protocols for iot systems: Mqtt, coap, amqp and http. In *2017 IEEE international systems engineering symposium (ISSE)*, pages 1–7. IEEE, 2017.

[22] Andreas Gutmann and Mark Warner. Fight to Be Forgotten: Exploring the Efficacy of Data Erasure in Popular Operating Systems. In *Privacy Technologies and Policy*, pages 45–58, Cham, 2019. Springer International Publishing.

[23] Andreas Gutmann and Steven J Murdoch. Taken Out of Context: Security Risks with Security Code AutoFill in iOS & macOS. In *Who Are You?! Adventures in Authentication Workshop (WAY 2019)*. USENIX, 2019.

[24] ENISA and TeleTrusT. Guideline "state of the art". Technical Report V 1.6_2020-01_EN, European Union Agency for Network and Information Security (ENISA), 2020.

[25] European Patent Office. The European Patent Convention, 2016. `http://www.epo.org/law-practice/legal-texts/html/epc/2016/e/ar54.html` (Accessed: 15 May 2020).

[26] Don Norman. *The design of everyday things: Revised and expanded edition*. Basic books, 2013.

[27] James Reason. *Human error*. Cambridge university press, 1990.

[28] David D Woods. *Behind human error*. Ashgate Publishing, Ltd., 2010.

[29] Neville A Stanton and Christopher Baber. Error by design: methods for predicting device usability. *Design Studies*, 23(4):363–384, 2002.

[30] Chauncey Wilson. *User experience re-mastered: your guide to getting the right design*. Morgan Kaufmann, 2009.

[31] Jakob Nielsen. Usability 101: Introduction to usability. 2012.

[32] Patrick W Jordan. *An introduction to usability*. CRC Press, 1998.

[33] Antoine Lemay, Joan Calvet, François Menet, and José M Fernandez. Survey of publicly available reports on advanced persistent threat actors. *Computers & Security*, 72:26–59, 2018.

[34] Matt Bishop and Carrie Gates. Defining the insider threat. In *Proceedings of the 4th annual workshop on Cyber security and information intelligence research: developing strategies to meet the cyber security and information intelligence challenges ahead*, pages 1–3, 2008.

[35] Marc Waldman and David Mazieres. Tangler: a censorship-resistant publishing system based on document entanglements. In *Proceedings of the 8th ACM conference on Computer and Communications Security*, pages 126–135, 2001.

[36] Isabel Lopez-Neira, Trupti Patel, Simon Parkin, George Danezis, and Leonie Tanczer. 'internet of things': How abuse is getting smarter. 2019.

[37] Kumar Chellapilla and Alexey Maykov. A taxonomy of javascript redirection spam. In *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, pages 81–88. ACM, 2007.

[38] Ross Anderson. Why cryptosystems fail. In *Proceedings of the 1st ACM Conference on Computer and Communications Security*, pages 215–227, 1993.

[39] Ross Anderson. *Security engineering*. John Wiley & Sons, 2008.

[40] Platon Kotzias, Leyla Bilge, and Juan Caballero. Measuring PUP prevalence and PUP distribution through pay-per-install services. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 739–756, 2016.

[41] Kurt Thomas, Juan A Elices Crespo, Ryan Rasti, Jean-Michel Picod, Cait Phillips, Marc-André Decoste, Chris Sharp, Fabio Tirelo, Ali Tofigh, Marc-Antoine Courteau, et al. Investigating commercial pay-per-install and the distribution of unwanted software. In *25th USENIX Security Symposium (USENIX Security 16)*, pages 721–739, 2016.

[42] Ahmed Aleroud and Lina Zhou. Phishing environments, techniques, and countermeasures: A survey. *Computers & Security*, 68:160–196, 2017.

[43] Luigi Mittone and Lucia Savadori. The scarcity bias. *Applied Psychology*, 58(3):453–468, 2009.

[44] Stanley Milgram. Behavioral study of obedience. *The Journal of abnormal and social psychology*, 67(4):371, 1963.

[45] Markus Jakobsson and Steven Myers. *Phishing and countermeasures: understanding the increasing problem of electronic identity theft*. John Wiley & Sons, 2006.

[46] Seung Yeob Nam, Dongwon Kim, and Jeongeun Kim. Enhanced arp: preventing arp poisoning-based man-in-the-middle attacks. *IEEE communications letters*, 14(2):187–189, 2010.

[47] Ramzi Bassil, Roula Hobeica, Wassim Itani, Cesar Ghali, Ayman Kayssi, and Ali Chehab. Security analysis and solution for thwarting cache poisoning attacks in the domain name system. In *2012 19th International Conference on Telecommunications (ICT)*, pages 1–6. IEEE, 2012.

[48] Lin Shung Huang, Alex Rice, Erling Ellingsen, and Collin Jackson. Analyzing forged ssl certificates in the wild. In *2014 IEEE Symposium on Security and Privacy*, pages 83–97. IEEE, 2014.

[49] Bruce Schneier. Two-factor authentication: too little, too late. *Communications of the ACM*, 48(4):136, 2005.

[50] Philipp Gühring. Concepts against man-in-the-browser attacks, 2006.

[51] Paul Black, Iqbal Gondal, and Robert Layton. A survey of similarities in banking malware behaviours. *Computers & Security*, 77:756–772, 2018.

[52] European Union Agency for Cybersecurity. Threat Landscape Report 2018, 2018. `https://www.enisa.europa.eu/publications/enisa-threat-landscape-report-2018` (Accessed: 20 January 2020).

[53] Ian Goldberg, David Wagner, Randi Thomas, Eric A Brewer, et al. A secure environment for untrusted helper applications: Confining the wily hacker. In *Proceedings of the 6th conference on USENIX Security Symposium, Focusing on Applications of Cryptography*, volume 6, pages 1–1, 1996.

[54] Rubin Xu, Hassen Saïdi, and Ross Anderson. Aurasium: Practical policy enforcement for android applications. In *Presented as part of the 21st USENIX Security Symposium (USENIX Security 12)*, pages 539–552, 2012.

[55] Michael Backes, Sven Bugiel, Christian Hammer, Oliver Schranz, and Philipp von Styp-Rekowsky. Boxify: Full-fledged app sandboxing for stock android. In *24th USENIX Security Symposium (USENIX Security 15)*, pages 691–706, 2015.

[56] Hajime Inoue and Stephanie Forrest. Anomaly intrusion detection in dynamic execution environments. In *Proceedings of the 2002 workshop on New security paradigms*, pages 52–60, 2002.

[57] Carsten Willems, Thorsten Holz, and Felix Freiling. Toward automated dynamic malware analysis using cwsandbox. *IEEE Security & Privacy*, 5(2):32–39, 2007.

[58] Adam Barth, Collin Jackson, Charles Reis, TGC Team, et al. The security architecture of the chromium browser. In *Technical report*. Stanford University, 2008.

[59] Moritz Lipp, Michael Schwarz, Daniel Gruss, Thomas Prescher, Werner Haas, Anders Fogh, Jann Horn, Stefan Mangard, Paul Kocher, Daniel Genkin, et al. Meltdown: Reading kernel memory from user space. In *27th USENIX Security Symposium (USENIX Security 18)*, pages 973–990, 2018.

[60] Donald C Latham. Department of Defense Trusted Computer System Evaluation Criteria. *Department of Defense*, 1986.

[61] Amit Vasudevan, Emmanuel Owusu, Zongwei Zhou, James Newsome, and Jonathan M McCune. Trustworthy execution on mobile devices: What security properties can my mobile platform give me? In *International Conference on Trust and Trustworthy Computing*, pages 159–178. Springer, 2012.

[62] David Cerdeira, Nuno Santos, Pedro Fonseca, and Sandro Pinto. Sok: Understanding the prevailing security vulnerabilities in trustzone-assisted tee systems. In *Proceedings of the IEEE Symposium on Security and Privacy (S&P), San Francisco, CA, USA*, pages 18–20, 2020.

[63] Roland van Rijswijk-Deij and Erik Poll. Using trusted execution environments in two-factor authentication: comparing approaches. *Open Identity Summit 2013*, 2013.

[64] Zongwei Zhou, Virgil D Gligor, James Newsome, and Jonathan M McCune. Building verifiable trusted path on commodity x86 computers. In *2012 IEEE symposium on security and privacy*, pages 616–630. IEEE, 2012.

[65] Samuel Weiser and Mario Werner. Sgxio: Generic trusted i/o path for intel sgx. In *Proceedings of the Seventh ACM on Conference on Data and Application Security and Privacy*, pages 261–268, 2017.

[66] JD Tygar and Alma Whitten. Www electronic commerce and java trojan horses. In *Proceedings of the 2nd USENIX Workshop on Electronic Commerce*, 1996.

[67] Jonathan S Shapiro, John Vanderburgh, Eric Northup, and David Chizmadia. Design of the eros trusted window system. In *USENIX Security Symposium*, pages 165–178, 2004.

[68] Zishuang Ye, Sean Smith, and Denise Anthony. Trusted paths for browsers. *ACM Transactions on Information and System Security (TISSEC)*, 8(2):153–186, 2005.

[69] Rebecca S Portnoff, Linda N Lee, Serge Egelman, Pratyush Mishra, Derek Leung, and David Wagner. Somebody's watching me? assessing the effectiveness of webcam indicator lights. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 1649–1658, 2015.

[70] Stuart E Schechter, Rachna Dhamija, Andy Ozment, and Ian Fischer. The emperor's new security indicators. In *2007 IEEE Symposium on Security and Privacy (SP'07)*, pages 51–65. IEEE, 2007.

[71] Adrienne Porter Felt, Robert W Reeder, Alex Ainslie, Helen Harris, Max Walker, Christopher Thompson, Mustafa Embre Acer, Elisabeth Morant, and Sunny Consolvo. Rethinking connection security indicators. In *Twelfth Symposium on Usable Privacy and Security (SOUPS 2016)*, pages 1–14, 2016.

[72] Mary Ellen Zurko. User-centered security: Stepping up to the grand challenge. In *21st Annual Computer Security Applications Conference (ACSAC'05)*, pages 14–pp. IEEE, 2005.

[73] Cristian Bravo-Lillo, Lorrie Cranor, Julie Downs, Saranga Komanduri, Stuart Schechter, and Manya Sleeper. Operating system framed in case of mistaken identity: measuring the success of web-based spoofing attacks on os password-entry dialogs. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 365–377, 2012.

[74] Grzergor Milka. Anatomy of account takeover. In *Enigma 2018 (Enigma 2018)*, 2018.

[75] Periwinkle Doerfler, Kurt Thomas, Maija Marincenko, Juri Ranieri, Yu Jiang, Angelika Moscicki, and Damon McCoy. Evaluating login challenges as adefense against account takeover. In *The World Wide Web Conference*, pages 372–382, 2019.

[76] Ariana Mirian, Joe DeBlasio, Stefan Savage, Geoffrey M Voelker, and Kurt Thomas. Hack for hire: Exploring the emerging market for account hijacking. In *The World Wide Web Conference*, pages 1279–1289, 2019.

[77] Javelin Strategy & Research. 2017 State of Authentication Report, 2017. `https://fidoalliance.org/wp-content/uploads/The-State-of-Authentication-Report.pdf` (Accessed: 4 February 2019).

[78] Josh Davis. Two Factor Auth List. `https://twofactorauth.org` (Accessed: 4 February 2019.

[79] Manal Adham, Amir Azodi, Yvo Desmedt, and Ioannis Karaolis. How to attack two-factor authentication internet banking. In *International Conference on Financial Cryptography and Data Security*. Springer, 2013.

[80] Steven J Murdoch and Ross Anderson. Verified by visa and mastercard securecode: or, how not to design authentication. In *International Conference on Financial Cryptography and Data Security*, pages 336–342. Springer, 2010.

[81] Chris Paget and Karsten Nohl. Gsm: Srsly. In *26th Chaos Communication Congress*, 2009.

[82] Adrienne Porter Felt, Matthew Finifter, Erika Chin, Steve Hanna, and David Wagner. A survey of mobile malware in the wild. In *Proceedings of the 1st ACM workshop on Security and privacy in smartphones and mobile devices*, pages 3–14. ACM, 2011.

[83] Positive Technologies. SS7 vulnerabilities and attack exposure report, 2018, 2018. `https://www.ptsecurity.com/ww-en/analytics/ss7-vulnerability-2018/` (Accessed: 4 February 2019).

[84] Slowe, Christopher. We had a security incident. Here's what you need to know., 2018. `https://www.reddit.com/r/announcements/comments/`

93qnm5/we_had_a_security_incident_heres_what_you_need_to/ (Accessed: 4 February 2019).

[85] Chavez-Dreyfuss, Gertrude. U.S. investor sues AT&T for $224 million over loss of cryptocurrency, 2018. https://www.reuters.com/article/idUSKBN1L01AA (Accessed: 4 February 2019).

[86] EY. IAPP-EY Annual Privacy Governance Report 2018. Technical report, International Association of Privacy Professionals, 2018. https://iapp.org/media/pdf/resource_center/IAPP-EY-Gov_Report_2018-FINAL.pdf (Accessed: 21 December 2018).

[87] ICO. Guide to the General Data Protection Regulation (GDPR), 2018. https://ico.org.uk/for-organisations/guide-to-data-protection/guide-to-the-general-data-protection-regulation-gdpr/individual-rights/right-to-erasure/ (Accessed: 21 December 2018).

[88] Simson L Garfinkel and Abhi Shelat. Remembrance of data passed: A study of disk sanitization practices. *IEEE Security & Privacy*, 1(1):17–27, 2003.

[89] Craig Valli and Andrew Jones. A UK and Australian Study of Hard Disk Disposal. 2005.

[90] Andy Jones, Craig Valli, and G Dabibi. The 2009 analysis of information remaining on USB storage devices offered for sale on the second hand market. In *Australian Digital Forensics Conference*, page 61, 2009.

[91] Andy Jones, Glenn S Dardick, Gareth Davies, Iain Sutherland, and Craig Valli. The 2008 Analysis of Information Remaining on Disks Offered for Sale on the Second Hand Market. *Journal of International Commercial Law and Technology*, 4(3), 2009.

[92] Tim Storer, Wm Bradley Glisson, and George Grispos. Investigating information recovered from re-sold mobile devices. In *Privacy and Usability Methods Pow-wow (PUMP) Workshop. ACM, University of Abertay, Dundee*, page 2, 2010.

[93] Nikki Robins, Patricia AH Williams, and Krishnun Sansurooah. An investigation into remnant data on USB storage devices sold in Australia creating alarming concerns. *International Journal of Computers and Applications*, 39(2):79–90, 2017.

[94] S Diesburg, CA Feldhaus, M Al Fardan, J Schlicht, and N Ploof. Is your data gone?: measuring user perceptions of deletion. In *Proceedings of the 6th Workshop on Socio-Technical Aspects in Security and Trust*, pages 47–59. ACM, 2016.

[95] Martin Westman. eMMC Chip Off – Benefits and Risks Workshop, 2017. `https://www.dfrws.org/conferences/dfrws-eu-2017/sessions/emmc-chip-benefits-and-risks-workshop` (Accessed: 21 December 2018).

[96] Martin Westman. Where Did That Incriminating Evidence Come From?, 2018. `https://www.dfrws.org/conferences/dfrws-eu-2018/sessions/where-did-incriminating-evidence-come` (Accessed: 21 December 2018).

[97] Gordon Hughes. Tutorial on disk drive data sanitization. 2006.

[98] NCSC. Secure sanitisation of storage media, 2016. `https://www.ncsc.gov.uk/guidance/secure-sanitisation-storage-media` (Accessed: 12 November 2020).

[99] NetApplications.com. Operating System Market Share. `https://www.netmarketshare.com/operating-system-market-share.aspx` (Accessed: 21 December 2018).

[100] Thomas Mahatody, Mouldi Sagar, and Christophe Kolski. State of the art on the cognitive walkthrough method, its variants and evolutions. *Intl. Journal of Human–Computer Interaction*, 26(8):741–785, 2010.

[101] Jakob Nielsen. Usability inspection methods. In *Conference companion on Human factors in computing systems*, pages 413–414. ACM, 1994.

[102] John Rieman, Marita Franzke, and David Redmiles. Usability evaluation with the cognitive walkthrough. In *Conference companion on Human factors in computing systems*, pages 387–388. ACM, 1995.

[103] Rick Spencer. The streamlined cognitive walkthrough method, working around social constraints encountered in a software development company. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, pages 353–359. ACM, 2000.

[104] Craig RM McKenzie, Michael J Liersch, and Stacey R Finkelstein. Recommendations implicit in policy defaults. *Psychological Science*, 17(5):414–420, 2006.

[105] Judith Donath. *The social machine: designs for living online*. MIT Press, 2014.

[106] National Central Banks in the European Union. Payments and Settlement Systems Statistics, 2019. `http://sdw.ecb.europa.eu/browse.do?node=9691547` (Accessed: 27 February 2020).

[107] European Banking Federation. Banking in Europe: EBF publishes 2019 Facts & Figures, 2019. `https://www.ebf.eu/wp-content/uploads/2019/09/EBF_038266-Banking-in-Europe-2019-Facts-and-Figures-press-release-11-Septembe pdf` (Accessed: 3 March 2020).

[108] Oleg Kupreev, Tatyana Sidorina, Victor Chebyshev, and Vladimir Kuskov. Financial threats in H1 2019, 2019. https://securelist.com/financial-threats-in-h1-2019/91899/ (Accessed: 2 March 2020).

[109] National Cyber Security Centre. UK and US investigations into harmful international cyber campaigns, 2019. https://www.ncsc.gov.uk/news/uk-and-us-investigation-into-harmful-international-cyber-campaigns (Accessed: 2 March 2020).

[110] National Cyber Security Centre. Advisory: Trickbot, 2019. https://www.ncsc.gov.uk/news/trickbot-advisory (Accessed: 2 March 2020).

[111] ENISA. Flash note: EU cyber security agency ENISA; "High Roller" online bank robberies reveal security gaps, 2012. https://www.enisa.europa.eu/news/enisa-news/copy_of_eu-cyber-security-agency-enisa-201chigh-roller201d-online-bank-robberies-reveal-se (Accessed: 15 May 2020).

[112] European Central Bank. Final recommendations for the security of payment account access services following the public consultation, 2014. https://www.zentral-bank.eu/pub/pdf/other/pubconsultationoutcome201405securitypaymentaccountaccessservicesen.pdf?3b8c24c7dc9fa5f57204d212c66f2dc7 (Accessed: 15 May 2020).

[113] European Parliament and Council of the European Union. Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015 on payment services in the internal market, 2015. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32015L2366.

[114] Vincent Haupert and Stephan Gabert. Where to look for what you see is what you sign? user confusion in transaction security. In *European Symposium on Research in Computer Security*, pages 429–449. Springer, 2019.

[115] Which? Consumer safeguards in the market for push payments – Which? super-complaint. https://www.which.co.uk/policy/347/ (Accessed: 15 January 2019).

[116] Which? Banks denying refunds to scam victims who ignore new warnings. https://www.which.co.uk/news/2020/01/banks-denying-refunds-to-scam-victims-who-ignore-new-warnings/ (Accessed: 19 May 2020).

[117] Ministerium für Ländlichen Raum und Verbraucherschutz Baden-Württemberg (MLR). Bezahlen im Internet. https://www.verbraucherportal-bw.de/Bezahlen+im+Internet (Accessed: 15 January 2019).

[118] Financial Conduct Authority. authorised push payment fraud – FCA Handbook. https://www.handbook.fca.org.uk/handbook/glossary/G3566a.html (Accessed: 19 May 2020).

[119] European Parliament and Council of the European Union. Directive 2011/83/EU of the European Parliament and of the Council of 25 October 2011 on consumer rights, 2011. https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex:32011L0083.

[120] Parliament of the United Kingdom. Consumer Credit Act 2006, SI 2006/14, 2006. https://www.legislation.gov.uk/ukpga/2006/14/contents.

[121] Parliament of the United Kingdom. The Payment Services Regulations 2017, SI 2017/752, 2017. http://www.legislation.gov.uk/uksi/2017/752/made.

[122] Deutscher Bundestag. Gesetz über die Beaufsichtigung von Zahlungsdiensten (Zahlungsdiensteaufsichtsgesetz - ZAG), 2018. https://www.gesetze-im-internet.de/zag_2018/.

[123] Deutscher Bundestag. Bürgerliches Gesetzbuch (BGB), 2019. https://www.gesetze-im-internet.de/bgb/.

[124] Deutscher Bundestag. Gesetz über das Kreditwesen (Kreditwesengesetz - KWG), 2020. https://www.gesetze-im-internet.de/kredwg/.

[125] Financial Conduct Authority. Payment Services and Electronic Money – Our Approach. https://www.fca.org.uk/publication/finalised-guidance/fca-approach-payment-services-electronic-money-sept-2017.pdf (Accessed: 14 January 2019).

[126] Bernd Lorenz. Sorgfaltspflichten im Umgang mit Passwörtern. *Datenschutz und Datensicherheit-DuD*, 37(4):220–226, 2013.

[127] Financial Ombudsman Service. Fraud And Scams: A Moving Picture. https://www.financial-ombudsman.org.uk/files/255900/Ombudsman-news-issue-145.pdf (Accessed: 14 January 2019).

[128] Marlene De Laine. *Fieldwork, participation and practice: Ethics and dilemmas in qualitative research.* Sage, 2000.

[129] Raymond M Lee. *Doing research on sensitive topics.* Sage, 1993.

[130] Mohammad Mannan and Paul C Van Oorschot. Security and usability: the gap in real-world online banking. In *Proceedings of the 2007 Workshop on New Security Paradigms*, pages 1–14, 2008.

[131] René M Dailey and Nicholas A Palomares. Strategic topic avoidance: An investigation of topic avoidance frequency, strategies used, and relational correlates. *Communication Monographs*, 71(4):471–496, 2004.

[132] Liezel Alsemgeest. Family communication about money: Why the taboo? *Mediterranean Journal of Social Sciences*, 5(16):516, 2014.

[133] Joan D Atwood. Couples and money: The last taboo. *The American Journal of Family Therapy*, 40(1):1–19, 2012.

[134] Richard Trachtman. The money taboo: Its effects in everyday life and in the practice of psychotherapy. *Clinical Social Work Journal*, 27(3):275–288, 1999.

[135] Kat Krol, Jonathan M Spring, Simon Parkin, and M Angela Sasse. Towards robust experimental design for user studies in security and privacy. In *The LASER Workshop: Learning from Authoritative Security Experiment Results (LASER 2016)*, pages 21–31, 2016.

[136] Shari Lawrence Pfleeger and Deanna D Caputo. Leveraging behavioral science to mitigate cyber security risk. *Computers & security*, 31(4):597–611, 2012.

[137] Mohammed AlZomai, Bander AlFayyadh, Audun Jøsang, and Adrian McCullagh. An exprimental investigation of the usability of transaction authorization in online bank security systems. In *Proceedings of the sixth Australasian conference on Information security-Volume 81*, pages 65–73. Australian Computer Society, Inc., 2008.

[138] Verena MIA Hartl and Ulrike Schmuntzsch. Fraud protection for online banking. In *International Conference on Human Aspects of Information Security, Privacy, and Trust*, pages 37–47. Springer, 2016.

[139] Kat Krol, Eleni Philippou, Emiliano De Cristofaro, and M Angela Sasse. ”They brought in the horrible key ring thing!” Analysing the Usability of Two-Factor Authentication in UK Online Banking. In *Workshop on Usable Security*, 2015.

[140] L Jean Camp. Mental models of privacy and security. *IEEE Technology and society magazine*, 28(3):37–46, 2009.

[141] Rick Wash. Folk models of home computer security. In *Proceedings of the Sixth Symposium on Usable Privacy and Security*, page 11. ACM, 2010.

[142] Karen Renaud and Melanie Volkamer. Exploring mental models underlying PIN management strategies. In *2015 World Congress on Internet Security (WorldCIS)*, pages 18–23. IEEE, 2015.

[143] Kelsey R Fulton, Rebecca Gelles, Alexandra McKay, Yasmin Abdi, Richard Roberts, and Michelle L Mazurek. The effect of entertainment media on mental models of computer security. In *Fifteenth Symposium on Usable Privacy and Security (SOUPS 2019)*, 2019.

[144] Ambar Murillo, Andreas Kramm, Sebastian Schnorf, and Alexander De Luca. “if i press delete, it’s gone”-user understanding of online data deletion and expiration. In *Fourteenth Symposium on Usable Privacy and Security (SOUPS 2018)*, pages 329–339, 2018.

[145] Ruba Abu-Salma, M Angela Sasse, Joseph Bonneau, Anastasia Danilova, Alena Naiakshina, and Matthew Smith. Obstacles to the adoption of secure communication tools. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 137–153. IEEE, 2017.

[146] Ruogu Kang, Laura Dabbish, Nathaniel Fruchter, and Sara Kiesler. “my data just goes everywhere:” user mental models of the internet and implications for privacy and security. In *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, pages 39–52, 2015.

[147] Kami E Vaniea, Emilee Rader, and Rick Wash. Betrayed by updates: how negative experiences affect future security. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2671–2674. ACM, 2014.

[148] Fahimeh Raja, Kirstie Hawkey, and Konstantin Beznosov. Revealing hidden context: improving mental models of personal firewall users. In *Proceedings of the 5th Symposium on Usable Privacy and Security*, page 1. ACM, 2009.

[149] Karen Renaud, Melanie Volkamer, and Arne Renkema-Padmos. Why doesn't jane protect her privacy? In *International Symposium on Privacy Enhancing Technologies Symposium*, pages 244–262. Springer, 2014.

[150] Leah Zhang-Kennedy, Sonia Chiasson, and Robert Biddle. Password advice shouldn't be boring: Visualizing password guessing attacks. In *2013 APWG eCrime Researchers Summit*, pages 1–11. IEEE, 2013.

[151] Fahimeh Raja, Kirstie Hawkey, Steven Hsu, Kai-Le Clement Wang, and Konstantin Beznosov. A brick wall, a locked door, and a bandit: a physical security metaphor for firewall warnings. In *Proceedings of the seventh symposium on usable privacy and security*, page 1. ACM, 2011.

[152] Rick Wash and Emilee Rader. Too much knowledge? security beliefs and protective behaviors among united states internet users. In *Eleventh Symposium On Usable Privacy and Security (SOUPS 2015)*, pages 309–325, 2015.

[153] Cristian Bravo-Lillo, Lorrie Faith Cranor, Julie Downs, and Saranga Komanduri. Bridging the gap in computer security warnings: A mental model approach. *IEEE Security & Privacy*, 9(2):18–26, 2010.

[154] Natalie Jones, Helen Ross, Timothy Lynam, Pascal Perez, and Anne Leitch. Mental models: An interdisciplinary synthesis of theory and methods. *Ecology and Society*, 16(1), 2011.

[155] Judith Reitman Olson and Henry H Rueter. Extracting expertise from experts: Methods for knowledge acquisition. *Expert systems*, 4(3):152–168, 1987.

[156] Robin S Grenier and Dana Dudzinska-Przesmitzki. A conceptual model for eliciting mental models using a composite methodology. *Human Resource Development Review*, 14(2):163–184, 2015.

[157] James M Royer, Cheryl A Cisero, and Maria S Carlo. Techniques and procedures for assessing cognitive skills. *Review of Educational Research*, 63(2):201–243, 1993.

[158] David H Jonassen. On the role of concepts in learning and instructional design. *Educational Technology Research and Development*, 54(2):177, 2006.

[159] Michelene TH Chi. Laboratory methods for assessing experts' and novices' knowledge. *The Cambridge handbook of expertise and expert performance*, pages 167–184, 2006.

[160] Donald A Norman. Some observations on mental models. In *Mental models*, pages 15–22. Psychology Press, 2014.

[161] Susan H Gray. Using protocol analyses and drawings to study mental model construction during hypertext navigation. *International Journal of Human-Computer Interaction*, 2(4):359–378, 1990.

[162] Eva-Maria Schomakers, Chantal Lidynia, and Martina Ziefle. Hidden within a group of people-mental models of privacy protection. In *IoTBDS*, pages 85–94, 2018.

[163] Christian Holz and Patrick Baudisch. Understanding touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2501–2510. ACM, 2011.

[164] Sondoss Elsawah, Joseph HA Guillaume, Tatiana Filatova, Josefine Rook, and Anthony J Jakeman. A methodology for eliciting, representing, and analysing stakeholder knowledge for decision making on complex socio-ecological systems: From cognitive maps to agent-based models. *Journal of environmental management*, 151:500–516, 2015.

[165] Cindy E Hmelo-Silver and Merav Green Pfeffer. Comparing expert and novice understanding of a complex system from the perspective of structures, behaviors, and functions. *Cognitive science*, 28(1):127–138, 2004.

[166] M Granger Morgan, Baruch Fischhoff, Ann Bostrom, Cynthia J Atman, et al. *Risk communication: A mental models approach.* Cambridge University Press, 2002.

[167] Alan J Bush and Joseph F Hair Jr. An assessment of the mall intercept as a data collection method. *Journal of Marketing Research*, 22(2):158–167, 1985.

[168] Kevin W Miller, Lora B Wilder, Frances A Stillman, and Diane M Becker. The feasibility of a street-intercept survey method in an african-american community. *American Journal of Public Health*, 87(4):655–658, 1997.

[169] Matthew B Miles, A Michael Huberman, Michael A Huberman, and Michael Huberman. *Qualitative data analysis: An expanded sourcebook.* sage, 1994.

[170] Paulien C Meijer, Nico Verloop, and Douwe Beijaard. Multi-method triangulation in a qualitative study on teachers' practical knowledge: An attempt to increase internal validity. *Quality and quantity*, 36(2):145–167, 2002.

[171] Serge Egelman and Eyal Peer. Scaling the security wall: Developing a security behavior intentions scale (sebis). In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 2873–2882. ACM, 2015.

[172] Patrick M Markey and Charlotte N Markey. A brief assessment of the interpersonal circumplex: The IPIP-IPC. *Assessment*, 16(4):352–361, 2009.

[173] Benjamin Saunders, Julius Sim, Tom Kingstone, Shula Baker, Jackie Waterfield, Bernadette Bartlam, Heather Burroughs, and Clare Jinks. Saturation in qualitative research: exploring its conceptualization and operationalization. *Quality & quantity*, 52(4):1893–1907, 2018.

[174] Greg Guest, Arwen Bunce, and Laura Johnson. How many interviews are enough? An experiment with data saturation and variability. *Field methods*, 18(1):59–82, 2006.

[175] Richard E Boyatzis. *Transforming qualitative information: Thematic analysis and code development*. Sage, 1998.

[176] Cynthia Weston, Terry Gandell, Jacinthe Beauchamp, Lynn McAlpine, Carol Wiseman, and Cathy Beauchamp. Analyzing interview data: The development and evolution of a coding system. *Qualitative sociology*, 24(3):381–400, 2001.

[177] American Psychological Association and others. Ethical Principles of Psychologists and Code of Conduct. *American psychologist*, 57(12):1060–1073, 2002.

[178] James Hedlund. Risky business: safety regulations, risk compensation, and individual behavior. *Injury prevention*, 6(2):82–89, 2000.

[179] Linda J Skitka, Kathleen L Mosier, and Mark Burdick. Does automation bias decision-making? *International Journal of Human-Computer Studies*, 51(5):991–1006, 1999.

[180] Mary T Dzindolet, Scott A Peterson, Regina A Pomranky, Linda G Pierce, and Hall P Beck. The role of trust in automation reliance. *International journal of human-computer studies*, 58(6):697–718, 2003.

[181] Saar Drimer, Steven J Murdoch, and Ross Anderson. Optimised to fail: Card readers for online banking. In *International Conference on Financial Cryptography and Data Security*, pages 184–200. Springer, 2009.

[182] Lennart Jaeger. Information security awareness: literature review and integrative framework. In *Proceedings of the 51st Hawaii International Conference on System Sciences*, 2018.

[183] European Parliament and Council of the European Union. Directive 2014/49/EU of the European Parliament and of the Council of 16 April 2014 on deposit guarantee schemes, 2014. `https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32014L0049`.

[184] Rajesh K Chandy, Gerard J Tellis, Deborah J MacInnis, and Pattana Thaivanich. What to say when: Advertising appeals in evolving markets. *Journal of marketing Research*, 38(4):399–414, 2001.

[185] Emmanuel Mogaji and Annie Danbury. Making the brand appealing: advertising strategies and consumers' attitude towards uk retail bank brands. *Journal of Product & Brand Management*, 2017.

[186] Marian Friestad and Esther Thorson. Emotion-eliciting advertising: Effects on long term memory and judgment. *ACR North American Advances*, 1986.

[187] Valerie S Folkes. The availability heuristic and perceived risk. *Journal of Consumer research*, 15(1):13–23, 1988.

[188] Norbert Schwarz, Herbert Bless, Fritz Strack, Gisela Klumpp, Helga Rittenauer-Schatka, and Annette Simons. Ease of retrieval as information: another look at the availability heuristic. *Journal of Personality and Social psychology*, 61(2):195, 1991.

[189] 'I lost my £193,000 inheritance - with one wrong digit on my sort code'. `https://www.theguardian.com/money/2019/dec/07/i-lost-my-193000-inheritance-with-one-wrong-digit-on-my-sort-code` (Accessed: 2 December 2020).

[190] Steven E Petersen and Michael I Posner. The attention system of the human brain: 20 years after. *Annual review of neuroscience*, 35:73–89, 2012.

[191] Nick Yeung, Leigh E Nystrom, Jessica A Aronson, and Jonathan D Cohen. Between-task competition and cognitive control in task switching. *Journal of Neuroscience*, 26(5):1429–1438, 2006.

[192] Duncan P Brumby, Anna L Cox, Jonathan Back, and Sandy JJ Gould. Recovering from an interruption: Investigating speed- accuracy trade-offs in task resumption behavior. *Journal of Experimental Psychology: Applied*, 19(2):95, 2013.

[193] Jonathan St BT Evans. Dual-processing accounts of reasoning, judgment, and social cognition. *Annu. Rev. Psychol.*, 59:255–278, 2008.

[194] Sarah Wiseman, Judith Borghouts, Dora Grgic, Duncan P Brumby, and Anna L Cox. The effect of interface type on visual error checking behavior. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 59, pages 436–439. SAGE Publications Sage CA: Los Angeles, CA, 2015.

[195] Anna L Cox, Sandy JJ Gould, Marta E Cecchinato, Ioanna Iacovides, and Ian Renfree. Design frictions for mindful interactions: The case for microboundaries. In *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, pages 1389–1397, 2016.

[196] Carol-anne Moulton, Glenn Regehr, Lorelei Lingard, Catherine Merritt, and Helen MacRae. 'slowing down when you should': initiators and influences of the transition from the routine to the effortful. *Journal of Gastrointestinal Surgery*, 14(6):1019–1026, 2010.

[197] Adam Beautement, M Angela Sasse, and Mike Wonham. The compliance budget: managing security behaviour in organisations. In *Proceedings of the 2008 New Security Paradigms Workshop*, pages 47–58, 2008.

[198] Anne Adams and Martina Angela Sasse. Users are not the enemy. *Communications of the ACM*, 42(12):40–46, 1999.

[199] Maria Bada, Angela M. Sasse, and Jason R. C. Nurse. Cyber security awareness campaigns: Why do they fail to change behaviour? *International Conference on Cyber Security for Sustainable Society*, 2015.

[200] Charles A O'Reilly III. Individuals and information overload in organizations: is more necessarily better? *Academy of management journal*, 23(4):684–696, 1980.

[201] Allen G Schick, Lawrence A Gordon, and Susan Haka. Information overload: A temporal approach. *Accounting, Organizations and Society*, 15(3):199–220, 1990.

[202] Shana K Carpenter, Nicholas J Cepeda, Doug Rohrer, Sean HK Kang, and Harold Pashler. Using spacing to enhance diverse forms of learning: Review of recent research and implications for instruction. *Educational Psychology Review*, 24(3):369–378, 2012.

[203] Carol-Anne E Moulton, Adam Dubrowski, Helen MacRae, Brent Graham, Ethan Grober, and Richard Reznick. Teaching surgical skills: what kind of practice makes perfect?: a randomized, controlled trial. *Annals of surgery*, 244(3):400, 2006.

[204] Ali Akdemir, Burak Zeybek, Ahmet M Ergenoglu, Ahmet O Yeniel, and Fatih Sendag. Effect of spaced training with a box trainer on the acquisition and retention of basic laparoscopic skills. *International Journal of Gynecology & Obstetrics*, 127(3):309–313, 2014.

[205] Shana K Carpenter, Harold Pashler, and Nicholas J Cepeda. Using tests to enhance 8th grade students' retention of us history facts. *Applied Cognitive Psychology: The Official Journal of the Society for Applied Research in Memory and Cognition*, 23(6):760–771, 2009.

[206] Nicholas J Cepeda, Edward Vul, Doug Rohrer, John T Wixted, and Harold Pashler. Spacing effects in learning: A temporal ridgeline of optimal retention. *Psychological science*, 19(11):1095–1102, 2008.

[207] House of Commons Treasury Committee. Economic Crime: Consumer View, 2019. `https://publications.parliament.uk/pa/cm201919/cmselect/cmtreasy/246/246.pdf` (Accessed: 2 July 2020).

[208] Jeff Conklin. Dialogue mapping. *Building Shared Understanding of Wicked Problems. West Sussex, England: John Wiley & Sons*, 2006.

[209] Grant Wiggins. Seven keys to effective feedback. *Feedback*, 70(1):10–16, 2012.

[210] Jamie C Brehaut, Heather L Colquhoun, Kevin W Eva, Kelly Carroll, Anne Sales, Susan Michie, Noah Ivers, and Jeremy M Grimshaw. Practice feedback interventions: 15 suggestions for optimizing effectiveness. *Annals of internal medicine*, 164(6):435–441, 2016.

[211] John Pruitt and Tamara Adlin. *The persona lifecycle: keeping people in mind throughout product design*. Elsevier, 2010.

[212] M Cameron Jones, Ingbert R Floyd, and Michael B Twidale. Teaching design with personas. 2008.

[213] Anke Dittmar and Maximilian Hensch. Two-level personas for nested design spaces. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 3265–3274. ACM, 2015.

[214] Ingbert R Floyd, M Cameron Jones, and Michael B Twidale. Resolving incommensurable debates: a preliminary identification of persona kinds, attributes, and characteristics. *Artifact*, 2(1):12–26, 2008.

[215] Jonathan Grudin and John Pruitt. Personas, participatory design and product development: An infrastructure for engagement. In *Proc. PDC*, volume 2, 2002.

[216] John Pruitt and Jonathan Grudin. Personas: practice and theory. In *Proceedings of the 2003 conference on Designing for user experiences*, pages 1–15. ACM, 2003.

[217] Tara Matthews, Tejinder Judge, and Steve Whittaker. How do designers and user experience professionals actually perceive and use personas? In *Proceedings of the SIGCHI conference on human factors in computing systems*, pages 1219–1228. ACM, 2012.

[218] Shamal Faily and Ivan Flechais. Persona cases: a technique for grounding personas. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 2267–2270. ACM, 2011.

[219] Amir Dotan, Neil Maiden, Valentina Lichtner, and Lola Germanovich. Designing with only four people in mind?–a case study of using personas to redesign a work-integrated learning support system. In *IFIP Conference on Human-Computer Interaction*, pages 497–509. Springer, 2009.

[220] Christopher N Chapman and Russell P Milham. The personas' new clothes: methodological and practical arguments against a popular method. In *Proceedings of the human factors and ergonomics society annual meeting*, volume 50, pages 634–636. SAGE Publications Sage CA: Los Angeles, CA, 2006.

[221] Rashmi Sinha. Persona development for information-rich domains. In *CHI'03 extended abstracts on Human factors in computing systems*, pages 830–831. ACM, 2003.

[222] Nan Tu, Xiao Dong, Pei-Luen Patrick Rau, and Tao Zhang. Using cluster analysis in persona development. In *2010 8th International Conference on Supply Chain Management and Information*, pages 1–5. IEEE, 2010.

[223] Sushmita Mitra, Sankar K Pal, and Pabitra Mitra. Data mining in soft computing framework: a survey. *IEEE transactions on neural networks*, 13(1):3–14, 2002.

[224] Eric WT Ngai, Li Xiu, and Dorothy CK Chau. Application of data mining techniques in customer relationship management: A literature review and classification. *Expert systems with applications*, 36(2):2592–2602, 2009.

[225] Patrício Soares Costa, Nadine Correia Santos, Pedro Cunha, Jorge Cotter, and Nuno Sousa. The use of multiple correspondence analysis to explore associations between categories of qualitative variables in healthy ageing. *Journal of aging research*, 2013, 2013.

[226] Maria Halkidi, Yannis Batistakis, and Michalis Vazirgiannis. On clustering validation techniques. *Journal of intelligent information systems*, 17(2-3):107–145, 2001.

[227] Jose M Cortina. What is coefficient alpha? an examination of theory and applications. *Journal of applied psychology*, 78(1):98, 1993.

[228] RA Johnson and DW Wichern. Applied multivariate correspondence analysis, 2007.

[229] Alan Baddeley. Exploring the central executive. *The Quarterly Journal of Experimental Psychology Section A*, 49(1):5–28, 1996.

[230] Arindam Chaudhuri, Krupa Mandaviya, Pratixa Badelia, and Soumya K Ghosh. Optical character recognition systems. In *Optical Character Recognition Systems for Different Languages with Soft Computing*, pages 9–41. Springer, 2017.

[231] Congzheng Song and Vitaly Shmatikov. Fooling ocr systems with adversarial text images. *arXiv preprint arXiv:1802.05385*, 2018.

[232] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna Estrach, Dumitru Erhan, Ian Goodfellow, and Robert Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations, ICLR 2014*, 2014.

[233] Wolfgang Köhler. Gestalt psychology. *Psychologische Forschung*, 31(1):XVIII–XXX, 1967.

[234] Anirban Chakraborty, Manaar Alam, Vishal Dey, Anupam Chattopadhyay, and Debdeep Mukhopadhyay. Adversarial attacks and defences: A survey. *arXiv preprint arXiv:1810.00069*, 2018.

[235] Alphanumeric character sets for optical recognition — Part 1: Character set OCR-A — Shapes and dimensions of the printed image. Standard, International Organization for Standardization, Geneva, CH, 1976.

[236] Alphanumeric character sets for optical recognition — Part 2: Character set OCR-B — Shapes and dimensions of the printed image. Standard, International Organization for Standardization, Geneva, CH, 1976.

[237] Information processing — Magnetic ink character recognition — Part 1: Print specifications for E13B. Standard, International Organization for Standardization, Geneva, CH, 2013.

[238] Information processing — Magnetic ink character recognition — Part 2: Print specifications for CMC7. Standard, International Organization for Standardization, Geneva, CH, 2013.

[239] Weilin Xu, David Evans, and Yanjun Qi. Feature Squeezing: Detecting Adversarial Examples in Deep Neural Networks. In *25th Annual Network and Distributed System Security Symposium, NDSS 2018, San Diego, California, USA, February 18-21, 2018*. The Internet Society, 2018.

[240] Åsa Blomquist and Mattias Arvola. Personas in action: ethnography in an interaction design team. In *Proceedings of the second Nordic conference on Human-computer interaction*, pages 197–200, 2002.

[241] Rosa Gudjonsdottir and Sinna Lindquist. Personas and scenarios: Design tool or a communication device? *From CSCW to Web 2.0: European Developments in Collaborative Design Selected Papers from COOP08*, 2008.

[242] Erin Friess. Personas and decision making in the design process: an ethnographic case study. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1209–1218, 2012.

[243] Martin Abadi and Roger Needham. Prudent engineering practice for cryptographic protocols. *IEEE transactions on Software Engineering*, (1):6–15, 1996.

[244] KR Laughery and MS Wogalter. Warnings and risk perception. *Handbook of Human Factors and Ergonomics, G. Salvendy (ed.), New York, NY: Wiley-Interscience*, 1997.

[245] Cormac Herley. So long, and no thanks for the externalities: the rational rejection of security advice by users. In *Proceedings of the 2009 workshop on New security paradigms workshop*, pages 133–144. ACM, 2009.

[246] Sampath Srinivas, Dirk Balfanz, Eric Tiffany, Alexi Czeskis, and F Alliance. Universal 2nd factor (u2f) overview. *FIDO Alliance Proposed Standard*, pages 1–5, 2015.

[247] Apple. Enhance SMS-delivered code security with domain-bound codes, 2020. `https://developer.apple.com/news/?id=z0i801mg` (Accessed: 21 August 2020).

[248] Apple. About the security content of watchOS 7.0, 2020. `https://support.apple.com/en-us/HT211844` (Accessed: 24 November 2020).

[249] GC Bello and V Colombari. The human factors in risk analyses of process plants: The control room operator model 'teseo'. *Reliability engineering*, 1(1):3–14, 1980.

[250] James Reason. The contribution of latent human failures to the breakdown of complex systems. *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, 327(1241):475–484, 1990.

[251] James Reason. *Managing the risks of organizational accidents*. Routledge, 2016.

[252] James Reason. Human error: models and management. *Bmj*, 320(7237):768–770, 2000.

[253] J Reason, E Hollnagel, and J Paries. Revisiting the swiss cheese model of accidents. *Journal of Clinical Engineering*, 27(4):110–115, 2006.

[254] Scott A Shappell and Douglas A Wiegmann. A human error approach to accident investigation: The taxonomy of unsafe operations. *The International Journal of Aviation Psychology*, 7(4):269–291, 1997.

[255] Douglas A Wiegmann and Scott A Shappell. Applying the human factors analysis and classification system (hfacs) to the analysis of commercial aviation accident data. 2001.

[256] Stephen Reinach and Alex Viale. Application of a human error framework to conduct train accident/incident investigations. *Accident Analysis & Prevention*, 38(2):396–406, 2006.

[257] Christine Chauvin, Salim Lardjane, Gael Morel, Jean-Pierre Clostermann, and Benoît Langard. Human and organisational factors in maritime accidents: Analysis of collisions at sea using the hfacs. *Accident Analysis & Prevention*, 59:26–37, 2013.

[258] Jessica M Patterson and Scott A Shappell. Operator error and system deficiencies: analysis of 508 mining incidents and accidents from queensland, australia using hfacs. *Accident Analysis & Prevention*, 42(4):1379–1385, 2010.

[259] Thomas Diller, George Helmrich, Sharon Dunning, Stephanie Cox, April Buchanan, and Scott Shappell. The human factors analysis classification system (hfacs) applied to health care. *American journal of medical Quality*, 29(3):181–190, 2014.

[260] Scott A Shappell and Douglas A Wiegmann. The human factors analysis and classification system–hfacs. 2000.

[261] Jonas Lundberg, Carl Rollenhagen, and Erik Hollnagel. What-you-look-for-is-what-you-find–the consequences of underlying accident models in eight accident investigation manuals. *Safety science*, 47(10):1297–1311, 2009.

[262] Kathryn Woodcock, Colin G Drury, Alison Smiley, and Jiao Ma. Using simulated investigations for accident investigation studies. *Applied ergonomics*, 36(1):1–12, 2005.

[263] Barry Strauch. *Investigating human error: Incidents, accidents, and complex systems*. CRC Press, 2017.

[264] Joni Salminen, Kathleen Guan, Soon-Gyo Jung, Shammur A Chowdhury, and Bernard J Jansen. A literature review of quantitative persona creation. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–14, 2020.

[265] European Commission. The European Digital Strategy, 2020. https://ec.europa.eu/digital-single-market/en/content/european-digital-strategy (Accessed: 26 November 2020).

[266] European Commission. Technologies and Systems for Digitising Industry (Unit A.2), 2020. https://ec.europa.eu/digital-single-market/en/content/technologies-and-systems-digitising-industry-unit-a2 (Accessed: 26 November 2020).

[267] European Commission. About the ENCASIA network, 2020. https://ec.europa.eu/transport/modes/air/encasia (Accessed: 26 November 2020).

[268] European Commission. European Maritime Safety Agency (EMSA), 2020. https://ec.europa.eu/transport/modes/maritime/emsa/emsa (Accessed: 26 November 2020).

[269] European Commission. European Data Protection Board, 2020. https://edpb.europa.eu (Accessed: 26 November 2020).

[270] Information Commissioner's Office. Home, 2020. https://ico.org.uk (Accessed: 26 November 2020).

[271] Der Bundesbeauftragte für den Datenschutz und die Informationsfreiheit. Internetauftritt des Bundesbeauftragten für den Datenschutz und die Informationsfreiheit, 2020. `https://www.bfdi.bund.de/` (Accessed: 26 November 2020).

[272] House of Commons Treasury Committee. Treasury Committee, 2020. `https://committees.parliament.uk/committee/158/treasury-committee` (Accessed: 26 November 2020).

[273] Financial Conduct Authority. Financial Conduct Authority, 2020. `https://www.fca.org.uk` (Accessed: 26 November 2020).

[274] Financial Ombudsman Service. Financial Ombudsman Service: our homepage, 2020. `https://www.financial-ombudsman.org.uk` (Accessed: 26 November 2020).

[275] Joseph L Fleiss. *Design and analysis of clinical experiments*, volume 73. John Wiley & Sons, 2011.

[276] Jum C Nunnally and Ira H Bernstein. Psychological theory, 1994.

[277] Laura Faulkner. Beyond the five-user assumption: Benefits of increased sample sizes in usability testing. *Behavior Research Methods, Instruments, & Computers*, 35(3):379–383, 2003.

[278] Sven Kiljan, Koen Simoens, Danny De Cock, Marko Van Eekelen, and Harald Vranken. A survey of authentication and communications security in online banking. *ACM Computing Surveys (CSUR)*, 49(4):1–35, 2016.

[279] Absa Bank Limited. Bank where you 'like' with ChatBanking, 2019. `https://www.absa.co.za/ways-to-bank/chat-banking/` (Accessed: 18 August 2020).

[280] Eli Ben Sasson, Alessandro Chiesa, Christina Garman, Matthew Green, Ian Miers, Eran Tromer, and Madars Virza. Zerocash: Decentralized anonymous

payments from bitcoin. In *2014 IEEE Symposium on Security and Privacy*, pages 459–474. IEEE, 2014.

[281] Tyler Aveni and Joep Roest. China's alipay and wechat pay. World Bank Other Operational Studies 30112, The World Bank, 2017.

[282] Peggy Valcke, Niels Vandezande, and Nathan Van De Velde. The evolution of third party payment providers and cryptocurrencies under the eu's upcoming psd2 and amld4. 2015.

[283] Claudine Badue, Rânik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius Brito Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago Meireles Paixão, Filipe Mutz, et al. Self-driving cars: A survey. *Expert Systems with Applications*, page 113816, 2020.

[284] Lilit Hakobyan, Jo Lumsden, Dympna O'Sullivan, and Hannah Bartlett. Mobile assistive technologies for the visually impaired. *Survey of ophthalmology*, 58(6):513–528, 2013.