

---

# SUPPORTING USERS IN PASSWORD AUTHENTICATION WITH PERSUASIVE DESIGN

---

## DISSERTATION

an der Fakultät für Mathematik, Informatik und Statistik  
der Ludwig-Maximilians-Universität München

vorgelegt von

**TOBIAS SEITZ**

M.Sc. Medieninformatik

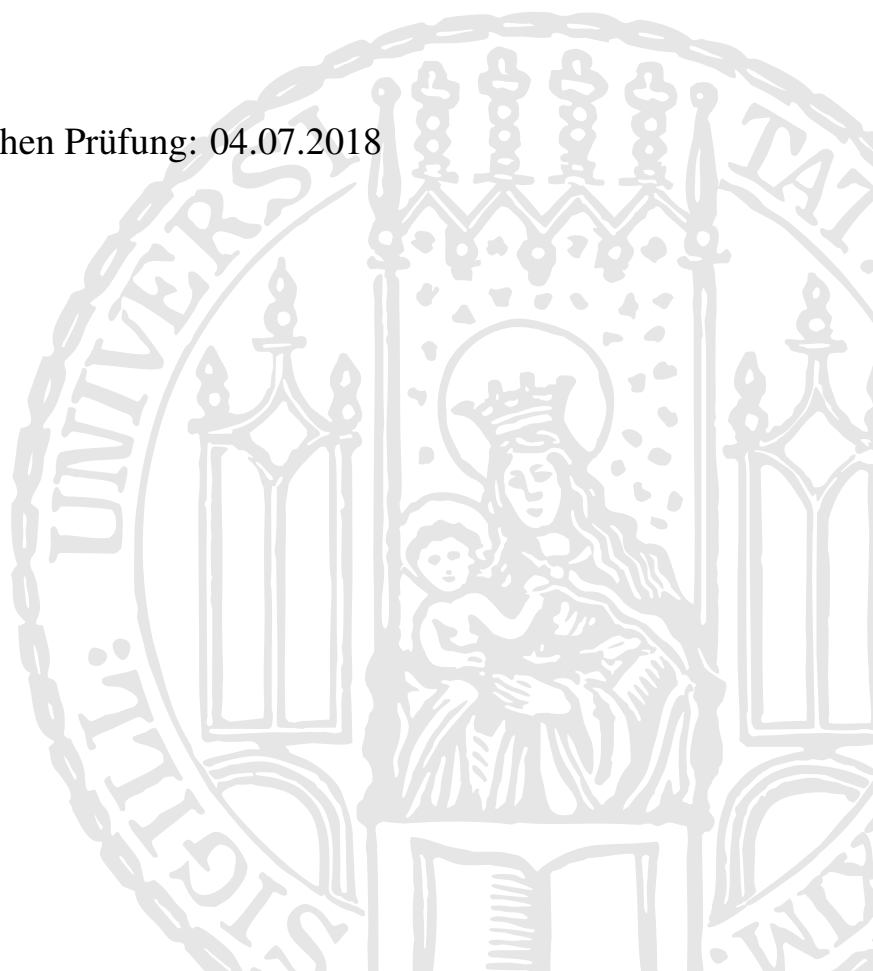
München, den 26.04.2018



---

Erstgutachter: Prof. Dr. Heinrich Hußmann  
Zweitgutachter: Prof. Dr. Konstantin Beznosov

Tag der mündlichen Prüfung: 04.07.2018



## ABSTRACT

Activities like text-editing, watching movies, or managing personal finances are all accomplished with web-based solutions nowadays. The providers need to ensure security and privacy of user data. To that end, passwords are still the most common authentication method on the web. They are inexpensive and easy to implement. Users are largely accustomed to this kind of authentication but passwords represent a considerable nuisance, because they are tedious to create, remember, and maintain. In many cases, usability issues turn into security problems, because users try to work around the challenges and create easily predictable credentials. Often, they reuse their passwords for many purposes, which aggravates the risk of identity theft. There have been numerous attempts to remove the root of the problem and replace passwords, e.g., through biometrics. However, no other authentication strategy can fully replace them, so passwords will probably stay a go-to authentication method for the foreseeable future.

Researchers and practitioners have thus aimed to improve users' situation in various ways. There are two main lines of research on helping users create both usable and secure passwords. On the one hand, password policies have a notable impact on password practices, because they enforce certain characteristics. However, enforcement reduces users' autonomy and often causes frustration if the requirements are poorly communicated or overly complex. On the other hand, user-centered designs have been proposed: Assistance and persuasion are typically more user-friendly but their influence is often limited. In this thesis, we explore potential reasons for the inefficacy of certain persuasion strategies. From the gained knowledge, we derive novel persuasive design elements to support users in password authentication.

The exploration of contextual factors in password practices is based on four projects that reveal both psychological aspects and real-world constraints. Here, we investigate how mental models of password strength and password managers can provide important pointers towards the design of persuasive interventions. Moreover, the associations between personality traits and password practices are evaluated in three user studies. A meticulous audit of real-world password policies shows the constraints for selection and reuse practices.

Based on the review of context factors, we then extend the design space of persuasive password support with three projects. We first depict the explicit and implicit user needs in password support. Second, we craft and evaluate a choice architecture that illustrates how a phenomenon from marketing psychology can provide new insights into the design of nudging strategies. Third, we tried to empower users to create memorable passwords with emojis. The results show the challenges and potentials of emoji-passwords on different platforms.

Finally, the thesis presents a framework for the persuasive design of password support. It aims to structure the required activities during the entire process. This enables researchers and practitioners to craft novel systems that go beyond traditional paradigms, which is illustrated by a design exercise.

---

## ZUSAMMENFASSUNG

Heutzutage ist es möglich, mit web-basierten Lösungen Texte zu editieren, Filme anzusehen, oder seine persönlichen Finanzen zu verwalten. Die Anbieter müssen hierbei Sicherheit und Vertraulichkeit von Nutzerdaten sicherstellen. Dazu sind Passwörter weiterhin die geläufigste Authentifizierungsmethode im Internet. Sie sind kostengünstig und einfach zu implementieren. NutzerInnen sind bereits im Umgang mit diesem Verfahren vertraut jedoch stellen Passwörter ein beträchtliches Ärgernis dar, weil sie mühsam zu erstellen, einzuprägen, und verwalten sind. Oft werden Usabilityfragen zu Sicherheitsproblemen, weil NutzerInnen Herausforderungen umschiffen und sich einfach zu erratende Zugangsdaten ausdenken. Daneben verwenden sie Passwörter für viele Zwecke wieder, was das Risiko eines Identitätsdiebstals weiter erhöht. Es gibt zahlreiche Versuche die Wurzel des Problems zu beseitigen und Passwörter zu ersetzen, z.B. mit Biometrie. Jedoch kann bisher kein anderes Verfahren sie vollkommen ersetzen, so dass Passwörter wohl für absehbare Zeit die Hauptauthentifizierungsmethode bleiben werden.

ExpertInnen aus Forschung und Industrie haben sich deshalb zum Ziel gefasst, die Situation der NutzerInnen auf verschiedene Wege zu verbessern. Es existieren zwei Forschungsstränge darüber wie man NutzerInnen bei der Erstellung von sicheren und benutzbaren Passwörtern helfen kann. Auf der einen Seite haben Regeln bei der Passwörterstellung deutliche Auswirkungen auf Passwortpraktiken, weil sie bestimmte Charakteristiken durchsetzen. Jedoch reduziert diese Durchsetzung die Autonomie der NutzerInnen und verursacht Frustration, wenn die Anforderungen schlecht kommuniziert oder übermäßig komplex sind. Auf der anderen Seite stehen nutzerzentrierte Designs: Hilfestellung und Überzeugungsarbeit sind typischerweise nutzerfreundlicher wobei ihr Einfluss begrenzt ist. In dieser Arbeit erkunden wir die potenziellen Gründe für die Ineffektivität bestimmter Überzeugungsstrategien. Von dem hierbei gewonnenen Wissen leiten wir neue persuasive Designelemente für Hilfestellung bei der Passwortauthentifizierung ab.

Die Exploration von Kontextfaktoren im Umgang mit Passwörtern basiert auf vier Projekten, die sowohl psychologische Aspekte als auch Einschränkungen in der Praxis aufdecken. Hierbei untersuchen wir inwiefern Mental Modelle von Passwortstärke und -managern wichtige Hinweise auf das Design von persuasiven Interventionen liefern. Darüber hinaus werden die Zusammenhänge zwischen Persönlichkeitsmerkmalen und Passwortpraktiken in drei Nutzerstudien untersucht. Eine gründliche Überprüfung von Passwortregeln in der Praxis zeigt die Einschränkungen für Passwortselektion und -wiederverwendung.

Basierend auf der Durchleuchtung der Kontextfaktoren erweitern wir hierauf den Design-Raum von persuasiver Passworthilfestellung mit drei Projekten. Zuerst schildern wir die expliziten und impliziten Bedürfnisse in punkto Hilfestellung. Daraufhin erstellen und evaluieren wir eine Entscheidungsarchitektur, welche veranschaulicht wie ein Phänomen aus der Marketingpsychologie neue Einsichten in das Design von Nudging-Strategien liefern kann. Im Schlussgang versuchen wir NutzerInnen dabei zu stärken, gut merkbare Passwörter mit

Hilfe von Emojis zu erstellen. Die Ergebnisse zeigen die Herausforderungen und Potenziale von Emoji-Passwörtern auf verschiedenen Plattformen.

Zuletzt präsentiert diese Arbeit ein Rahmenkonzept für das persuasive Design von Passworthilfestellungen. Es soll die benötigten Aktivitäten während des gesamten Prozesses strukturieren. Dies erlaubt ExpertInnen neuartige Systeme zu entwickeln, die über traditionelle Ansätze hinausgehen, was durch eine Designstudie veranschaulicht wird.



# Disclaimer

## Personal Contribution Statement

In this thesis, I discuss projects that I carried out in collaboration with several students and colleagues. Most of them were accomplished as part of the bachelor or master theses of talented students that I was lucky to have supervised. In the following, I declare my personal contribution to each of these projects.

**Chapter 6** is based on project work for the course Advanced Topics in HCI (in 2016). I developed the idea and project scope, and the work was conducted in collaboration with Manuel Hartmann, Jakob Pfab, and Samuel Souque. In regular meetings, each step was jointly discussed and agreed upon. I set the general course, and the three students carried out the audit of the policies. After the initial analysis, I evaluated further details of the dataset. The work was published as an extended abstract at the *ACM CHI Conference on Human Factors in Computing Systems* (CHI '17) [288] with Manuel, Jakob, and Samuel as co-authors. I reworked their initial draft several times and revised the paper based on the reviewers' feedback.

**Chapter 7** reports on three studies that were carried out as part of the bachelor theses of three students: Timo Erdelt [101], Paul Huber [168], and Aline Neumann [241]. I developed the idea to study personality in the domain of password authentication. In regular meetings, we discussed each step. However, I provided guidance and was responsible for the key decisions about the study design, execution, and evaluations. For statistical analyses, we also consulted the LMU's internal statistic consultancy (StabLab), as well as Clemens Stachl from the psychology department. I validated and analyzed the data set independently from the students.

**Chapter 8** is based on a project that was part of Martin Prinz' Master thesis [256]. I developed the research questions, project roadmap, and methodology. Martin then carried out the interviews and created a large part of the mental model structure, whereas I connected the dots in broader analyses. We met regularly to discuss the progress. The results were published as an extended abstract at the Symposium on Usable Privacy and Security (SOUPS '17) [257]. I revised Martin's draft in several iterations and redacted the paper based on the reviewers' feedback.

**Chapter 9** encompasses two projects from the bachelor theses of Caroline Olsienkiewicz [248] and Katharina Schwarz [284]. For both projects, I developed the initial ideas, provided guidance, and made the key decisions about the scope, methodology, and analyses. The specifics were always jointly discussed and agreed upon. Both Caroline and Katharina crafted wireframes, implemented prototypes, and executed the studies. I analyzed the data independently and derived broader concepts and paradigms.

**Chapter 10** is partially based on Stefanie Meitner's bachelor thesis [231]. I had already developed the concepts and especially the choice architectures (in part with Isabel Schönewald) before Stefanie joined the project. I provided guidance, and defined the project scope and

---

methodology. Stefanie implemented the prototype and executed the data collection before I performed the data analyses. The results have been published at the European Workshop on Usable Security (EuroUSEC '16) with Emanuel von Zezschwitz, Stefanie Meitner, and Heinrich Hußmann as co-authors. I wrote the main corpus of the paper and revised the submission based on the feedback from the reviewers, especially our shepherd Karen Renaud.

**Chapter 11** is based on Florian Mathis' bachelor thesis [118]. I developed the original idea, provided guidance through the project, and made the key decisions. Specific aspects were always jointly discussed and agreed upon. Florian implemented the prototype and I performed source code reviews. He also was the experimenter and I shadowed him in user sessions. Each of us independently coded qualitative aspects and created the codebook. I also performed further quantitative analyses with the dataset. The results have been published at the Australian Conference on Human-Computer Interaction (OzCHI '17) with Florian Mathis and Heinrich Hußmann as co-authors. I wrote the paper and revised it based on feedback from my co-authors before and from the reviewers after the submission. Moreover, I redacted the paper for the final submission.

**Chapter 12** includes a design exercise that was carried out together with Magdalena Siferlinger and Martin Prinz as part of their theses [256, 306]. I developed the idea for the exercise and defined the goals and scope of the project. I provided guidance and discussed all steps with them in weekly meetings. The students executed the interviews and created the prototypes, while I independently analyzed the data and drew inferences from it.



## Acknowledgments

Before I take you with me on a journey to the password universe, I want to say “thank you” to the people who have supported me throughout the years and who gave me the strength to finish this thesis. Words cannot describe how lucky I am to have (had) so many awesome colleagues, friends, and family members and I’m deeply humbled by your unconditional kindness. I would really love to give credit to each and everyone of you. But I need to embrace the essence of user experience and *keep it simple*, so I’ll focus on the people without whom I simply could not have written this thesis.

First and most of all, I want to thank my wife **Lena**, who couldn’t have known what she was getting into when we first met in Kindergarten. Thank you for constantly making me a better human and empowering me to pursue my dreams. I promise this was my first and last doctorate and I’ll have more time for you again.

Working at the LMU media informatics group was incredibly great, thanks to so many fantastic people who all create an amazing work atmosphere. I have learned so much from my advisor **Heinrich Hußmann**, who always has an open ear for all sorts of affairs – from bleeding edge web technology over research conundrums to personal matters. I wish I could deal with that amount of incoming requests as calmly as you. Thank you for enabling me to reach my goals! **Andreas Butz** and **Florian Alt** also have a great sense for hiring the right people. Thanks for your great input that sparked new ideas and corresponding publications.

For the longest time, I shared an office with **Axel Hösl** and will really miss our daily philosophical conversations about punk and hip-hop subcultures, the broken system, the universe, and all the crazy things the Internet has to offer. I’m so glad that you live on my street, so I can just drop by to continue the conversation. Before Axel and I formed “437 Forschungsbande”, we were part of the “A206-crew” together with **Emanuel von Zezschwitz**. Ema has always been a special role-model because he is a workhorse who never complains about workload, he’s my favorite DJ, leader into debauchery, and ever helpful USEC rockstar – I’m thankful for every bit of that. I deeply admire **Hanna Schneider** for her endless perseverance in so many situations. Thank you for opening my eyes for new lines of thoughts, your indulgence when I was being an idiot, and for empowering me to play Schafkopf. My work overlap with **Julie Wagner** lasted only a few months, but that was enough to become the best buddies. Thanks for sharing your passion for wild ideas, serendipity, and joint road-trips! **Sarah Tausch** is the most lovable person ever, a great neighbor who ensures our cats don’t starve, and a formidable jam partner. Thanks for your friendship and spending so much time with me. Thank you, **Henri Palleis** for sharing your dry humor and musical skills with me. Also, I hope one day you’ll let me buy one of your fantastic artworks. **Christian Mai** is one of a kind in terms of energy and a go-to guy to get a job done. Thanks for all the fun we had on our road trip through Colorado and for boosting my confidence all the time. **Daniel Buschek** sometimes intimidates me with his wisdom, but his matchless serenity make him an awesome person to hang out with in all circumstances. **Renate Häuslschmid** could snap my neck if she wanted to. Thank you for not doing that and being super nice and helpful instead. When **Malin Eiband** and I met for the first time, she thought I was an intern be-

---

cause I didn't have a proper desk at the window at the time. Thanks for all the laughs we had over coffee / tea breaks in the kitchen. Thank you, **Mohamed Khamis**, for sharing your funky ideas – and rooms at IDCs. When it was time to start writing the thesis, I was so glad to have **Nada Terzimehic** to take over the majority of my teaching duties. Without her, I'd probably still torture students with the peculiarities of JavaScript. Finally, two people act behind the scenes, but they really are the backbone of the media informatics group: **Franziska Schwamb** and **Rainer Fink**. Franziska and I have always been in tune in terms of community spirit and helping each other out. Her comments made me laugh so often and without Franziska's help I probably would have gone mad over expensing business travels. Thanks for everything, Franziska. I still don't know how Rainer manages to do so many things at once. Whatever I asked for, I never expected to receive it right away, but Rainer delivers on requests faster than you can say "Windows-account termination policy enforcement". All that without ever losing his unique sense of humor – thank you!

Some people were actively involved in finalizing the thesis: I was smart enough to ask a bunch of folks to give feedback on the dissertation draft. **Hanna Schneider, Judith Kratzer, Julie Wagner, Lena Seitz, Malin Eiband, Mohamed Khamis, Molly Throgmorton, Sarah Tausch, Sophie Himmelstoß** - you are my heroes! Thanks for taking the time to provide your thoughts on such a dry topic. I still hope I can return this huge favor somehow. Speaking of favors: Thank you, **Alexander De Luca**, for making good things happen for me, although I've never really understood why you do this. Alex also connected me to **Kosta Beznosov** who volunteered to read the thesis, write a review, add tons of helpful suggestions, and travel to Munich to my defense – thank you so much for the huge amount of time you sacrificed for me, although we had never met before. **Albrecht Schmidt** also jumped in as a committee member and helped close this chapter - thanks for your flexibility! I'll never forget the people with whom I started the PhD journey: **Marion Koelle** and **Patrick Lindemann** – thank you for your great company and long-lasting friendship.

Last, but not least, I want to express my gratitude to my family, but I prefer doing this in German. — Danke **Irmgard** und **Klaus Stockinger**, dass ihr mir den Weg geebnet habt, um diesen Meilenstein zu erreichen. Danke für's Zuhören wenn's mal wieder nicht so voran ging wie gewollt und die Finanzspritzen, um mich mit Urlaub oder Musikkram abzulenken. Danke, dass ihr meine ab und an einsetzende Klugscheißerei überhört. Danke für euer Verständnis, wenn ich mal länger nicht angerufen habe oder wenn ich der Frage "und wie geht's der Doktorarbeit?" ausgewichen bin. Danke, dass ihr mir immer zu verstehen gegeben habt, dass ihr glaubt ich schaffe das, und dass es auch okay wäre, wenn's nicht klappt. Ist zum Glück alles gut gegangen. Ich möchte mich auch bei meinen Schwiegereltern **Claudia** und **Werner Seitz** bedanken, dass sie mich seit so langer Zeit bei sich zu Hause aufnehmen und versorgen. Ohne die Zeit im "Schreibexil" bei euch hätte sich der Abschluss der Arbeit noch viel weiter verzögert. Danke, dass ihr immer für mich da seid! —

I wish I had space here to thank everybody who deserves to be mentioned. To deal with this limitation, I continue at [tobi-seitz.com/thanks](http://tobi-seitz.com/thanks). Since I dedicated some research to emojis, the webpage lists a few more names and emojis that I associate with them. Thanks for being there! Let's roll!

# TABLE OF CONTENTS

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	The State of the World . . . . .	1
1.2	Problem Statement and Research Objectives . . . . .	3
1.3	Main Contributions . . . . .	5
1.4	Thesis Structure . . . . .	6
1.5	Style Choices . . . . .	9
<b>I</b>	<b>FOUNDATIONS OF USABLE AUTHENTICATION</b>	<b>11</b>
<b>2</b>	<b>Foundations</b>	<b>13</b>
2.1	A Brief History of Passwords . . . . .	13
2.2	Attacks on Passwords . . . . .	15
2.3	What is Password Strength? Metrics and Statistics . . . . .	23
2.4	What is a “Bad” Password? . . . . .	29
2.5	Authentication beyond Passwords . . . . .	31
<b>3</b>	<b>Passwords – A User Perspective</b>	<b>45</b>
3.1	Methodology: Running Password Studies . . . . .	46
3.2	Password Coping Strategies and Risky Behaviors . . . . .	54
3.3	Guiding and Aiding Users . . . . .	62
3.4	Persuasive Interventions . . . . .	70
<b>4</b>	<b>Related Work Summary</b>	<b>83</b>

---

## **II EXPLORING THE CONTEXT FACTORS** **87**

### **5 Mental Models of Password Strength** **89**

5.1	Background and Context . . . . .	90
5.2	Approach: PASDJO - The Password Game . . . . .	91
5.3	Log Analysis . . . . .	98
5.4	Discussion . . . . .	105
5.5	Summary . . . . .	108

### **6 Password Policies and Reuse** **111**

6.1	Background and Context . . . . .	112
6.2	Method: Reverse-Engineering Password Policies . . . . .	113
6.3	Results . . . . .	114
6.4	Discussion and Implications . . . . .	116
6.5	Conclusion and Future Work . . . . .	120

### **7 Password Personality** **121**

7.1	Background and Related Work . . . . .	122
7.2	Study 1: Policies . . . . .	123
7.3	Study 2: Strength Perceptions . . . . .	132
7.4	Study 3: Password Selection and Coping . . . . .	143
7.5	Discussion and Implications . . . . .	148
7.6	Conclusion and Future Work . . . . .	154

### **8 Mental Models of Password Managers** **157**

8.1	Background and Context . . . . .	158
8.2	User Interviews . . . . .	159
8.3	Mental Model . . . . .	165
8.4	Opportunities and Challenges . . . . .	166
8.5	Conclusion . . . . .	168

<b>III</b>	<b>PERSUASIVE DESIGN STRATEGIES</b>	<b>171</b>
<b>9</b>	<b>Exploring Needs in Persuasive Feedback</b>	<b>173</b>
9.1	Background and Context . . . . .	174
9.2	Explicit and Observable Needs . . . . .	174
9.3	Tacit and Latent Needs . . . . .	179
9.4	Discussion . . . . .	184
9.5	Conclusion . . . . .	185
<b>10</b>	<b>Password Selection and the Decoy Effect</b>	<b>187</b>
10.1	Background and Context . . . . .	188
10.2	Designing Password Choice Architecture . . . . .	192
10.3	Quantitative Evaluation . . . . .	193
10.4	Discussion . . . . .	200
10.5	Conclusion . . . . .	202
<b>11</b>	<b>Extending the Password Space with Emojis</b>	<b>205</b>
11.1	Background and Context . . . . .	207
11.2	User Study . . . . .	208
11.3	Results . . . . .	213
11.4	Discussion . . . . .	220
11.5	Conclusion . . . . .	222
<b>IV</b>	<b>SYNTHESIS</b>	<b>225</b>
<b>12</b>	<b>Persuasive Design for Password Support (P4P)</b>	<b>227</b>
12.1	Research Lens . . . . .	229
12.2	Design Lens . . . . .	230
12.3	A Design Exercise with the P4P Framework . . . . .	231
12.4	Summary . . . . .	238

---

<b>13 Summary</b>	<b>241</b>
13.1 Central Contributions and Insights . . . . .	242
13.2 Limitations . . . . .	246
<b>14 The End - Ideas and Final Remarks</b>	<b>249</b>
14.1 Ideas for Future Work . . . . .	249
14.2 Final Thoughts . . . . .	253
<b>Bibliography</b>	<b>255</b>
<b>Appendix</b>	<b>295</b>
<b>Glossary</b>	<b>301</b>

# 1

## Introduction

We have become accustomed to using multiple passwords every day: We enter a four-digit PIN to unlock our phone to check incoming messages even before breakfast. Paywalls shield the news of the day, so they force us to log into our favorite news website before we can read them during our commute. Once arrived at work, our fingers automatically type the password to unlock the computer. A colleague requests access to a client's platform, so we write down the credentials or share them via a password manager that itself is password protected. When we come home, we find the kids have used the family's tablet to log into their social media accounts, so we need to re-authenticate if we want to check our own. Interacting with systems relying on password authentication has become ubiquitous, and we merely carry out the task. Passwords are the de facto standard when it comes to controlling access to resources on the Internet. While it may seem straightforward to deal with passwords because we are so acquainted with them, there is a wide range of problems they entail: The phone requests the PIN whenever we take it out of our pocket, so we spend considerable time per day authenticating [155]. The news website informs us that our password has expired, so we need to pick a new one that is not part of the ones we had used before. The work PC requires not only our credentials but also a one-time password that our phone generates for us every sixty seconds. At one point, the colleague leaves the company, so we have to request a new set of credentials from the client to maintain confidentiality. And the kids used the tablet at home to shop online because we had stored the password in the browser for convenience. So, we delete the stored password but fail to recall it a week later, because we had not typed it for a very long time. These and other situations lead to users feeling a considerable burden generated by password authentication.

### 1.1 The State of the World

In a wide sense, password authentication encompasses digit-only personal identification numbers (PINs), graphical schemes like the Android screen lock pattern (see Section 2.5.1),

---

and alphanumeric passwords consisting of letters and digits. Throughout this thesis, however, we focus on the latter because they still are the go-to method on the Internet: In 2007, users had around 25 accounts and roughly six distinct passwords on average [111], while more recent numbers suggest that users keep twelve unique passwords for different purposes [363]. With the growing number of accounts the issues around passwords amplify: users pick weak passwords, write them down in unprotected locations, reuse many of them, forget the exotic ones, and share credentials with other people [316]. These behaviors are referred to as *password coping strategies*.

Combating the risks entailed by these issues, researchers and practitioners have developed numerous approaches to support users in password authentication. Four central themes have emerged in their efforts: education, enforcement, assistance, and persuasion.

## **Education**

Weak password practices were attributed to a lack of understanding of the consequences for the longest time [276]. Thus, first approaches to mitigate the situation leaned on educating users through trainings, and textual instructions [162]. Explaining all risks and defense strategies to users in lengthy prose is doomed to fail, because password security is a secondary goal that is dominated by a primary task, e.g., reading emails [371]. Nevertheless, carefully crafted advice and explanations may be a viable strategy for those who actively seek information [277, 341].

## **Enforcement**

Since the inception of passwords in the 1960s, users have tried to create memorable passwords that are often based on simple dictionary words. To combat low complexity, enforcing password rules through policies is commonplace. Passwords must then meet length and complexity requirements, e.g., a certain number of uppercase letters, digits, or symbols. The choice of a specific policy is not trivial for service providers, and many organizations make poor tradeoffs [112, 295]. The plethora of policies even fosters insecure password practices, because users try to find easy ways to comply with the rules, which they do in very predictable ways [172, 200, 345].

## **Assistance**

Users have to deal with a high number of passwords that also interfere with each other, so people reuse passwords and write them down on paper or digital files. To lower the risks entailed by these coping strategies, assistive tools like password managers (PWMs) and password generators automate certain interactions to boost both security and usability. However, they come at a price, e.g., lock-in effects to a particular vendor that make it difficult to move to a different tool in the future. Thus, many users rationally refrain from adopting them [64].



## Persuasion

Finally, the youngest strategy to support users in password authentication is based on principles of *persuasive technology*. Fogg defined this paradigm as “any interactive computing system designed to change people’s attitudes and behaviors” [120, p. 1]. While assistive systems also often meet that particular goal, persuasive technology tries to create sustainable impact even when the assistive trigger is absent. Therefore, such interventions often come as *behavior-change support systems* [246]. In the realm of authentication, password meters are the most representative form of persuasive interventions. Those user interface (UI) elements often appear on registration pages of web services, and give feedback on the strength of a user-selected password. They often implement various *nudges*, i.e. small transparent attempts to change behaviors [332, p. 4], to convince the user to pick a more suitable password. The user stays in control and is free to move on without following the advice. Overcoming inertia is perhaps the biggest challenge in the design of persuasive password support. So far, the use of nudges has shown mixed results in empirical research [99, 267, 341], and the spectrum of persuasive interventions in the wild is fairly narrow.

## 1.2 Problem Statement and Research Objectives

Password-related challenges and risks for the users are at the heart of this dissertation. The balance between usability and security, especially for passwords, has been under investigation for several decades. This allows us to observe tectonic shifts in the hassles that users have to bear. The notion of the inconsiderate user, who is the “weakest link” in the authentication chain and notoriously refuses security measures, has started to crumble: There is considerable evidence that many users want more control over their security and that they are willing to give up usability for it [192]. This sacrifice might not even be necessary, nor lie in their best interest, which I illustrate below.

### 1.2.1 Balancing the Costs

Changing the status quo of the authentication world can be seen as a game-theoretic problem [35]. It involves risks and opportunities that need to be balanced in terms of their costs for different “players”.

#### Maintaining Usable Password Practices

Fortunately, the costs of being attacked remain hypothetical for many users [164]. However, there is a growing number of people who have experienced an attack with all its consequences to resolve the damage and recover from it [45]. First, usable but risky password practices, like excessive reuse and obvious password choices, can cause financial losses for end-users. For instance, an attacker who manages to impersonate a user might be able to

---

withdraw money from bank accounts. At the moment, attacks on digital wallets containing cryptocurrency are highly lucrative, so protecting these assets with strong passwords is vital<sup>1</sup>. Herley et al. noted that “*money is the most obvious loss, but time, frustration and reputation are also at stake*” [165], let alone the emotional distress [296]. Accounts that have a weak password are more likely to be hijacked [358], and it often takes victims painful effort to recover from identity theft<sup>2</sup>. Although social engineering, where an attacker lures people into forfeiting their credentials, is a central threat for companies, the employees’ overall password practices still generate considerable financial losses<sup>3</sup>.

## Striving for Stronger Password Practices

Solving the problems raised by passwords, we also need to consider the other end of the security-usability spectrum. Moving to strong password practices often inflates usability challenges for users that are largely neglected by security experts. For instance, typing an overly complex password takes long and is error prone [299]. Such passwords are especially tedious to enter on mobile phones or devices that were not originally designed for text input, e.g., smart TV sets [232]. Moreover, most people are incapable of creating and memorizing a strong, unique password for every single account on their own. So, it is unrealistic to expect that they will do so without the use of external aids like handwritten notes or password managers. The cost of using such methods is a dependency on the tools that users did not ask for in the first place. In any case, strong passwords are no panacea in boosting online security. They do not stand a chance in fending off social engineering attacks where the victim unwillingly surrenders the password in plain text. In an effort to alleviate the risks of unknowingly forfeited credentials, service providers often require users to reset passwords after a given period. Such expiration policies are rather ineffective [53] and responsible for many support desk calls whose resolutions are costly [3, 277].

### 1.2.2 The Challenge: Improving an Innately Annoying Interaction

Passwords are annoying for users, and as we can see above, there is no easy solution to alleviate this situation. It is impossible to remove all usability pain points [34]. Yet, no alternative can fully replace passwords, either (see Section 2.5). Thus, Herley and Van Oorschot point out that “*supporting passwords better is a vast opportunity for improvement*” [165], because making even a small change can have an impact on so many users.

---

<sup>1</sup> <https://blog.dashlane.com/cryptocurrency-exchange-password-power-rankings-2018/> (last accessed 24.03.2018)

<sup>2</sup> <https://www.businesswire.com/news/home/20151006006149/en/Latest-Data-Breach-Spotlights-Identity-Restoration> (last accessed 11.01.2018)

<sup>3</sup> <https://www.helpnetsecurity.com/2017/09/19/infosec-weakest-links/> (last accessed 09.04.2018)

Up until now, persuasive strategies have produced mixed results regarding their efficacy in supporting users with passwords. This could be due to several issues: Mental models and coping strategies evolve over time [316, 352], which has seen little attention in the design of persuasive interventions. Much of the past research dealt with one-shot triggers in isolation, but many context factors and costs for different stakeholders were left out. In an analogy to the design principle “form follows function”, a better understanding of the functions of user support can help design assistive and persuasive solutions (the form). Moreover, many highly effective nudges from other domains have not been adapted for password support, so we simply may not have discovered the best intervention, yet. However, since nudging strategies are nuanced, we need a structured exploration of how we can bring them to password authentication to remove its most important pain points.

### 1.2.3 Research Questions

In this thesis, I take a holistic approach to address the problem of providing users with the right support in their password practices. To accomplish this, the research presented here tries to answer the following questions:

- RQ1** What is the role of psychological factors and mental models for password selection and coping strategies?
- RQ2** How can password authentication be simplified for users?
- RQ3** How can we design persuasive strategies to support users in any password-related tasks?

## 1.3 Main Contributions

As highlighted above, several aspects of password support have been left out in the literature, although there are many reasons to consider their importance. First and foremost, a broader understanding of contextual factors that contribute to the formation of specific coping strategies is necessary to improve password support. The primary goal of this thesis is to provide this understanding on a fundamental level by exploring existing factors and addressing them with persuasive designs. The solutions include new paradigms for the design of persuasive interventions.

### Insights into Context Factors of Password Practices

Researchers seem to have reached consensus that the context in which a password goes through its life-cycle [316] is an explanatory variable for users’ practices. However, contextual factors have merely been addressed in the discussions of empirical findings. Only a few studies specifically correlated users’ backgrounds to their password practices (e.g.

---

[192, 226]) and they mostly focused on demographic factors. We contribute several insights that enrich the understanding of a wide range of context factors. Specifically, we address how users' mental models are associated with their password practices. We do this through a novel method to study mental models in-the wild with the aid of a game. Moreover, we are the first to thoroughly investigate the interconnection between personality traits and password authentication. The insights gathered in three online studies revealed interesting associations between personality, attitudes, and behaviors regarding passwords. Lastly, we performed an extensive audit of the real-world constraints that form the context for password reuse. All these insights shape our understanding of the problem space as the foundation for persuasive interventions.

### **Investigation of Persuasive Strategies**

With the context factors in mind, we extended the range of persuasive design strategies. As a starting point, we investigated users' explicit and implicit needs in password feedback. From this exploration, we contribute the "*show-explain-help-empower*" paradigm that serves as a heuristic for persuasive password assistance. We followed this up with two studies that were carried out both in the lab and in the field: The first study was the first of its kind to evaluate the Decoy effect for choice architectures in password authentication. Here we learned important lessons about the interplay between feedback and feedforward, and about the role of simplification in persuasion. The second study was focused on empowerment. We evaluated different dimensions of usability of emojis inside text-based passwords. The study delivers timely insights, because an increasing number of web-services enable users to pick such emoji-passwords and there are some issues that need attention from the very start.

### **A Structured Process for the Persuasive Design of Password Support**

Finally, the exploration of the context factors, and the design studies on persuasive assistance in password authentication are synthesized into a framework for structuring future design processes in this domain. I contribute the Persuasive Design for Password Support (P4P) framework. It respects the dynamism of the status quo and aids in both finding the right interventions and implementing them successfully. To that end, we go through a design exercise that demonstrates how the P4P framework can be used in practice.

## **1.4 Thesis Structure**

This dissertation encompasses four major parts that unravel the different aspects of persuasive password support. I chose to structure the content with fourteen self-contained chapters. Although they do follow a narrative, it is possible to read them in any order by following the provided cross-references for the necessary background information. Part I is an exhaustive overview over the related work that serves as the basis for all the discussions in later parts. In Part II, I report on empirical research that explores the various contextual factors

of password selection and coping strategies. Part III then shows how these factors helped to craft novel persuasive design strategies. Lastly, Part IV establishes a research and design framework, and concludes with a reflection on the gained insights and future work. In the following, I highlight the contents of the individual chapters with the questions they try to answer.

## **Part I: Foundations of Usable Authentication**

**Chapter 2: Foundations** This chapter provides an overview of password-based authentication from a system-perspective. Questions answered:

- How has password authentication evolved over time?
- What benefits, drawbacks, and threats do passwords entail?
- What is a strong, what is a weak password?
- Why do we still need passwords when there are more advanced schemes?

**Chapter 3: Human Factors** I describe the method space to study passwords, before discussing findings about users' password practices. The chapter also highlights the central approaches that have been implemented to mitigate security risks on the user side. Questions answered:

- How do we conduct valid research on passwords with humans and ethics in mind?
- How do users cope with passwords? What makes their practices particularly risky?
- What can we do to steer people away from risky behavior?

**Chapter 4: Related Work Summary** This chapter describes the status quo of password authentication and highlights ill-defined aspects that warrant further research.

## **Part II: Exploring the Context Factors**

**Chapter 5: Mental Models of Password Strength** We present a novel approach to study the perception of password strength: PASDJO, the password game. A longitudinal field study aimed to quantify common misconceptions about the benefits of password complexity, which are an underlying context factor for password practices. Questions answered:

- How well can users gauge password strength?
- Do we have to update our views on users' capabilities?
- Is a game suitable to collect the necessary data?
- How effective is the game to educate users?

**Chapter 6: Policies and Reuse** This chapter reports on a thorough audit of the password policies of the most-visited websites in Germany. It explains external context factors that shape password reuse in the real world. Questions answered:

- How consistent are password policies in the wild?
- Is it possible to find a password that meets all requirements at once?

---

**Chapter 7: Personality in Password Practices** This chapter presents three empirical studies about the role of personality traits in password practices. In particular, we shed light on the psychometric context factors for the usability of policies, mental models of password strength, and password selection behavior. Questions answered:

- Is personality associated with password practices, attitudes, and behaviors?
- How well can we model such associations?
- What are the specific implications on the design of personalized password support?

**Chapter 8: Mental Models of Password Managers** We present a qualitative user study eliciting the users' motivations to either adopt or dismiss password managers. A fine-grained mental model is established to depict biases as context factors. Questions answered:

- Why are people (not) using password managers?
- How do they make sense of their functionality?

### **Part III: Persuasive Design Strategies**

**Chapter 9: Feedback Requirements** This chapter presents two studies on users' explicit and implicit expectations around password feedback. We derive a paradigm for persuasive password support. Questions answered:

- What are users' needs in persuasive feedback?
- How would they design a feedback system?

**Chapter 10: The Decoy Effect** We carefully craft a choice-architecture for password support and explore a marketing phenomenon as nudging strategy. The chapter reports on an online study and highlights the interconnection between feedback and feedforward. Questions answered:

- Does the decoy effect work to make stronger passwords more attractive?
- How effective is feed-forward in combination with feedback?

**Chapter 11: Emoji-passwords** This chapter presents emoji-passwords as an approach to simplify memorization in persuasive ways. We investigate different facets of usability and report on a mixed-methods study. Questions answered:

- How usable are emoji-passwords?
- What are the risks and potentials of emoji-passwords?
- How do platform-dependent differences affect memorability?

## Part IV: Synthesis

**Chapter 12: P4P Framework** This chapter synthesizes the insights from the first three parts to establish a new framework for the design of persuasive password support. Through a design exercise, I show how it can be applied to develop a novel password manager. Questions answered:

- How can we design persuasive password support in a structured way?
- How do we practically apply the framework?

**Chapter 13: Summary** In this chapter, I reflect on the presented research and draw conclusions. The contributions are summarized, and contrasted by the limitations of the methodology. I provide eight meta recommendations for future design work. Questions answered:

- What have we learned?
- What are the implications and limitations?
- What do we need to consider in the future?

**Chapter 14: The End** In the final chapter of this thesis I present open research topics and show their potentials. The dissertation concludes with a reflection on the role of password-authentication in the present and the future. Questions answered:

- What research topics have been opened up by this thesis?
- What still needs to change to make users' authentication practices easier?

## 1.5 Style Choices

**Singular They:** Throughout this dissertation pronouns are used in the plural although speaking about an individual, e.g. “the user” is mostly referred to as “they” instead of “he” or “she”, to avoid discrimination of certain demographic groups.

**Plurals:** As is common in HCI literature, the author utilizes “We” instead of “I” to acknowledge the work of collaborators. In later more opinionated parts, the explicit usage of “I” intends to communicate the subjective nature of thoughts and interpretations.

**Footnotes:** Throughout the thesis, footnotes are excessively used to link to web content. Publications in scientific archives appear in the list of references at the end of the thesis.

---



# I

## FOUNDATIONS OF USABLE AUTHENTICATION



# 2

## Foundations

Passwords are but one puzzle piece in the realm of cyber security. They are part of the access control paradigm, which is commonly divided into three steps: identification, authentication, and authorization [268]. Identifying a user is usually done with a prompt for a user name, so the system first tries to answer the question “who are you?”. Authentication is about proving that a user – or more broadly an entity – is who they claim to be, i.e. authentication is about *verification of an entity*. In other words, the system asks “How can you prove that you are who you claim to be?”. This verification process can be based on three central elements, namely something that you know (any kind of secret), something that you are (any kind of property), or something that you have (any kind of secret token). One could argue that the second category could be extended by “something that you do”, e.g. provide your individual signature. However, authentication does not necessarily *require* identification as a first step. It is well possible to authenticate entities even if they are anonymous. One everyday example are passwords for WiFi networks: Most of the time, the access point does not require a user name; the password, or pre-shared key, in the WPA protocol is enough to join the network. Last, authorization is the decision over which resources an authenticated entity is allowed to use, or an answer to “can the user access this?”.

For this thesis, **authentication** remains the center of attention. In this chapter, we take a look at various forms of authentication and establish an argument as to why “something that you know” is still the most prevalent and significant authentication paradigm to date.

### 2.1 A Brief History of Passwords

The idea of protecting resources with “something that you know” is hundreds of years old. Think about the magic words “open, Sesame!” that Ali Baba spoke to enter a den used as treasury by forty thieves<sup>1</sup>. The story already illustrates password misuse, because Ali Baba

---

<sup>1</sup> Ali Baba is a fictional character in a story from “Thousand and One Nights” as recorded by Antoine Galland in the early 18th century. <http://www.pitt.edu/~dash/alibaba.html> (last accessed 16.12.2017)

---

was not the rightful owner of the den and he impersonated the thieves. Nonetheless, this authentication paradigm was brought to the digital world in the early 1960s by the Massachusetts Institute of Technology (MIT)<sup>2</sup>. Researchers had built a mainframe computer that was programmable by multiple users. At the time, one of the most valuable resources beside the physical device was the *time* granted to use the machine. Thus, the Compatible Time Sharing System (CTSS) was conceived to give every user a certain quota of hours to operate the computer. The quota was enforced with the aid of dedicated user accounts that were password protected. The interaction very much resembled what we still use today: after typing the user name, the password is requested and hidden from the screen during entry to prevent others from observing the credentials.

Shortly afterwards, the flaws of the system started to become evident when Alan Scherr became the first “hacker”<sup>3</sup>. He desired more usage time, so he needed to impersonate other users of the computer and use their quota. The list of passwords on the system was not well protected, which allowed him to access the credentials and carry out what was probably the first password exploit in computer history. Scherr benefited from the fact that passwords were kept in plain text and could be accessed with a special punch card. Interestingly the attack was not detected immediately. Morris and Thompson mention strange behavior in their 1979 paper and blame the issue on a “software design error” [237], while in fact it was Scherr who was responsible for it<sup>4</sup>. With the rise of the UNIX operating system in the 1970s, encrypted passwords became standard. One of the cornerstones was the Data Encryption Standard (DES) which was developed by IBM and propagated by the US National Bureau of Standards (today called the National Institute of Standards and Technology, NIST) [25]. This algorithm was widely used until the late 1990s when computing power was sufficient to efficiently carry out attacks, which rendered DES infeasible. At the time, the Advanced Encryption Standard had already been proposed and was able to replace DES in a straightforward manner, so the issue was resolved quickly.

---

<sup>2</sup> <https://www.wired.com/2012/01/computer-password/>, (last accessed 16.12.2017)

<sup>3</sup> [https://www.slideshare.net/CAinc/history-of-the-password/7-In\\_1962\\_a\\_software\\_bug](https://www.slideshare.net/CAinc/history-of-the-password/7-In_1962_a_software_bug) (last accessed 16.12.2017)

<sup>4</sup> <https://www.wired.com/2012/01/computer-password/> (last accessed 05.03.2018)

**Table 2.1:** Benefits and Drawbacks for different stakeholders in a password-based authentication. SPs = service providers.

Stakes	Benefits	Drawbacks
<b>SPs</b>	Low costs Easy to implement Replaceable when compromised Revocable by administrator Enforceable policies	Large number of attack vectors Anomaly detection costly Attacks are simple to carry out Attack automation simple Attacks can have severe consequences
<b>Users</b>	Fast entry on desktops Most users already familiarized Easy to learn Sharable with others High degree of control / freedom	Memory overload from too many passwords Suboptimal coping strategies Stronger passwords difficult to memorize Entry on mobile devices difficult Mastery difficult Disliked by many users / perceived as burden
<b>Misc</b>	Independent of identification Adjustable security level	Weak passwords are a risk for users and SPs

Around 50 years after CTSS, one of its creators, Fernando Corbató, said in an interview with the Wall Street Journal that password-based authentication has “become kind of a nightmare with the World Wide Web”<sup>5</sup>. The surge of the Web has led to many services requiring authentication. Alphanumeric passwords were the go-to solution because the system is easy to implement and has almost no set-up costs other than a database. This has led to passwords becoming the de-facto standard for authenticating users on the Web.

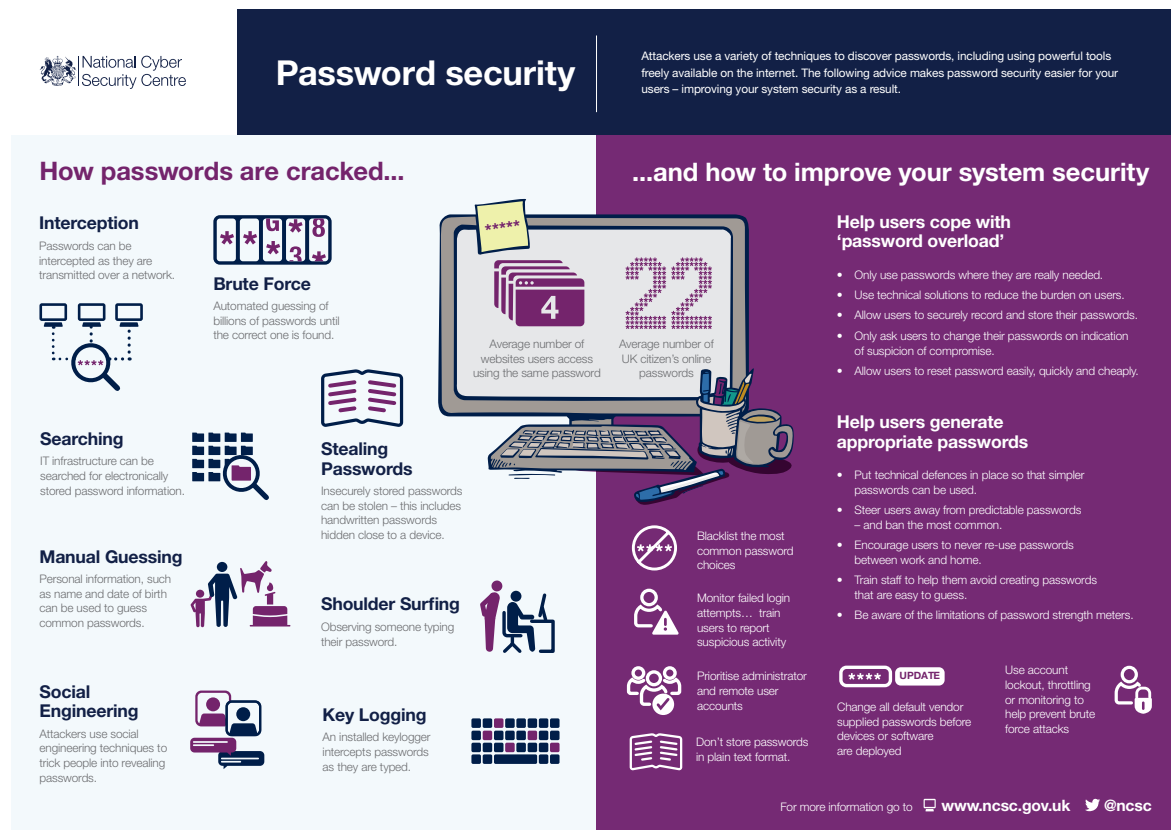
However, passwords do have shortcomings for all the stakeholders involved in the authentication process, e.g. users who struggle with remembering a multitude of passwords, or service providers who need to deal with leaked passwords (see Table 2.1). Consequently, there have been many attempts to replace passwords as a whole to either minimize security risks, make things easier for the users, or – ideally – both at the same time. Until now, though, no alternative authentication mechanism has been able to fully replace alphanumeric passwords on the web, which we investigate in detail in Section 2.5. Put short, the benefits provided by passwords outweigh the drawbacks most of the time.

## 2.2 Attacks on Passwords

As mentioned above, computer passwords were attacked shortly after their inception. Attacks have since become more sophisticated and manifold (see Figure 2.1). In the first attack on passwords, Scherr benefited from very weak protection and was able to simply print the

<sup>5</sup> <http://on.wsj.com/1sVQ0Iv>, (last accessed 18.12.2017)

passwords and hack into the system. Nowadays, an attacker, or “the bad guy” as Morris and Thompson used to call them [237], has a number of ways to obtain a user’s password(s). The following sections depict how the attacks work, what the countermeasures look like, and if changing one’s password behavior serves as effective countermeasure.



**Figure 2.1:** Overview of attacks and high-level mitigations. Image courtesy of the National Cyber Security Centre, UK <https://www.ncsc.gov.uk/guidance/password-collection> (last accessed 21.12.2017)

**Online Guessing** An adversary tries to impersonate the user by trying different combinations of user names and passwords, which are sent to the authenticating service directly. If successful, the attacker can log in without the user noticing and steal personal information, act on their behalf, and try to use the credentials on other services as well. This attack, which is commonly known as an “online attack”, is in many cases thwarted by throttling the number of unsuccessful login attempts per account. The service provider can lock down a user account entirely after a given number of failed login attempts. Afterwards, the user either has to manually reset their password, or they need to wait a certain time until the account is unlocked and new login attempts can be made. In the latter scenario, a strategy to hamper attacks is to implement a *backoff* algorithm<sup>6</sup> like, e.g., in the Ethernet protocol [328, p. 285]. The idea behind this scheme is to (exponentially) increase the time a user (or attacker) has

to wait until they can log in again after the account was locked down. Brostoff and Sasse suggest allowing ten attempts until the account is locked [40]. This should give users enough trials to go through their list of passwords which is usually shorter than ten [111]. Though this kind of countermeasure may seem like the best option, for system providers it comes at a higher price than for users: A malicious party could easily lock out a large number of users with a denial-of-service attack. For example, if a lockout policy is in place that invalidates passwords after ten failed login attempts, an attacker would only need to take a list of email addresses (readily available on the Internet) and run ten or more guesses per user. This would lock all of them out at once and the financial damage for the service provider to respond to user requests and/or reactivate accounts manually is probably large. There are defense strategies for this kind of threat, too, but their success cannot be guaranteed [112].

Perhaps, online guessing to access user accounts might only be feasible for determined attackers who target specific victims, but this type of attack is not entirely uncommon [113, 116, 164, 358]. It is sometimes argued that such an attacker might automate up to one million guesses until the attack becomes profitless because it would simply take too long [35, 115]. However, if login attempts are not throttled whatsoever, attackers have reasonable chances for payoff. This situation can lead to massive attacks, like the largest attack on WordPress to date in December 2017<sup>7</sup>. Florêncio et al. argue that users are well advised to pick passwords that can at least withstand this type of attack, because they stand a reasonable chance to fight them off. On the other hand, it becomes too difficult to protect themselves against offline guessing [115, 116, 117].

**Stealing / Offline Guessing** Since online attacks are often impractical due to time consumption, offline attacks have proliferated in recent years<sup>8</sup>. In this scenario an attacker breaks into the server of a service provider, usually by exploiting security holes. If this goes unnoticed, the intruder can often access the entire database containing the user account data. He or she downloads the data to their own machine, which allows them to use cracking tools like John the Ripper<sup>9</sup>, hashcat<sup>10</sup>, or PassFault [250]. These sophisticated tools use dictionaries, mangling rules and brute force to calculate password hashes which are then compared to the entry in the database. If the hashes match, the password was cracked and its plain text version is written to a file. Ideally, the passwords in the database are cryptographically hardened through salting and a slow hash function like bcrypt [259], which drastically reduces an attacker's chances to crack the password. At the other side of the attack surface, the passwords could be stored in plain text, which would not require any cracking automation at

<sup>6</sup> <https://devcentral.f5.com/articles/implementing-the-exponential-backoff-algorithm-to-thwart-dictionary-attacks> (last accessed 20.12.2017)

<sup>7</sup> <https://www.wordfence.com/blog/2017/12/aggressive-brute-force-wordpress-attack/> (last accessed 21.12.2017)

<sup>8</sup> <http://breachlevelindex.com/> (last accessed 20.12.2017)

<sup>9</sup> <http://www.openwall.com/john/> (last accessed 20.12.2017)

<sup>10</sup> <https://hashcat.net/hashcat/> (last accessed 20.12.2017)

---

all. Unfortunately, some of the most famous data leaks revealed that data was stored in plain text. The RockYou breach in 2009 contained 32 Million user accounts for its gaming website with plain-text passwords [32, 365]. At the time, RockYou developed games for MySpace and Facebook and the database also contained credentials for these sites<sup>11</sup>, which made the leak even more severe. Strong passwords would not have helped at all to avoid losing personal data. Perhaps RockYou’s loose policy (five characters or more, no further restrictions) helped in safeguarding other accounts where more complex policies were in place, because users were not able to reuse their RockYou password there. In other instances of stolen password databases, the passwords were indeed hashed, e.g. the infamous LinkedIn breaches<sup>12</sup> – again with millions of rows of user data [169]. Users are challenged to find a password that withstands this kind of attack. The large issue is that attackers are basically only limited by the time and money they want to spend on calculating password hashes [27]. On modern machines with a single GPU, thousands of hashes can be calculated per second even for slow algorithms<sup>13</sup>. Using a cloud instance with multiple CPUs can speed up this process even further<sup>14</sup>. Perhaps, this is why Florêncio et al. argue that it is futile to encourage users to pick a password that would withstand such an attack [115, 117].

**Phishing / Social Engineering** Social engineering has become one of the biggest threats for a user’s passwords with a growing number of incidents and fierce financial damage [45]. Former criminal hacker Kevin D. Mitnick, who calls himself a social engineer, defines the term like this:

“Social Engineering uses influence and persuasion to deceive people by convincing them that the social engineer is someone he is not, or by manipulation. As a result, the social engineer is able to take advantage of people to obtain information with or without the use of technology.” [236, Frontmatter]

Put simply, an attacker fools a victim into revealing certain kinds of information, including passwords. The most common social engineering attack on passwords is phishing, which typically involves two components: a fraudulent website that mimics another service and an email that lures the user onto this website [85, 301]. The email usually utilizes persuasive techniques like scaring (“we noticed someone logged into your bank account and you need to reset your password”) or time pressure (“you need to act *now* to avoid further damage”). If the website looks just like the original (like the web pages in Figure 2.2), users might

---

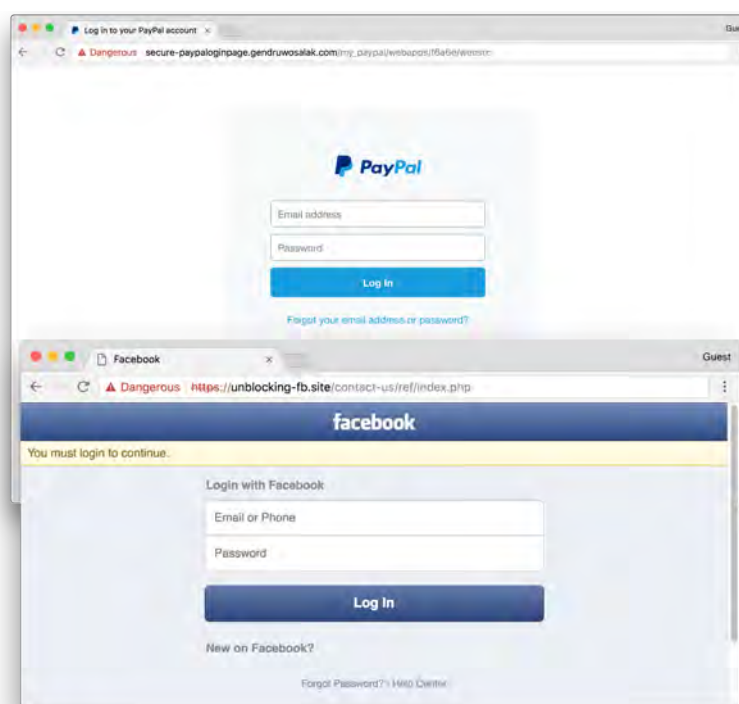
<sup>11</sup> <https://techcrunch.com/2009/12/14/rockyou-hack-security-myspace-facebook-passwords/> (last accessed 20.12.2017)

<sup>12</sup> <http://fortune.com/2016/05/18/linkedin-data-breach-email-password/> (last accessed 20.12.2017)

<sup>13</sup> <https://gist.github.com/epixoip/9d9b943fd580ff6bfa80e48a0e77520d> (last accessed 20.12.2017)

<sup>14</sup> <https://linuxundich.de/gnu-linux/erfolgreicher-brute-force-angriff-auf-pwdhash/> (last accessed 07.01.2018)

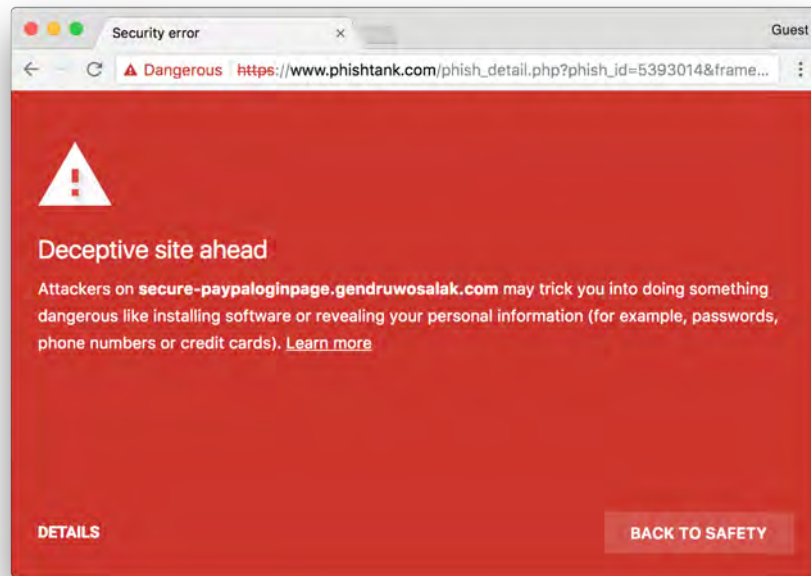




**Figure 2.2:** Actual phishing websites targeting Facebook and Paypal (online and accessible on 21.12.2017). The URLs contain “fb” (short for Facebook) and keywords like “secure” “paypal” which contribute to falsely trusting the authenticity of the sites.

fall for it and enter their password to log in. In that case, it does not matter whether it is a strong, complex password or simply 12345 – the attacker knows the username/password combination from that point on [329]. If this tuple is used on other services, the attacker immediately gains access to those as well.

It is very challenging for users to validate the authenticity of a given webpage [85, 121]. Usually, the URL is the best indicator. Dhamjia et al. among others argue that it is unrealistic to keep an eye on the URL at all times. Since the URL and padlock-icons are often ineffective, much research has been dedicated to help users in this validation and prevent phishing attacks. For example, Lin et al. found that domain highlighting in the URL bar only has a small effect on the effectiveness of phishing attacks, even after their participants were explicitly instructed to take note of the URL [212]. Wu et al. showed that browser toolbars do not really help users, either [378]. Dhamjia et al. proposed a *trusted path* between the user and the legitimate service [84]. In this system, users are supposed to verify the authenticity of a given website by comparing visual patterns in a trusted window and on the website. Together with Max-Emanuel Maurer and Alexander De Luca, I created a browser extension to visualize the usage of different types of SSL certificates [225]. The extension makes the SSL state more obtrusive by coloring the entire frame of the browser window, e.g., in green if all communication is securely encrypted. The user then does not



**Figure 2.3:** Chrome v63’s warning on a phishing website. The user is urged to leave the site but still has the chance to visit it by clicking on “Details”.

have to look for a small padlock icon or the “https://” URL prefix. We deployed it publicly and launched a feedback survey, which indicated that changing the browser skins is obtrusive enough to raise awareness and makes users more confident while browsing the Web. Until the recent change of platform APIs<sup>15</sup> and resulting incompatibility problems, the extension named “SSLPersonas” had seen 47000 downloads, which is an indicator of both the necessity and the success of our solution.

However, ideally the browser would detect phishing websites and users from visiting them in the first place. Current versions of the major browsers try to do this and urge the user to leave the site, as is shown in Figure 2.3. This gives attackers only a short time-frame until the webpage has been classified as phishing, which takes 15 hours on average according to a recent Webroot report [134]. Older sources report a bit longer lifespans between 20 hours<sup>16</sup> and 54 hours<sup>17</sup>.

<sup>15</sup> <https://blog.mozilla.org/addons/2017/11/20/extensions-in-firefox-58/> (last accessed 21.12.2017)

<sup>16</sup> <https://www.lightbluetouchpaper.org/2007/05/16/how-quickly-are-phishing-websites-taken-down/> (last accessed 21.12.2017)

<sup>17</sup> [https://news.netcraft.com/archives/2004/08/14/life\\_span\\_of\\_a\\_phishing\\_site\\_averages\\_54\\_hours.html](https://news.netcraft.com/archives/2004/08/14/life_span_of_a_phishing_site_averages_54_hours.html) (last accessed 21.12.2017)

**Malware and Eavesdropping** Secretly stealing plain-text passwords is also possible by infiltrating the user's system with *malware*, which is a term for *malicious software* [19]. One of most common attacks is to install a keylogger that sends all keyboard input to the attacker. For the most part, this either happens as a “drive-by-download” when the user visits an infected website or by opening a malicious email attachment [45]. In the former scenario, the sole line of defense lies with the service provider who needs to make sure their website is not infected. Wash identified that users are generally aware of this kind of threat, but their mental model of malware is sub-par [362]. Hence, the countermeasures taken by the users are often insufficient. As with the phishing scenario, a strong password is incapable of preventing a malware attack. Ideally, users use an anti-virus solution, keep their software updated at all times and refrain from opening suspicious email attachments.

Moreover, user credentials can be intercepted in transit, e.g. after the user submits them through a web form. These so called man-in-the-middle attacks are tough to carry out, but strong passwords do not prevent them either. One of the typical solutions to minimize the risk is encrypting the traffic with a secure protocol like SSL/TLS [328, p. 853ff.], which is based on a public/private key infrastructure. This makes it harder for an adversary to act as the man in the middle, because they would require the private key of either party. So, attackers must tamper with the certificates which usually causes browsers to show a warning [225]. Users, however, do not necessarily understand such warnings, because their understanding of the technical details is low [162, 371]. Consequently, the HCI community has invested much effort to convey the essential messages in a clear and actionable way [105, 107, 225, 309, 324].

**Observation** To obtain the password of a specific user, one can also simply watch them enter it and figure out what the password was. This attack is typically referred to as “shoulder surfing” because the person who is watching figuratively “surfs” on the user’s “shoulder” to look at their screen [329]. Shoulder surfing is relatively easy and does not require technical sophistication. But, although friends and family can carry out such an attack, it is safe to say that it is more problematic in public spaces where unknown bystanders are present. Typical interactions that take place in such environments can be found with PIN entry, e.g. on an ATM, or entering any kind of password on a mobile device, e.g. on a bus in close vicinity to other people. De Luca et al. dedicated some research to both scenarios. For instance, they investigated contextual factors during ATM usage and the design space for alternative ATM authentication mechanisms [78]. They developed the ColorPIN scheme, where numerical input needs to match a previously enrolled color sequence [77]. Entering the ColorPIN is done indirectly. The user needs to first identify the correct digit (e.g. “1”), and then type a corresponding letter in the right color (e.g. a black “Q”). A shoulder-surfer is hereby challenged by overwhelming their short-term memory [88].

Moreover, long alphanumeric passwords are a seldom-researched topic with regard to shoulder surfing. Shaub et al. looked into the effect of different designs of virtual keyboards on shoulder surfing susceptibility [279]. Somewhat unsurprisingly, they found that keyboards with more cumbersome access to special characters are less prone to shoulder surfing be-

---

cause it is harder for an attacker to keep track of the keyboard switches. Moreover, we can see in the ColorPIN example that PINs and graphical passwords are more likely to be under attack. Users usually enter graphical passwords more slowly [329, 373, 265], which on the one hand gives an attacker more time to observe, but on the other hand burdens the short-term memory a bit more. Also, some visual authentication mechanisms require more screen real estate than a username/password form and entry is often not masked [23]. Still, there is not a lot of evidence that observation is a severe threat in the real world, other than for ATM PINs and any kind of credentials of people of public interest. Herley and Pieters point out that the attack is not scalable via algorithms [164]. Hence, Maguire and Renaud come to the conclusion that “shoulder-surfing may well be a non-issue in authentication design” [218]. Using desktops or laptops to authenticate on regular websites, observation is rarely a major concern. Passwords are usually masked in the browser by replacing each character with an asterisk or a bullet symbol. However, masking has recently been questioned because it prevents proactive error checking on the user side [278].

**Other attacks** Apart from the attacks described above, there are other approaches that should not go unmentioned (see Table 2.2). First, people in one’s own social circle, e.g. friends and family, have considerable amounts of personal information and often physical access to devices and passwords written onto post-it notes. Although the intent is often not purely malicious, it can be easy for these people to have an informed guess of a user’s credentials. Flechais et al. use the term “spouse attack” to describe this kind of threat [109], Dunphy et al. call it a “friend attack”, while Sasse and Flechais framed it as “insider attack” [277]. Ur et al. found that users underestimate its likelihood [342]. A strong, complex password that is only memorable to the legitimate user might help in that scenario. Weirich and Sasse, however, argue that choosing an overly complex secret could make the user appear “paranoid” [368] and Flechais et al. see password sharing as an important sign of trust [109, 110].

Finally, some credentials are shared unintentionally on the web. Software developers who share open source code on the web are prone to this issue [59]. Recently, the node package manager (npm) platform realized that many of its users published their passwords with the packages<sup>18</sup>. An adversary could simply crawl public repositories on GitHub and collect the passwords in plain text. The npmjs.org operators had to invalidate the credentials to secure the accounts. One solution that reduces the severity of credential loss is using multi-factor authentication (cf. Section 2.5.3).

Security-wise, the main objective of this thesis is combating online and offline attacks, i.e., algorithmic attacks, where password coping strategies really make a difference. A panacea for all kinds of threats, each of which has warranted multiple PhD theses, is probably impossible to find.

---

<sup>18</sup> <http://blog.npmjs.org/post/161515829950/credentials-resets> (last accessed 22.12.2017)

**Table 2.2:** Threats on passwords and potential countermeasures for users and service providers (SP). Other stakeholders are left out of this analysis.

Threat	Countermeasures	Responsible
<b>Online Attack</b>	Throttling	SP
	Anomaly detection	SP
	Set-up Multi-factor authentication	SP
	Moderately complex password	User
	Enable Multi-factor authentication	User
<b>Offline attack</b>	Slow hash algorithm	SP
	Secure database design	SP
	Security audits, fix vulnerabilities	SP
	Complex, strong password	User
<b>Phishing</b>	Education and Warnings	SP
	Unique passwords	User
	Check security indicators	User
	Utilize spam filter	User
	Vigilance regarding emails	User
<b>Malware</b>	Anti-Virus software	User
	Caution on the web	User
<b>Observation</b>	Password masking	SP
	Awareness of surroundings	User

## 2.3 What is Password Strength? Metrics and Statistics

Bishop and Klein stated in 1995: “[a] good password is one that is easily remembered, yet difficult to guess.” [25, p. 231]. The second part of this statement describes password *strength* on a high level. But there are problems if we try to objectively measure the “guessing difficulty” of a password: is it difficult for an attacker with nearly unlimited resources, or only for an attacker that merely has one shot per day thanks to lock-out mechanisms? The question is highly context dependent and there is unfortunately no single true answer.

Moreover, while “{X@T;cuXw[bJnUH” is a randomly generated password that seems difficult to guess, it is not necessarily a *strong* password (see Section 2.4): if it was found in a password database after a data breach, it is likely to be included among the first guesses in subsequent dictionary attacks. The number of leaked passwords has steadily increased in the past few years and thus has rendered many such strong passwords unsuitable<sup>19</sup>. In the same vein, Yang et al. state: “[if] a strategy is widely used, then attackers may develop strategy-specific methods which can efficiently guess the passwords” [381]. Economists call this the “Tragedy of the Commons” [156]. For example, shifts in password practices imply that if passphrases (cf. Section 3.3.3) do become commonplace, attackers will optimize their

<sup>19</sup> <http://breachlevelindex.com/> (last accessed 12.04.2018)

---

attack strategy, rendering the well-intended efforts to strengthen passwords less effective. Consequently, the challenge of considering such effects and realistically measuring password strength has sparked some discussion, which has led to a small set of suitable models which are discussed below.

### 2.3.1 Entropy

In 1951, Claude Shannon put forward an information-theoretical approach that describes the encoding of English letters and words [292]. In it, he defines the term “entropy” as *“a statistical parameter, which measures [...] how much information is produced on the average for each letter of text in the language. If the language is translated into binary digits (0 or 1) in the most efficient way, the entropy  $H$  is the average number of binary digits required per letter of the original language”*. In other words, entropy represents the amount of information of a given word in bits.

William (Bill) Burr and his colleagues at NIST took to this understanding and translated it into a measure for password strength, which was officially published for the first time in 2004 [47, Appendix A]. They define “guessing-entropy” as “an estimate of the average amount of work required to guess the password of a selected user”. Apart from guessing a single user’s password, the macro perspective defines the “min-entropy” as “a measure of the difficulty of guessing the easiest single password to guess in the population”. Although many citing publications (e.g. [32, 99]) do not sufficiently point it out, Burr et al. explicitly acknowledge the limitation of using entropy as metric for user-selected passwords. They base their estimate on frequency distributions, meaning more frequently used passwords are guessed first and are thus lower in entropy.

For random passwords, the NIST guideline gives the following entropy calculation formula.

$$H = \log_2(b^l)$$

$b$  is the size of the alphabet, e.g. the 94 International Organization for Standardization (ISO)-printable characters, and  $l$  is the length of the password. For an 8-character password this would give an entropy of  $\log_2(94^8) \approx 52 \text{ bits}$ .

Most users do not use random passwords (for a detailed discussion see Section 3.2), therefore this kind of entropy calculation is rarely realistic. In other words, the effective or practical password space is much smaller than the theoretical password space. The original NIST guideline tries to take this into account and gives another calculation approach based on user behavior (see page 49f. in [47]). It lists a few entropy heuristics, e.g. the first character gives 4 bits of entropy, the next 7 characters add 2 bits of entropy each, while the 9th through 20th character only add 1.5 bits. Burr et al. do not provide a formal rationale as to why this was chosen. But again, the publication specifically points out that this is still an inaccurate approach because this would require “examining in detail the passwords that users actually



select under the rules of the password system”. Moreover, they admitted that “NIST would like to obtain more data on the passwords users actually choose, but, where they have the data, system administrators are understandably reluctant to reveal password data to others”. In the meantime, however, there have been many data leaks which fill this hole (see Table 3.1 in the following chapter).

### 2.3.2 Guess Numbers: Markov Models, PCFGs, and Neural Networks

Using such leaked data, Weir et al. developed a cracking method based on probabilistic context free grammar (PCFG) in 2009 [366]. The idea behind PCFG is that passwords often follow a rather predictable scheme (e.g. a word followed by a sequence of digits and an exclamation mark, see Section 3.2). One can model such patterns with so-called mangling rules. Weir et al.’s approach was to find a set of mangling rules that crack the most passwords in a given data set. A mangling rule takes a certain password input and creates one or more alternative versions of it, for instance the rule could be to replace all occurrences of the letter “s” with the “\$” symbol (password would become pa\$\$word). The ultimate goal is to minimize the number of guesses to crack passwords. Mangling rules also are essential for password cracking software like John the Ripper or Hashcat, but those tools primarily use them to generate word-lists, and not for guessing-order optimization. Veras et al. [348], as well as Komanduri extended Weir et al.’s PCFG approach afterwards [198].

A year later, Weir et al. evaluated the usage of the NIST entropy formulae against real-world passwords and policies [365] to address the issues pointed out by Burr et al. For the evaluation, they performed guessing attacks with PCFG on leaked password sets (most notably RockYou). For comparison, they calculated the NIST entropy for each password and correlated it with cracking success rates. Their analyses show that the calculated entropies were not satisfyingly predictive of cracking success, and they concluded that *“there is no way to convert the notion of Shannon entropy into the guessing entropy of password creation policies”*. Weir et al. thus proposed “guessing probability” as a more accurate password strength metric and the result of an analysis is a kind of lookup-table split into different probability groups.

Since the computing power required to perform the PCFG attack was relatively high, Kelley et al. tried to improve the algorithm by testing multiple training sets and variations of guess algorithms [190]. They used the mTurk platform to collect new passwords created under different policies (more on policies in Section 3.3.1) and tried to crack them. Tuning the right parameters for cracking by exploiting a priori knowledge (e.g. password policy) led to much more cost-efficient analyses, but the training still takes approximately 24 hours. However, once the Markov-chain model is generated, one can simply pass in any kind of password and retrieve its *guess number*. The higher the guess number, the stronger it is. After a certain threshold, the password is considered unguessable – which is also one of the limitations of the approach because given the time and resources virtually anything can be cracked. Kelley

---

et al.’s approach soon became a gold standard to measure password strength, especially at Carnegie Mellon University (Pittsburgh, Pennsylvania, USA) (CMU). Komanduri et al. were probably the last ones in 2011 that still used entropy as strength metric [200]. Kelley et al.’s guess numbers caught on and are still used today, perhaps because they directly represent the “*number of attempts that an attacker would need in order to guess it*” [82] or as Carnavalet put it “*the amount of effort an adversary must employ to break the password*” [48].

Almost simultaneously to Kelley et al., Bonneau developed his idea of efficient and effective guessing, which has also generated a lot of impact since then [32]. He collected leaked passwords and tried to attack them in different ways to find patterns in user behavior that could be leveraged in real-world attacks. Fundamentally, the attacks differ in the choice of dictionaries. He found that the success rates strongly depend on the dictionary that is used for it. He aimed to translate his findings to a new entropy paradigm and concluded that 10 bits of entropy are probably enough to defend against online attacks, respectively 20 bits for offline attacks.

Ur et al. took guessing to the next level by evaluating the most sophisticated approaches to date against each other [346]. They compared the performance of John the Ripper, Hashcat/oclHashcat, Markov chains/PCFG and professional password recovery companies. Moreover, they tested several configurations of the tools and checked their guess success rates for passwords created under different password policies. They concluded that a multi-tiered approach is capable of giving the most conservative metric for password strength. One of the major contributions of their work is the Password Guessability Service (PGS) that they have begun offering to the community<sup>20</sup>. After successful registration, researchers can upload a file containing plain-text passwords, e.g. after collecting them through a user study (cf. Section 3.1). The user has to provide a few additional parameters, like the policy that was utilized, which enhances the guessing efficiency and reliability. Then a set of cracking approaches are run in parallel and the analysis is sent back to the uploader. Although this has become an established, state-of-the-art tool which has already been used in numerous publications (e.g., [142, 146, 230, 285, 341, 369]), it does not come without caveats. First, passwords need to be uploaded in plain text. This entails a more careful handling and storage if they originate from a user study, because participants might in fact have provided their real-world passwords (see 3.1). Plain-text passwords also imply that users disclosed them somehow and the mere act of disclosing might already reduce the strength of passwords, which is not factored into the guess numbers. However, the team at CMU counteracts this problem by deleting uploads after at most two weeks. Besides, the analyses are time-consuming, and obtaining results can in fact take several weeks, because the system is shared with other users. Moreover, if a study aims to compare multiple password policies, each condition needs to be separately uploaded and subsequently analyzed. Finally, in personal conversations with the research team, they pointed out that certain unicode characters are not supported. This makes it infeasible for passwords that were collected in countries

---

<sup>20</sup> <https://pgs.ece.cmu.edu/> (last accessed 27.12.2017)



whose languages heavily use umlauts (ä,ö,ü). Nevertheless, it appears to be one of the best strength proxies the community possesses at the moment.

A rather novel approach that has not been integrated into the PGS are cracking algorithms based on neural networks. Melicher et al. demonstrated an opportunity to configure such algorithms to perform better than former state-of-the-art techniques like PCFG and Markov models [233]. They implemented neural networks based on Monte-Carlo simulations, i.e. “smart sampling” (cf. [82]). A thorough evaluation against PCFG, Markov-models, and word-list crackers showed that neural networks can outperform all of them if properly trained. Moreover, Melicher et al. were also able to implement their approach in JavaScript. This highlights one major advantage of using neural networks: once the model has been trained, it can be packaged and shipped to browsers, which only requires a few hundred kilobytes. They showed that this type of client-side strength estimation fares really well. One caveat is that the solution does not work “out of the box”. Developers have to train their models and this means that faulty configurations might lead to erroneous strength estimations.

### 2.3.3 The zxcvbn Approach: Lightweight, Robust, Simple

Daniel Wheeler presented an approach towards password strength estimation by providing a conservative expected guess-number [369]. Let us take a closer look at it, because it served as a strength metric for much of the work in this thesis.

Similar to PCFG and mangling rules, the idea is based on pattern matching against dictionaries and leaked password corpora. The result is a calculation of the minimum rank over a series of frequency ranked lists, i.e. a guess number. In other words, the approach is *heuristic* instead of *probabilistic*, because the rank is based on *searching* through the patterns. The pattern-ranks themselves are not necessarily based on likelihoods, but on a search sequence that may lead to different prioritization of heuristics depending on the found patterns. The implementation of the algorithm is called zxcvbn<sup>21</sup>. Wheeler showed that in an online attack scenario [115] the algorithm estimates the number of guesses accurately within an order of magnitude of 2 – consistently better than NIST guidelines to date and other lightweight strength estimators. Beyond the online-attack threshold, the results are mixed, but adding more tokens to the dictionary further improves robustness. Moreover, zxcvbn provides a numerical password score from 0 to 4. The README describes the thresholds, which are motivated through different attack scenarios (see Section 2.2):

**0** too guessable: risky password – guess number  $< 10^3$

**1** very guessable: protection from throttled online attacks – guess number  $< 10^6$

<sup>21</sup> The name zxcvbn originates from the bottom row on a QWERTY keyboard. Many users mistakenly consider the keyboard pattern approach secure because the resulting password looks fairly random.

- 
- 2 somewhat guessable: protection from **un**throttled online attacks – guess number  $< 10^8$
  - 3 safely unguessable: moderate protection from offline slow-hash scenario – guess number  $< 10^{10}$
  - 4 very unguessable: strong protection from offline slow-hash scenario – guess number  $\geq 10^{10}$

These steps are plausible and can be derived from academic literature, e.g. [32, 115, 346, 360]. One of the advantages of zxcvbn is that its implementation is fairly small (1.5 Megabytes in total). It comes with 100,000 tokens which allows zxcvbn to conservatively estimate the guess numbers. Moreover, it can be extended with site-specific word-lists. Zxcvbn performs its calculations within milliseconds, which makes it suitable to run strength analyses on the client. This is a major advantage, e.g., to provide users with real-time feedback on password strength. Moreover, it can also be easily used on the server to enforce more advanced password policies, e.g. with NodeJS. Since zxcvbn is open-source, one can modify and adapt it to meet context-dependent requirements. This makes it extremely useful for user studies: One can strip sensitive information from the analysis and save zxcvbn's output straight to a database, which is ethically reasonable and speeds up further analyses.

Despite the benefits, one needs to be aware of zxcvbn's drawbacks. Carnavalet compared several real-world password strength estimators, one of which was zxcvbn v2.x [73]. Although they concluded that zxcvbn was the best of the tested estimators, they see the major limitation in the static dictionary size. Zxcvbn ships with index terms from the English Wikipedia, English words from TV and movie subtitles, a list of roughly 47000 frequency-ranked passwords, female and male first names, and surnames from English-speaking countries. Hence, all words that cannot be matched from these wordlists including mangling rules, are seen as random strings. Zxcvbn goes on to assume these can only be cracked by brute force, but real-world attackers possessing a priori knowledge of the target population might find ways to guess these passwords more efficiently. However, Melicher et al.'s neural network suffers from similar limitations. For instance, Carnavalet and Mannan's [48] example "dolce&gabana" (an Italian luxury brand) receives a guess number  $358000010000 \approx 1e11.5$  from zxcvbn, and  $2587762225797 \approx 1e12.4$  from Melicher et al.'s tool – so both estimates are perhaps too optimistic. Finally, Carnavalet also criticized that zxcvbn did not detect reversed words, but this heuristic was later added in version 3.5.0<sup>22</sup>. So although neural networks are a viable alternative, zxcvbn is one of the best estimators currently available.

Zxcvbn's usefulness is underlined by its impact during the first years of its existence. In the industry, zxcvbn has become the standard password-checker on WordPress since version 3.7 [73], and for Dropbox (zxcvbn's author works there). In academic literature it has received praise from Ur et al. [341], Komanduri et al. [199], and Wang et al. [360]. It was also actively utilized as strength proxy in by Groß et al. to measure associations between cognitive depletion and password strength [142], by Yang et al. to study mnemonic phrase based

---

<sup>22</sup> <https://github.com/dropbox/zxcvbn/releases/tag/3.5.0> (last accessed 28.12.2017)

passwords [381], by Lyastani et al. to study the impact of password managers on password strength [217] and also recently by Al-Jaffan to build a new strength visualization [11].

All this made us confident to utilize zxcvbn in our own studies which are reported in later sections.

### 2.3.4 Other strength proxies

Apart from entropy and guess numbers, not many strength metrics have succeeded enough to become prevalent. For the sake of completeness, we briefly point out more related work on the topic. Most notably, Bonneau combined two *partial guessing* metrics to go beyond entropy and guess work [32]: From the  $\beta$ -success rate [38] and the  $\alpha$ -work-factor [255], he derived the  $\alpha$ -guesswork metric. The former roughly describes the success rate of an attacker that is limited to  $\beta$  guesses, while the  $\alpha$ -work factor denotes the minimum required guesses to crack a desired proportion of  $\alpha$  passwords. Yang et al. chose to solely rely on  $\beta$ -success rates [381], although Bonneau discourages this in his PhD thesis [31].

Beside those metrics, Dell’Amico et al. investigated using a multi-level cracking approach to find a cut-off threshold (as strength metric) when an attack becomes infeasible [81]. In another paper, they use probabilities of succeeding with different cracking methods [82]. Finally, Li et al. incorporated personal information into PCFG and suggest a new metric named “Coverage” [209]. Essentially, this proxy tells us how likely it is to crack a password if we obtain personal information of the user.

In a sense, these additional strength metrics could be subsumed under “guesswork proxies”, too. Until now, they have not seen significant attention in the community nor by the industry. Perhaps their success is a bit hampered because they do not provide a straight-forward interpretable calculation, respectively result, which makes other solutions like PGS, neural networks, and zxcvbn more attractive to researchers in HCI.

## 2.4 What is a “Bad” Password?

Having looked at strength metrics, it is fair to ask what exactly a “bad” password is. Unfortunately, the question is more or less impossible to answer, especially without context. The word “bad” is judgmental on its own and depends on whom is asked. A security expert might call a password “bad” if it only contains lowercase letters because she knows how to crack such passwords, but a regular mainstream user might call it “bad” because he cannot type it quickly enough or keeps forgetting it. So, “bad” depends on one’s own standards, knowledge, beliefs, preferences, and usage context [132, 154]. There is a fine difference between the adjective pairs “good”-“bad” and “strong”-“weak”. A weak password might still be *good*, when it fits the purpose, i.e. when it is good *enough*. For example, four-digit PINs

---

are widely used to authenticate at ATMs. Theoretically, there are only 10000 possible combinations, which could be brute-forced within milliseconds. Still, the second factor (banking card) and lock-down mechanisms after a certain number of attempts effectively prevent such attacks, so PINs are *good enough* for the purpose. As another example, a user might choose a dictionary word like “spacetime” to sign up for an online forum. If they do not reuse this password for a more important service like accessing emails, the seemingly weak password might still be *good enough* for this purpose. Zhang-Kennedy et al. [389] as well as Florêncio et al. [114, 115, 116, 117] seem to agree, and Herley and van Oorshot frame it nicely [165]:

“Fifth, when are offline attacks a threat? While dependent on implementation, access to salted hashed passwords requires attacker effort; long gone are the days when password hash files were by default world readable. A disgruntled ex-sysadmin who steals hashed passwords is the often-conjectured foe in this attack; yet, if untrusted individuals have had unfettered unaudited access to the authentication server, a site’s problems go well beyond password strength. Sixth, are there ways to protect against off-line attacks besides password strength? Mandating password changes once hashes leak might be better than strong policies at all times. Only if a leak goes unnoticed (and a password change isn’t forced) does strength potentially help. Of course, reliably detecting leaks or break-ins itself remains difficult. Finally, how much strength is required to protect against offline attacks? The bar is clearly much higher than for online attacks (assuming lockout or rate-limiting policies in place), but at what strength are attacks effectively addressed?”

So, in that vein, if we invert Bishop and Klein’s definition, we would get “a bad password is one that is hard to remember, yet easy to guess” [25]. Human capabilities must not be neglected, which is why we discuss it in detail in Chapter 3. But neither humans nor contexts are all equal, so the answer as to what a “bad” password is, remains a solid “it depends”. Still, users sometimes seek guidance (cf. Section 3.3.2) and continue to ask this question. Let us thus have a look at more practical answers rather than “it depends.” Burnett provides a list of high-level heuristics that a user can decide to employ to figure out if their preferred password choice is “bad” [46] (list is excerpted, but directly quoted):

1. If you typed your password in Google, would you get no results?
2. Are you the only person who knows the password?
3. If you have your password recorded somewhere, is it in a secure location?
4. Do you remember your password without having to look it up?
5. Can you type your password quickly without making mistakes?

Bullets 1. - 3. are solely focused on security, while the remaining two are user-centered. He provides more heuristics, however, they are questionable because they are demanding on the user's cognitive abilities. From related work, we can also derive more questions that generally aim at identifying inadequate passwords:

6. Is your password used by somebody else? [32]
7. Is your password short, that is, less than eight characters? [365]
8. Has your password leaked (check via <https://haveibeenpwned.com/>)? [369]
9. Does your password only consist of personal information that can be found on your public social network profiles? [209]

On a more opinionated level, I personally argue that a “bad” password is one that you *care about*, but do not sufficiently try to strengthen, although you are aware of the options to do so.

## 2.5 Authentication beyond Passwords

The benefits and drawbacks of using passwords have been studied extensively. Some drawbacks already become apparent when we recall Ali Baba's story: Ali Baba overhears the thieves saying the magic words and can immediately authenticate with them (a big security issue). He also told the “password” to his brother, but he fails to recall it (the biggest usability issue).

Table 2.1 shows a high-level summary of the benefits and drawbacks that come along with password-based authentication. We can observe that the drawbacks constitute important constraints and it is natural to try to remove such limitations by considering alternatives. To give the reader a sense of the alternatives, this section briefly covers the most notable approaches to authenticate users without alphanumerical passwords or with additional security measures.

### 2.5.1 Graphical Passwords & Visual Authentication Methods

In 1995, Blonder filed a patent with the name “Graphical Password” [29], which marked the beginning of a range of new authentication schemes: Graphical (or visual) authentication. The patent presented a mechanism that prompts the user to prove their identity through correctly clicking a number of locations in an image. His goal was to find an alternative to text-based passwords that is easier to remember, without losing security benefits [23, 265]. Studies in cognitive psychology have revealed that pictures are easier to remember than words, which is often referred to as the *picture superiority effect* [251]. Grady et al.

---

state that it works the best if there is an episode that serves as mnemonic device [139]. A number of studies in HCI demonstrate the benefits of graphical authentication. For instance, Chiasson et al. report on a lab study which showed that graphical passwords are easier to remember and less error-prone in the short term [52]. In the long term, however, there was no significant difference to textual passwords. Elizabeth Stobert, who dedicated a large part of her PhD work to graphical passwords [318], also found memorability benefits [314]. Besides memorability benefits, there are other advantages like personalization and tailoring schemes to specific user groups. Imran found that graphical passwords are easily understood by children [171], while Carter presented a new effective system specifically designed for older adults [49].

In the 20 years that followed the patent grant, three different approaches prevailed: searchmetric, locimetric, and drawmetric systems. In the following, we discuss them and provide a few notable examples.

**Searchmetric Systems** The idea behind all searchmetric (also recognition-based) systems is to recognize objects or images in a “challenge set” [353], i.e., to *search* for the correct image. The commercial Passfaces<sup>23</sup> system is a paragon in this domain (see Figure 2.4). The user is challenged with identifying the picture of a person in a grid of nine pictures of people. To increase security, this challenge is executed three times in a row<sup>24</sup>. The three images the user needs to identify were chosen beforehand in the *enrollment* stage. Brostoff and Sasse, found in a field study that users were able to log in with Passfaces even after a long period of inactivity [41]. However, they also report that participants spoke unfavorably of Passfaces, because of the increased authentication time. By random guessing, an attacker would have a chance of  $\frac{1}{9} * \frac{1}{9} * \frac{1}{9} = 0.001$ , i.e., one in one thousand to find the user’s chosen faces. Davis et al., however, found that user selection of Passfaces is predictable, thus reducing the effective password space [70]. Dunphy et al. created a mechanism that is based on photos [88]. In a two-week field trial they found that participants successfully authenticated in 77% of attempts, and human attackers needed between 4.5 and 7.5 observations to shoulder-surf the passwords. These numbers would need to be higher to justify more widespread adoption.

Dhamija and Perrig created a similar mechanism called “Déjà Vu” [83], where the user selects a number of random-art images instead of faces (see Figure 2.4a). Although participants in a lab study generally liked Déjà Vu, the system was never adopted on larger scope. Another alternative solution to using faces is the visual identification protocol (VIP), which is based on pictures of objects (see Figure 2.4d) [71, 72]. De Angeli et al. iteratively improved the prototype, and found that recognizing six simple and concrete objects worked best, but was still much slower than entering a PIN. Bicakci et al. also explored authenticating through clicking a sequence of icons as master password for a password manager (iP-

---

<sup>23</sup> <http://www.realuser.com/> (last accessed 29.12.2017)

<sup>24</sup> The current implementation challenges the user three times, while the original proposal included four challenges.



MAN) [22]. They found that some icons were more attractive than others and thus impaired security. An icon-based mechanism that strongly focuses on mitigating shoulder-surfing is Wiedenbeck et al.'s Convex-Hull-Click (CHC) system [373]. The challenge set consists of icons which are randomly distributed in the grid. The idea behind CHC is that the user does not click on their enrolled icons, but anywhere inside the convex-hull which is formed by the icons. In other words, the user clicks a random icon within the virtual polygon that is formed by the icons if they were connected by straight lines. Wiedenbeck et al. showed that CHC has security benefits, but the large grid of icons negatively impacted authentication times. Moreover, Golla et al., respectively Kraus et al. recently evaluated emojis instead of PINs for mobile phones (see Figure 2.4b) [136, 203]. Contrary to traditional pins, the emojis are randomly ordered in the grid, which makes their system a typical “searchmetric” approach. The idea actually originates from “Emoji passcode”, a commercial solution from Intelligent Environments<sup>25</sup>. Golla/Kraus et al. were the first to empirically evaluate the scheme and found some selection bias but also user experience benefits.

Hayashi et al. went beyond mere image recognition and additionally blurred images during authentication in “Use Your Illusion” (see Figure 2.4e) [158]. Their hypothesis was that only the legitimate user would be able to make sense of the distorted images and recognize the original picture, while an attacker can only randomly guess. Across all conditions of their user study, most participants were able to log in successfully even after four weeks. Only those who were given a system-assigned portfolio of images were slightly more troubled. Although the system was never adopted in real-world authentication, it sparked further ideas to leverage people's abilities to recognize distorted images [50, <http://arima.okoze.net/illusion/>].

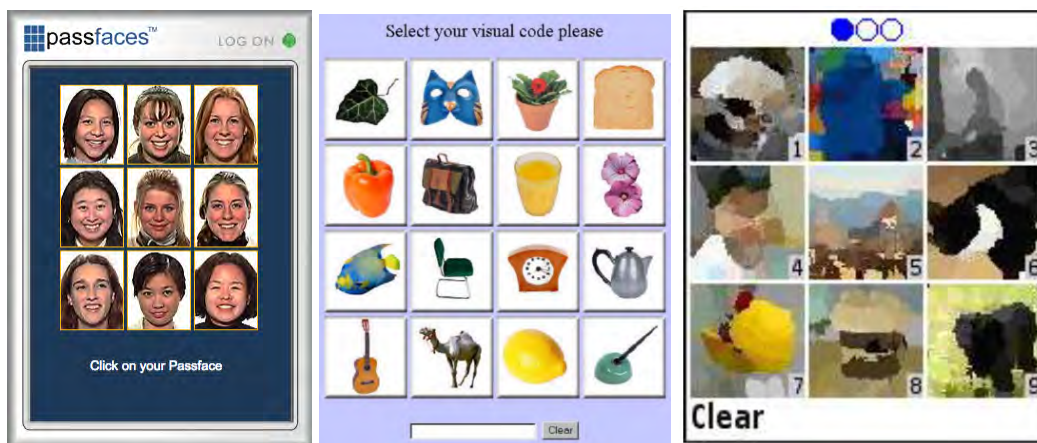
**Locimetric Systems** If the user needs to authenticate by clicking specific points in a challenge image (rather than a challenge *set*), the mechanism can be classified as *locimetric system*, from Latin *locus* = place, position. Blonder's patent application [29] is a classical example. Wiedenbeck et al. evaluated PassPoints, which in essence is one possible implementation of Blonder's idea [372]. Security-wise there were benefits over alphanumeric passwords, but usability results were mixed. Chiasson et al. combined the PassPoints and Passfaces approaches with the Cued Click Point (CCP) scheme [54], and the Persuasive Cued Click Point scheme [51] (CCP). To authenticate, the user has to click a specific point inside an image (cf. PassPoints) for a sequence of images (cf. Passfaces). The PCCP version tries to nudge the user to pick less predictable locations (one result of the CCP-study), which is supposed to further increase the “strength” of the graphical password. Success rates for log-ins were high, but there was no possibility to correct an error, so participants in the lab study had to start over. Perhaps, this is a caveat in terms of real-world adoption. Since alphanumeric passwords do have their advantages, Forget et al. later suggested allowing the user to pick the authentication scheme that best fits their current context [125]. Object Passtiles (OPT, another implementation variant of VIP) and PCCP were the two available

<sup>25</sup> <https://www.intelligentenvironments.com/now-you-can-log-into-your-bank-using-emoji/> (last accessed 29.12.2017)



(a) Déjà Vu [83]

(b) EmojiAuth / Pictopass [136, 203]



(c) PassFaces by RealUser

(d) VIP 3 [72]

(e) Use Your Illusion [158]

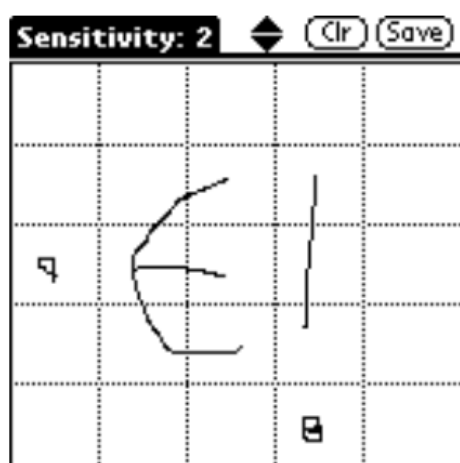
**Figure 2.4:** A selection of searchmetric graphical authentication schemes. The user authenticates by *searching* their enrolled sequence of images.

graphical authentication schemes. Interestingly, after the study participants had tried out a graphical scheme, they later switched back to text-based passwords.

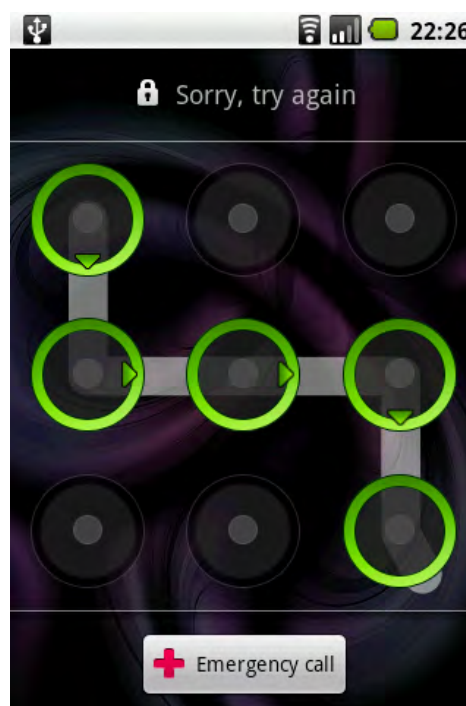
**Drawmetric Systems** A third category of graphical authentication are drawmetric systems (also known as recall-based systems). Here, the user has to either draw a shape configured during enrollment, or perform a gesture as a kind of virtual drawing. At this point, this is by far the most successful graphical authentication paradigm, because one such implementation – “pattern unlock” – was added to the Android operating system already in its early days (see Figure 2.5b) [15]. The idea of drawmetrics most likely has its origins in the Draw-a-Secret



(DAS) scheme by Jermyn et al. [180]. The user is asked to create a secret drawing in a grid of 16 cells (see Figure 2.5a). The system maps the drawing to simple  $(x,y)$  coordinate-pair sequences, but the password space, as disseminated by Jermyn et al. is large enough to provide sufficient protection of handheld devices. Dunphy et al. later aimed to improve the memorability of the drawn shapes and extended Draw-A-Secret with translucent background images [89]. In a small lab study they observed no striking memorability nor security benefits. Sherman et al. removed the grid and let the user authenticate with free-form multi-touch gestures, which required a more sophisticated matching process [302]. This may be part of the reason the system had equal error rates (EERs) between 3.34 % and 15.97%. The latter case means that a legitimate user would enter their gesture correctly and not be granted access in 16% of attempts, while an adversary would gain access just as likely. This makes the approach rather unsuitable for usage outside the lab. Another hybrid approach that combines drawmetric with searchmetric authentication is Schlöglhofer and Sametinger's GesturePuzzle [280]. As the name indicates, the authentication is based on a puzzle: upon a challenge set of images, from which the user has to recognize the ones from their portfolio, the user enters a gesture that corresponds to combination of images. Solving this puzzle appears too demanding on the user's skills and patience to be adopted on larger scale.



(a) Draw A Secret [180]



(b) Android pattern unlock (image: Mike Dent)

**Figure 2.5:** Iconic drawmetric systems. The user's password is a shape that they need to draw to authenticate.

---

However, once pattern unlock on Android had become a standard authentication technique for millions of users, a number of formal evaluations were carried out by the academic community. Harbach et al. pointed out that participants in their longitudinal study spent around 2.8 minutes on average per day to unlock their phone [155]. Interestingly, patterns were entered significantly faster than traditional PINs on average. Regarding their proneness to attacks, it was recognized early that grid-based patterns on a touchscreen are also prone to “smudge attacks” where attackers can restore the user’s pattern by looking at the oily residue left on touchscreens after entering the pattern [15]. De Luca et al. conceived an approach to authenticate on the back of smartphones and iterated the concept [79, 76]. Authenticating on the back has the big advantage to be resilient against shoulder-surfing attacks. Von Zezschwitz et al. also investigated which specific pattern types were most resilient to shoulder-surfing even if the attacker can observe them directly [356]. Since the pattern unlock on Android cannot include a given dot multiple times, Colley et al. extended it that way [58]. This approach effectively thwarts smudge attacks with a minimal change to the existing system. Moreover, Uellenbeck et al. [340], as well as Von Zezschwitz et al. [357] quantified the effective pattern space. They both concluded that user-chosen patterns are very predictable, so the hypothesized security benefits of patterns over PINs are questionable. At the same time, Von Zezschwitz et al. tried to nudge users to pick a less predictable starting position with different types of background images [357]. Their approach resembled Background-Draw-a-Secret (BDAS) [89] and Persuasive Cued Click Points (PCCP) [51]. When evaluating the concept through an online study, the effects on user selected patterns, however, remained small.

**Summary and Future Directions** Apart from the pattern unlock for on Android, graphical authentication systems have not been adopted widely. Text-passwords are still the preferred option for many service providers and are still available as authentication method on mobile devices. Maybe we need an improved understanding as to why the supposed advantages of greater memorability and security have not been able to outweigh the disadvantages (authentication times, shoulder surfing). As of now, little is known about the mental models of graphical password security. A step in this direction was Katsini et al.’s recent work [187]. They explored mental models and strategies that users employ to create graphical passwords. Thorpe et al. also found out that there are presentation effects that influence how people pick graphical passwords [334].

For the future, we could narrow down the most feasible direction for visual authentication [23]. To get there, Stobert and Biddle compared searchmetric, locimetric and drawmetric systems in terms of password memorability [314]. All participants in their study were assigned a system-generated password to reduce biases. Unsurprisingly, recognition based systems performed better than recall based systems (“recognition rather than recall is one of the most important usability heuristics [242]). Nonetheless, this seems to be the most feasible approach that might see new ideas in the near future. However, De Angeli and Renaud et al. doubt that visual authentication methods will prevail in the long run [72, 265]. They argue that the promise of more memorable passwords has not been fulfilled by the approaches so far and that we should keep looking elsewhere. Potentially, we will rely on

less knowledge-based authentication in the future, but more on biometric and multimodal approaches. Those are discussed in the following section.

## 2.5.2 Biometrics and Multimodal Authentication

Recent market analyses indicate that over 70% of smartphones will ship with a fingerprint- or other biometric sensors in 2018<sup>26</sup>. Those sensors are most commonly used for user identification and authentication. In this section we take a brief look at the different features that can be used in biometric authentication, the advantages of combining them, and where they can fail.

For biometric authentication, there exist two common categories: *Explicit* and *implicit* authentication. In an explicit authentication scheme, the user is prompted to provide the proof of identity, e.g. the fingerprint. The idea behind implicit authentication, as Jakobsson et al. describe it, is essentially to “authenticate mobile users based on actions they would carry out anyway” [178]. For example, walking, typing, or simply using the device in a specific environment.

### Explicit Biometrics

**Fingerprints** are becoming<sup>27</sup> the most common feature used in biometric authentication nowadays, due to the widespread device-support (mobile phones, tablets, laptops) and obvious usability benefits as perceived by the users [362]. On modern phones, after a short enrollment, fingerprint recognition takes a fraction of a second and has low error rates under normal circumstances [161]. Thus, fingerprint authentication is fairly usable and reasonably secure. However, a wet surface, grease, or dirt can limit the functionality [21]. Moreover, there are different kinds of attacks if the victim is not in the vicinity (like fingerprint spoofing with a 2D/3D print). Such attacks are becoming easier to carry out, though they still require decent effort [221]. But if, for example, the victim is asleep, it is enough to hold the finger onto the sensor to gain access to the device (which can indeed cause relationship fights that can force airplanes to land early<sup>28</sup>).

Recently, more systems are counting on **iris and facial recognition** to authenticate users. Windows Hello is a framework for biometric authentication that includes not only fingerprint support, but also authenticates users through their iris or face<sup>29</sup>. Mobile phones usually have

<sup>26</sup> <https://www.counterpointresearch.com/more-than-one-billion-smartphones-with-fingerprint-sensors-will-be-shipped-in-2018/> (last accessed 02.01.2018)

<sup>27</sup> <https://www.deloitte.co.uk/mobileuk/#gold-finger-fingerprints-lead-biometric-authentication> (last accessed 02.01.2018)

<sup>28</sup> <https://www.theguardian.com/world/2017/nov/08/qatar-airways-plane-forced-to-land-after-wife-discovers-husbands-affair-midflight> (last accessed 28.12.2017)

<sup>29</sup> <https://support.microsoft.com/en-us/help/17215/windows-10-what-is-hello> (last accessed 03.01.2018)

---

a front-camera facing the user, so Android has included the “Face-Unlock” feature since version 4 (2011), which was rebranded to “Trusted Face” in version 5 (2014)<sup>30</sup>. Moreover, on the iPhone X (2017), Apple has removed the home button and with it the fingerprint reader and the TouchID system in favor of FaceID<sup>31</sup>. With FaceID, the user simply picks up the phone and looks at it to unlock it. The chance of another person unlocking the phone the same way is 1 to 1 Million, according to Apple. However, it is possible to fool face-unlock systems with different spoofing attacks like using a face mask<sup>32</sup>, or in some cases a simple image of the legitimate user. All the aforementioned systems use a PIN/passcode as fallback method that dominates the biometric schemes. De Luca et al. evaluated the reasons for (not) using biometric authentication on mobile devices in an online survey [75]. They found that users prioritize usability over security when deciding which unlock mechanism to use. The respondents who had deactivated Face Unlock mostly did so for usability (read: interaction times) and reliability reasons. Interestingly, most respondents were aware of the security trade-off that face recognition entails.

Finally, a last example for explicit biometric authentication is **voice recognition** [10]. Android and Google Home are among the few consumer-oriented systems that allow users to authenticate through their voice. Hang et al. evaluated a password reset system that uses voice recognition to verify the reset request [152]. They point out the importance of considering embarrassment in the design of natural voice interaction, e.g. in an office environment. Moreover, “[b]oth face and voice recognition logins are extremely situational dependent; [...] speaking into a microphone doesn’t always work in noisy environments”<sup>33</sup>, so it is important to create systems that adapt to specific contexts [66].

## Implicit Biometrics

Implicit authentication is done without prompting or challenging the user by continuously verifying his or her identity [254]. One goal is to achieve a more natural interaction, similar to human-human interaction. Greenstadt and Beal postulate a system that recognizes people like humans do: “in most everyday interactions, we recognize people by who they are and how they behave, rather than by the secrets that they know” [141]. Of course, this excludes situations where two strangers need to authenticate one another, e.g. at a passport control at the border. A specific key benefit of successful implicit authentication would be that users do not have to spend time authenticating explicitly, which can save them a couple of minutes per day [155].

---

<sup>30</sup> <https://www.androidcentral.com/smart-lock-screen-security-options-android-50-lollipop> (last accessed 03.01.2018)

<sup>31</sup> <https://support.apple.com/en-us/HT208108> (last accessed 03.01.2018)

<sup>32</sup> <https://www.theverge.com/2017/11/13/16642690/bkav-iphone-x-faceid-mask> (last accessed 03.01.2018)

<sup>33</sup> <https://www.inauth.com/blog/fingerprints-popular-biometric/> (last accessed 02.01.2018)

The way a person walks is one of the features that can be used to recognize them, which humans also rely upon [65]. One of the early approaches to exploit gait patterns for authentication can be traced back to Ailisto et al. who equipped laptop computers with accelerometers [7]. They successfully identified users this way with an EER of 6.5%. Tamviruzzaman et al. implemented this approach for mobile phones that at the time (2009) had started to integrate acceleration sensors and gyroscopes [327]. For their “ePet” system, they also combined gait patterns with GPS-location traces, but did not report on a user study. Muaaz and Mayrhofer tackled the issue of device orientation and achieved an EER of 7.05% in the same study session, but only 18.96% across multiple sessions. As of now, gait recognition has not been widely adopted, but Android allows the user to activate “on body detection”, which keeps the device unlocked while it is in motion<sup>34</sup>. The documentation says that “some devices” can learn the gait pattern and lock the device if they detect a different walking style. As shown in Table 2.3, gait detection faces a number of problems which are difficult to overcome.

Another theme in implicit authentication is *behavioral* biometrics. For instance, the way people use their mobile phones and in which contexts can be used to build a user model [178]. Keystroke dynamics, i.e., characteristic typing patterns, can serve as feature. The first algorithms were conceived for desktop environments [63], but Bouchoux and Clarke proposed an early solution in 2008 that achieves authentication via keystrokes on smartphones [44]. They state that explicitly enrolling users might be too bothersome for them, so it should be possible to enroll a user implicitly. This in turn produces a risk of including data from impostors, too. De Luca showed that users enter grid-based patterns in a unique way, which can be leveraged for implicit authentication [74]. Their second user study focused on this type of unlock mechanism and, with an average accuracy above 90%, they could decide whether a correctly entered pattern was actually entered by the legitimate user. Interestingly, at the time of writing this thesis, De Luca et al.’s paper is the most-cited CHI publication in the past five years<sup>35</sup>, which highlights the impact and potential of this approach. What is more, this is one of the few approaches that makes explicit authentication more secure through a second, biometric layer. While the user models of the aforementioned approaches are created and maintained on the devices, Nauman and Ali proposed a system architecture that shifts keystroke analysis to a Trusted Third Party, which authenticates the user [240]. However, their approach requires more sophisticated hardware (trusted platform modules), which renders it more costly.

## Summary and Outlook

Since all biometric features, if considered independently, can be spoofed in some way (see Table 2.3), approaches to combine multiple features have started to proliferate. Although the idea of *multimodal* signals for authentication is not particularly new (some implementations were done in the mid 1990s [43]), perhaps the computing power both on the clients and back-

<sup>34</sup> <https://support.google.com/nexus/answer/6093922?hl=en> (last accessed 03.01.2018)

<sup>35</sup> [https://scholar.google.com/citations?hl=en&vq=eng\\_humancomputerinteraction&view\\_op=list\\_hcore&venue=6NNnGQq9\\_mAJ.2017](https://scholar.google.com/citations?hl=en&vq=eng_humancomputerinteraction&view_op=list_hcore&venue=6NNnGQq9_mAJ.2017) (last accessed 03.01.2018)

ends is now large enough to handle the input streams. Bigun et al. developed a framework that continues to learn from multiple signals [24], which requires such computing power. Rybinicek et al. presented a roadmap with obstacles and opportunities [274] that can serve the design of future multimodal biometric authentication schemes. Google, among other companies, tries to incorporate the notion of such multimodal implicit authentication in their Smart Lock solution<sup>36</sup>.

In summary, we can take away that biometric authentication provides many usability benefits like reduced time spent on authentication and also security enhancements. Though implicit authentication is unlikely to replace explicit authentication completely, it can still make explicit schemes more secure as demonstrated by De Luca et al. [74]. In Section 2.5.4, however, we discuss why biometrics are still dominated by knowledge based authentication especially in situations where the system cannot make a final decision due to low certainty.

Feature	Capturing Method	Implicit / Explicit	Spoofing Threats	Problems
Face & Iris	Camera	Both	Photographs of the legitimate user	Lighting situation and make-up
Fingerprint	Fingerprint reader	Explicit. Implicit imminent.	Play-Doh casts, Scotch Tape, 2D/3D prints	Dirt, Water, Finger injuries
Speech	Microphone	Both	Recordings of the user's voice	Sickness, natural voice changes, background noise
Gait	Camera or Accelerometer	Both	Imitation	Injuries, carrying load, footwear, ground surface, being seated
Keystroke	Hard/soft keyboard	Primarily implicit	Imitation	Enrollment, reliability
Location	GPS or infrastructure calculated position	Primarily implicit	Forged GPS locations	limited places, precision
Network/File/App Usage	Software protocol	Implicit	Imitation	Precision

**Table 2.3:** Comparison of different biometric authentication methods. Adapted from [319]

## 2.5.3 Multi-Factor, Token-based and Federated Authentication

As a final aspect of authentication beyond passwords, let us take a look at systems that do not necessarily aim to replace passwords, but to make them more user-friendly and/or secure.

<sup>36</sup> <https://get.google.com/smartlock/> (last accessed 03.01.2018)



**Two-Factor Authentication** In multi-factor authentication, the user needs to present multiple credentials to the system in whichever form. For instance, the Google 2-step verification mechanism combines the user-chosen password with a *one-time password* (OTP)<sup>37</sup>. If users log into their Google account on a computer in an Internet café, they will receive an OTP on a secondary device, usually inside a text message sent to their personal phone. If no more than two separate channels (“*out of band authentication*”, OOBAs) are involved, the scheme is usually referred to as Two-factor authentication or 2FA for short. Multi-factor authentication aims to prevent impersonation in case an attacker has obtained a victim’s primary credential, but not their secondary factor. There were hints to combining multiple such levels to make authentication more robust already in the late 1970s. Morris and Thompson mentioned that “some UNIX systems have instituted what is called an ‘external security code’ that must be typed when dialing into the system, but before logging in. If this code is changed periodically, then someone with an old password will likely be prevented from using it [the system, author’s note]” [237]. Moreover, Haskett shared an idea about a second level of authentication in 1984 [157]. With the *Pass-Algorithms* mechanism, the user first authenticates with their regular credentials and then proves their identity by responding to a random prompt by completing a pre-shared procedure. The algorithm is altered by the system administrator and then shared with the legitimate users, who are supposed to memorize the algorithm rather than a static password. For instance, the current algorithm could be to use the subsequent letter in the alphabet based on the challenge: If the challenge is BEL, the user enters CFM as second layer of authentication. The proposed benefit is that the algorithm is easier to remember and can still be personalized. Multi-factor authentication has since been widely adopted and more web services are following. Many banks have the users confirm transactions by a transaction number (TAN) sent to their phones, although they are already logged in with their banking password [224]. The primary benefit of multi-factor authentication lies in the increased confidence of the legitimacy of account actions. In any case, there is a small usability caveat, because adding more factors leads to longer interactions, which is not recommendable for secondary tasks like authentication [3]. However, the increased security might in turn improve user experience because it also reassures users that they are interacting with the legitimate party. Attacks on multi-factor authentication are possible<sup>38</sup>, but do not scale well [163].

Moreover, one-time passwords and “Magic Login Links”<sup>39</sup> can also replace user-chosen passwords. The user only provides their user name, and the system sends them a magic link or a random one-time password to their verified email account or phone. So, in that sense, accessing one’s personal email account is the primary authentication factor. After clicking the link or logging in with the OTP, the link or OTP expires. If implemented at scale, this could allow users to only memorize the credentials for their email account and log in to all

<sup>37</sup> <https://www.google.com/landing/2step/> (last accessed 04.01.2018)

<sup>38</sup> <https://www.theverge.com/2017/9/18/16328172/sms-two-factor-authentication-hack-password-bitcoin> (last accessed 04.01.2018)

<sup>39</sup> <https://www.sitepoint.com/lets-kill-the-password-magic-login-links-to-the-rescue/> (last accessed 04.01.2018)

---

other systems by receiving magic links. However, the approach is only secure under the assumption that the email account is well protected. It might be fair to assume that users might underestimate the risk of relying on magic links. For low-value accounts, it seems like a viable and promising solution, although trust issues on the user side need to be further addressed [272].

**Hardware Tokens and Proxies** Hardware tokens are often a special flavor of multi-factor authentication, but it is worthwhile to treat them separately, because in some cases they can act as standalone replacement of passwords, too. In general, this solution uses a hardware device to store user secrets [34]. In multi-factor environments, the RSA SecurID token<sup>40</sup> displays a random numerical which is valid for at most 60 seconds, and the user has to enter it as second factor. The YubiKey tokens<sup>41</sup> are inserted into the USB port of the computer or held against an NFC-equipped smartphone to prove the user's identity (see Figure 2.6). While the two devices improve account protection, the downside is that they are easily lost or stolen, and have to be with the user when they need to authenticate.

To save users from entering PINs, patterns, or passwords to unlock their phones, Jakobsson developed a bracelet that serves as a secure token [176], which is not as easily lost or stolen. This idea is also embraced in commercial products. Smart watches and other wearables can serve as authentication tokens: Google offers the “Trusted Device” feature since Android version 5. If a trusted device, e.g. a smart watch or wireless headphones, is connected to the phone via Bluetooth, the phone turns off the primary authentication method, e.g. pattern unlock while the device is connected. If the phone has been inactive for four hours or had been manually locked, the user needs to re-authenticate by providing the primary credential. This can drastically reduce the number of authentication events, given phones are unlocked between 27 and 86 times per day according to recent estimates[374, 311].

Other approaches leverage devices that the users carry with them anyhow as authentication token or proxy. Aebischer et al. present the Pico Framework where the user has to scan a QR code with a special app to pass-by manual authentication [6]. Some banks require the user to scan special image codes as second factor during authentication<sup>42</sup>. A few popular websites have already started to adopt the system. Roalter et al. presented a system that allowed users to access shared rooms in a university by authenticating with their smartphone [270]. There is some evidence that users generally appreciate this kind of interaction [272]. However, using the phone as proxy or second factor has the disadvantage that batteries might be empty or the phone might be out of reach.

**Federated Single Sign-On** The idea behind single sign-on (SSO) is to allow the user to log into one service and other services can use this login state without requiring any more ex-

---

<sup>40</sup> <https://www.rsa.com/en-us/products/rsa-securid-suite/rsa-securid-access/securid-hardware-tokens> (last accessed 04.01.2018)

<sup>41</sup> <https://www.yubico.com/start/> (last accessed 04.01.2018)

<sup>42</sup> <http://www.wikibanking.net/onlinebanking/verfahren/phototan/> (last accessed 04.01.2018)





(a) The RSA SecurID token displays a one-time password every 30 to 60 seconds      (b) YubiKey uses NFC to authenticate users.

**Figure 2.6:** Commercially available hardware tokens. The RSA SecurID is more business-oriented, while the YubiKeys can be used with consumer products like Gmail, Facebook, and Dropbox, too.

plicit authentication actions. Thus, the user must only remember one set of credentials [91]. With federated single sign-on, websites can authenticate users by temporarily redirecting them to a trusted third party [34]. The third party verifies the user's identity. Afterwards it signs the authentication request and returns a text-based token that the client needs to present in future interactions. A historically successful example is Kerberos [197]. However, SSO approaches did not receive widespread acceptance for about twenty years [322]. Currently, the OpenID protocol is a state of the art approach, where user identities are verified in a decentralized approach [263]. Any web server can act as identity provider, but it is most feasible if trustworthy parties enable OpenID. Currently, Microsoft, Google and Oracle are among the biggest OpenID identity providers. **OAuth** is a related protocol, but is not designed as identity proof but to allow “relying parties” to use resources on other websites [34]. For example, a web app can use OAuth to access profile information on Facebook or publish Twitter messages on the user's behalf. The web app can create another dedicated user-account for its own purposes with the data it receives from the OAuth interfaces, and automatically authenticate the user without further notice. Perhaps this is the reason OAuth is often mistakenly seen as single sign-on mechanism.

Different analyses of the security and user experience of SSO mechanisms have yielded mixed results. Ruoti et al. found that their participants ranked OpenID-like approaches as the preferred way to authenticate [272]. Sun et al., however, found that their survey respondents had often erroneous mental models about how single sign-on works [323]. For instance, many people thought that their passwords are shared with relying parties and also around 40% were concerned about privacy in this study. Contrarily, Egelman pointed out that users who used Facebook Connect are fairly cognizant of the data that is shared with relying parties [91]. Still, 15% of respondents expressed privacy concerns and refrained from using such mechanisms. Bonneau et al. also criticize big companies as identity providers

---

by noting that “Facebook Connect (a version of OAuth), incentivizes relying parties with user data, mandating a central role for Facebook as the sole identity provider, which does little for privacy” [35]. For users, leaving an identity provider like Facebook also entails tedious account recovery on relying services like Spotify<sup>43</sup>. In summary, SSO theoretically has the potential to drastically reduce the number passwords but it brings out a new range of problems.

## 2.5.4 Passwords are Here to Stay

Bonneau et al. argue that passwords are an imperfect technology that is difficult to replace, and so they answer the question as to if we still need passwords with a differentiated “yes”: “*Passwords appear to be a Pareto equilibrium*”<sup>44</sup> [35]. The industry has found ways to work around the drawbacks that passwords certainly entail. Thus, if passwords are not going away, multi-factor and multi-modal systems may turn out to be the most promising solution. The challenge is to establish a minimally privacy-invasive solution that embraces user-centered design to a higher degree than previous multi-factor/-modal approaches. Academic research can, according to Bonneau et al., carry out those foundational experiments whose costs would potentially have a detrimental effect on business successes. This might help in pushing viable solutions forward when there is enough evidence for their business merit. As Herley and van Oorshot point out: “[...] it is time to admit that passwords will be with us for some time, and moreover, that in many instances they are the best-fit among currently known solutions” [165]. They propagate a more systematic approach to make users’ lives easier instead of trying to find the single panacea that replaces passwords entirely.

---

<sup>43</sup> <https://tobiasseitz.wordpress.com/2016/12/20/how-to-use-spotify-after-leaving-facebook/> (last accessed 04.01.2018)

<sup>44</sup> By equilibrium, the authors most likely mean a Pareto efficient state. From Wikipedia: “Pareto efficiency or Pareto optimality is a state of allocation of resources from which it is impossible to reallocate so as to make any one individual or preference criterion better off without making at least one individual or preference criterion worse off.” [https://en.wikipedia.org/wiki/Pareto\\_efficiency](https://en.wikipedia.org/wiki/Pareto_efficiency), last accessed 04.01.2018

# 3

## Passwords – A User Perspective

*“his thoughts were so full of the great riches he should possess, that he could not think of the word to make it open, but instead of ‘Sesame,’ said, ‘Open, Barley!’ and was much amazed to find that the door remained fast shut. He named several sorts of grain, but still the door would not open, and the more he endeavoured to remember the word ‘Simsim,’ the more his memory was confounded, and he had as much forgotten it as if he had never heard it mentioned.”*

– Kasim’s predicament in *Ali Baba and the Forty Thieves*

Morris and Thompson were already concerned with user behavior regarding passwords in 1979 [237]. They identified that users choose predictable passwords and that this can be leveraged for attacks. So, they suggested enforcing a certain minimum password length (six characters). At the time, the users were mostly professionals who received training to operate computers and could thus also have been trained to pick less predictable passwords [218]. But as computers were introduced to a larger audience, more people were exposed to password authentication. Naturally, this also induced a growing number of attacks, and it is increasingly difficult for users to defend themselves against them (see. Section 2.2). Nowadays, password policies are in place that require not only a typical minimum of eight characters, but also mandate mixed-case letters, digits and special symbols to start with. The HCI community noticed the users’ struggle in the 1990s and that we can – and should – design authentication systems with usability in mind. Perhaps, one of the breaking points where a new school of thought turned up in the literature was a paper by Adams and Sasse in 1999 [3]. The central and novel theme in there was a shift from *fixing* the user to *acknowledging* user behavior and designing for it. The paper managed to see over 1500 citations as of writing this thesis.

This chapter looks at the literature that mostly came after this seminal work. It discusses the users’ problems, solutions, feelings, and opinions about using passwords. An essential goal is to give the reader an empathetic perspective and provide background information to understand why it is challenging to come up with viable solutions to make users’ lives

---

less frustrating. To get there, we first take a brief look at conducting user research with passwords. Hereafter we disseminate common coping strategies and solutions.

## 3.1 Methodology: Running Password Studies

Before we report on insights about user behavior regarding passwords, we take a look into running studies that focus on passwords. There are two central aspects that make collecting data particularly challenging: acting ethically and maintaining high ecological validity of the data. In fact, these two goals create an area of tension that demands a critical selection of methods. Komanduri et al. note that “ideally, password studies would be conducted by collecting data on real passwords created by real users of a deployed system” [200]. But this would mean that researchers obtain access to the user accounts that were under investigation. This is ethically questionable [92]. Maybe the researchers themselves are benevolent, but the data is precious and thus could bring attackers to the scene. Since absolute security can barely be guaranteed, it is best to avoid that users disclose their actual credentials during a user study to the researchers.

### 3.1.1 General Considerations for User Studies as Data Source

If we cannot collect the users’ real password in its original form, what is the best way to measure, e.g., cause and effect of novel interventions? There are several alternatives.

#### Password Creation Tasks

First, one asks participants to create a new password during the user study and stores these passwords as part of the dataset. This approach resolves the issue of real-world password disclosure, but introduces a number of problems. Studies should be ethical and thus transparent for the most part. Hence, the study topic should be known to the participants. However, if participants know that their passwords are studied, this could induce protective reactions to prevent giving hints about their real passwords. In that case, the participants’ selection strategy does not resemble their real-world behavior much and thus the ecological validity of the data is low [297]. Although it appears trivial, Fahl et al. suggest that in this scenario, asking the participants whether they had acted like they normally would is a suitable indicator that helps in weighting the data [103]. It is also recommendable to give users a specific scenario that allows them to immerse themselves in the task. Komanduri et al. argue that having participants create passwords for fictional email accounts leads to more authentic behavior [200]. Some users, however, are less protective and provide one of their real passwords regardless of the instruction (e.g. 26.5% in Fahl et al.’s study [103]). The result is the same as if the purpose of the study was concealed through an act of deception, which is occasionally done in psychology studies (for a discussion see [326]). For example, it can suffice to tell participants a convincing *cover story*, e.g. that the purpose of the

study is to do a usability test of a social networking site, which also happens to involve an account setup process (see [129]). The data would be ecologically valid because the cover story removes observer-expectancy and other biases, but users might still pick a password that is not representative for their usual practices. Thus, for the researchers, it is extremely difficult to tell “real” passwords and “new” passwords apart. Phishing or man in the middle attacks are sometimes carried out to collect realistic passwords. Haque et al. conducted a laboratory study where they told participants a cover story to create new accounts for popular websites [154]. The websites, however, were re-created by the researchers and stored the passwords on their own servers instead of performing actual registrations. Egelman et al. used a proxy server to intercept traffic between the users and a real online portal [99]. They also altered the websites to communicate their cover story that the password had expired. This, however, creates an ethical conundrum, if the dataset is published along with the paper. To allow others to verify that research is valid, reliable, and generalizable, a published dataset is desirable, but in Egelman et al. or Haque et al.’s studies this would put real user accounts at risk. Legal constraints like the EU’s General Data Protection Regulation (GDPR) can prevent publishing data as well. Much in the same vein, sharing research about successfully attacking passwords produces a similar dilemma. For instance, one can put forward new cracking approaches (e.g. [222, 239, 281, 366]) that potentially affect common strength metrics (see Section 2.3) – but attackers also benefit from this kind of knowledge. From an HCI perspective, one can also unfold how users select passwords, which allows optimizing cracking efficiency [365, 369].

There are several methods to avoid acting unethically in studies where users are required to create passwords. First of all, studies involving human subjects are assessed by an Institutional Review Board (IRB), especially in the United States. This is done to ensure an ethical study design that is unlikely to cause participants any harm. The IRB might mandate a thorough debriefing of participants and meticulous documentation of the experiment. In Europe, however, password studies are less commonly evaluated by an IRB (or authors fail to mention the process in their papers – often there are remarks that universities do not have such institutions like in [103]). Secondly, one can refrain from releasing the data set, even if user names are removed. In fact, almost all publications on passwords collected during a user study omit publishing the corresponding data set. Only the abstract analysis is published and this is a widely-accepted standard practice, despite the questionable reliability. A rather novel approach that reduces the likelihood of made-up data relies on the idea of publishing differentially private data sets. Here, algorithmically generated noise, which is indistinguishable from the original data, is added to the data set to preserve privacy of users. For password frequency lists, passwords could be mangled and extended by generated passwords that resemble real ones. Adversaries lack information whether the data is usable as signal or noise. This way, Blocki et al. managed to release a private frequency list of passwords at Yahoo that Bonneau had already anonymously analyzed [32]. Moreover, instead of collecting newly generated passwords in plain text, it is possible to store a hashed version. For instance, Wash et al. had participants install a browser extension that logged all form submits that included a password field [363]. To study reused passwords, they hashed and sent them over a secure connection to their servers. As long as a slow hash function

---

and a strong salt are used, this approach is uncritical. However, it merely allows observing if the hashes match on several sites. Finally, as a last option to collect password data, researchers can log meta-data about passwords. For instance, Von Zezschwitz et al. used a “meta password” that described the participants’ actual passwords [354], but which was insufficient to reconstruct the original. This description can include the number of characters, upper-/lowercase letters, digits, and even proactive strength estimations. To collect the data, participants in von Zezschwitz et al.’s study were provided with an offline password analysis tool. They entered their password into that and copy-pasted the result of the analysis into the questionnaire form. If one does not want to examine the full range of a password’s qualities, this approach is absolutely feasible. Florêncio and Herley used a similar approach for the large-scale data collection with around 500,000 participants to avoid running into privacy issues [111]. The information transmitted to the logging server was pseudonymized and contained only meta features. The only downside is that one cannot run further analyses on the passwords after the data is collected, which might be necessary, e.g., if a new strength metric is established.

## Retrospective Self Report

If one wants to refrain from having users create a new password, one can study their past behavior in different ways. In its simplest form, participants are simply asked to describe how they create passwords. Stobert & Biddle did this extensively to create the Password Life Cycle model [316] (see Section 3.2). Ur et al. used interviews to find out what users do to make their passwords stronger [345]. Das et al. found out through retrospective interviews that social contacts have a strong impact on users’ security decisions [68]. Most commonly, however, typical online surveys feature a number of questions about personal behavior and attitudes, e.g. [4, 133, 206, 269, 298]. Questions about passwords are easy to implement in a survey and respondents can always choose how much they want to share. However, one has to stay aware that social desirability lowers the reliability of the data: since news articles also do their part in shaming users for picking “bad” passwords (see Chapter 14), people may respond dishonestly about their password behavior. Many people are uncomfortable admitting their password is as simple as 12345. Another problem results from fading memories. Since most users have more than one password, it might be difficult for them to recall the correct past behavior and their motivation for it. Not only password studies suffer from this bias, but any study involving self report in general.

## Principles

From user research in Usable Security and Privacy (USEC) of the past decades, Krol et al. derive a set of general principles that researchers ought to consider when conducting experiments in security and privacy [205]. We can integrate password studies in there:

**Primary Task** Creating a password should not be the sole task in the study. Instead, participants should achieve a primary task by authenticating with passwords, e.g. using a new system for a period of time [41]. The reasoning behind this principle is that security tasks



are secondary tasks and this constraint needs to be reflected in the experiment. However, re-focusing on a separate primary task is not always possible, e.g. in surveys.

**Realistic Risk** Users should be able to realistically estimate the risk for secure interactions. As mentioned above, Komanduri et al. suggest carefully selecting real-world scenarios to achieve this [200].

**No Priming** Whenever human behavior is studied, experiments should avoid influencing and biasing participants with certain information. This avoids unnatural behavior.

**Double Blind Experiments** If possible, the person carrying out the actual experiment should not be involved in the planning and design of the study. Moreover, the participants should not know the details of the study, either. On the experimenter side, unconscious bias and influence is mitigated, and participants also do not know the “treatment” they receive (if any). Although this is a desirable goal, there is little evidence that experiments typically follow this principle.

**Context Definition** To increase internal validity, it is necessary to define the terms *threat model*, *security*, *privacy*, *usability*, depending on which are relevant for the study. A precise definition avoids misunderstanding and improves transparency, credibility, and trustworthiness of the experiment.

These principles can serve as a rough quality assessment of presented research, although in many instances, not all principles will be fully addressed.

### 3.1.2 Analyzing Password Leaks and (Semi-)Public Data

Instead of users creating new passwords, it is possible to make inferences about their behavior from already existing data (for an overview see Table 3.1). Password frequency lists are readily available on the Internet<sup>1</sup>. The sources can often be traced back to illegal attacks, which makes the use of such data somewhat questionable. However, it is a widely accepted method to contrast real-world practices and study behavior. The data set that has probably been studied the most originates from a breach at RockYou, a software development firm specialized at games for social networks. In 2009, an attacker used an SQL injection to download around 32 Million plain-text passwords. For instance, Veras et al. visualized semantic properties of passwords in this dataset and highlight the high occurrence of dates in there [349]. Wheeler relied on it to build the zxcvbn password strength estimation system [369]. More often, though, these data sets serve as training data for password guessability benchmarks. Weir et al. took the RockYou passwords to train their PCFG which served as a demonstration that entropy is not a feasible strength metric for user-chosen passwords. Afterwards, it was integrated into the training set of PGS. PGS is mostly used to gauge passwords collected through a user study, e.g. under different policies [295] or interventions [341]. In conclusion, publicly leaked data sets can often serve as ground truth for studies that aim to provide new insights.

---

<sup>1</sup> An example repository containing a wide range of leaked passwords is available under <https://github.com/danielmiessler/SecLists/tree/master/Passwords> (last accessed 09.01.2018)

**Table 3.1:** Example password leaks of the past five years (data source: <https://haveibeenpwned.com/PwnedWebsites>). Some of the data served security researchers to analyze user behavior and create more effective strength estimation algorithms. \*multiple leaks from different years

Data Source	# PWs	Year	Usage Examples
MySpace	360 M	2008*	[73, 67, 81, 209, 226, 295, 343, 348, 369, 366]
RockYou	32 M	2009	[17, 26, 30, 37, 73, 82, 194, 99, 177, 200, 233, 295, 348, 361, 365]
Dropbox	68 M	2012	[27, 129, 250]
Yahoo	0.5 M	2012*	[27, 67, 145, 170, 194, 226, 295, 346, 359, 369]
LinkedIn	164 M	2016*	[73, 115, 175, 198, 213, 348]

### 3.1.3 User Study Methods in Password Research

Research in USEC takes advantage of the toolbelt of HCI research in general. In the following, the most common methods for password studies are portrayed.

**Laboratory studies** To gain the greatest control over experimental parameters, laboratory studies are the go-to method. Both qualitative and quantitative studies are carried out in the lab, with a slight surplus of qualitative studies. Among the most common methods, one finds (semi-)structured interviews ([4, 133, 316, 345, 368]) and usability testing ([99, 123, 125, 142, 171, 228, 230, 272, 382, 355]). Password memorability can be studied in the lab ([99, 103, 122, 230, 380]). Studying short-term recall usually follows a mental-rotation task (e.g. [203, 382]). Long-term memorability studies in the lab are less common, because they require participants to return to the lab, which may be too bothersome for many. However, this may be the only option to evaluate alternative authentication schemes that are not yet mature enough to be distributed.

Although technically it is not a “lab” environment, café studies sometimes are closely related to lab studies. Von Zeszschwitz et al. collected qualitative and quantitative data on participants’ past password behavior by inviting customers in a café to join them for a free coffee [354]. The experimenter has almost as much control as in a lab to answer certain research questions. Only the surroundings may distract somewhat. Other methods like participatory design / co-creation ([61, 262]) and focus groups ([90, 151, 155, 308]) are possible, but less frequently reported than interviews and usability tests. A common drawback of lab studies lies in the high costs, time consumption to carry them out, smaller sample sizes, and reduced ecological validity.

**Field-Studies** Field studies for password research come in many flavors. **Online surveys** are among the most common methods used in the field ([132, 148, 154, 169, 206, 226, 269, 351]), due to their lower cost and comparatively easy implementation. Other advantages like increased sample size and more diversity in the data speak in favor of online surveys. Survey tools like surveymonkey.com come in handy, but usually lack seamless integration



of interactive prototypes. If a prototype should be evaluated through a survey, however, one has to either implement the entire survey structure or redirect participants from the survey platform to the prototype and back. Surveys are also the weapon of choice if there is an opportune moment that is worth studying. Mazurek et al. took the opportunity to distribute online questionnaires after a new password policy was introduced at CMU [226]. Fahl et al. profited from a similar situation at Leibniz-University Hannover, and Renaud et al. could even distribute surveys on the same topic across multiple years in this way [267]. Interestingly, there does not seem to be a special, standardized survey construct to measure the usability, respectively user experience, of password systems. Other Human-Computer Interaction (HCI) sub-fields more frequently use, for example, the NASA-TLX, Positive Affect and Negative Affect Scale (PANAS) or AttrakDiff constructs to establish comparability with other studies. Notable exceptions were reported by Kraus et al., who used AttrakDiff to evaluate emoji-based authentication [203]. The NASA-TLX was used by Fraune et al., [128], Sherman et al. [302], and Yang et al. [382]. Lately, the Security Behavior Intentions Scale (SeBIS) gains more attention, because it serves as a self-assessment that can help the interpretation of actions taken during a user study [97, 94, 363, 364].

A special kind of online studies that has been extensively used and propagated by CMU researchers leverage **crowd-sourcing** platforms like the Amazon Mechanical Turk (mTurk)<sup>2</sup>. Survey respondents are recruited by paying each one a small amount of money for a valid response. This way, increasing the sample size is straight-forward, if many users have already signed up on the platform and are eligible for the Human Intelligence Task (HIT). Workers (known as “turkers”) form an increasingly diverse population [271], which is another benefit. In password research, for instance, Kelley et al.’s high-impact work on password guessability collected around 12000 passwords using mTurk [191]. Ur et al. had participants rate the strength and memorability of a given set of passwords, which allowed them to identify certain misconceptions [342]. Mazurek et al. compared features of passwords created by turkers to real passwords of students and staff at their university [226]. They take the large similarities of the two data sets as evidence that passwords created during an mTurk study are a reliable and valid data source, so there is no urgent need to analyze passwords of a deployed system. Shay wrote a PhD thesis specifically about evaluating password policies with crowd-sourced data [293]. In many cases, e.g. [299, 295, 341], the primary task is to create a fictional account or merely a password under certain constraints, which apparently violates Krol et al.’s study principles [205]. It is especially interesting that most studies are announced as some kind of password study, probably mandated by IRBs. But Mazurek et al.’s work demonstrates that this limitation is bearable. Moreover, studying the long-term memorability of passwords is facilitated, because participants can be invited to return through an internal, anonymous messaging system. It is also possible to create more complex study designs with mTurk, e.g. if multiple device types should be used by the turkers to create passwords [232]. Despite the wide range of advantages, there are shortcomings of crowd-sourced approaches as with any study method. First, turkers are incentivized to complete as many HITs as possible on the platform to earn money. Thus, completing a

<sup>2</sup> <https://www.mturk.com/> (last accessed 10.01.2018)

---

survey by providing quick answers without reading the questions could lead to unreliable data. Instructional Manipulation Checks (IMCs) and attention check questions (ACQs) can mitigate this problem [249, 253]. Turkers are only paid if the commissioner accepts the HIT as valid, thus IMCs and attention checks are useful indicators here. Moreover, as of now, the mTurk platform can only be used in certain countries, e.g. the USA or UK. European alternatives exist, but are not yet par in terms of user base, response times, and feature set [253].

Aside from surveys, **diary studies** about passwords have proven feasible in the past. Inglesant and Sasse found out through a diary study that employees struggle with frequent password changes, which might not have become evident using other study means [172]. Hayashi and Hong used this method to analyze password re-use across different computers, services, and organizations [159]. Since password authentication is a secondary task, keeping a diary of authentication events helps participants provide reliable behavioral data. However, it requires much effort to continuously stay aware of one's actions and log them. To avoid that participants forget logging and other self-reporting bias [364], it can be worthwhile to ask them small questions in situ. This method, known as **Experience Sampling Method (ESM)**, requests short responses either in predefined intervals or when the system detects a relevant event. ESM has not seen much attention in password studies on the web (Lyastani et al. provide one exception [217]), but mobile authentication has been studied with this method [155]. Users carry their personal mobile with them almost all the time, so there is a high chance of successfully receiving the experience sample. ESM was also found useful for studies about security warnings in browsers [8, 106].

Perhaps, ESM is underused for password studies because it can be substituted well with different, less costly methods. It is possible to automatically detect relevant events and survey the participants before and/or after the **automatic data collection**. Florêncio and Herley conducted one of the largest studies to date on password habits with this method [111]. Their intention was to find out among other things A) how often people type passwords, B) how many sites share a password C) how many distinct passwords a user has, and D) how the strong the passwords are. Working at Microsoft, they were granted to utilize the Windows Live Toolbar for Internet Explorer to collect in-the-wild data from up to 500,000 users during three months of running the collection. They conceived the method of protected password lists (PPL) to avoid intruding into people's privacy – a kind of meta description of passwords which is sent to the logging server instead of the original password. It was thus not possible to trace the incoming data back to a specific user. However, there are a number of limitations this method. The authors point out that the anonymity of the incoming data-stream might have resulted in over-counting of entries. Also, it was not measured how long the actual password entries takes. If users only used regular dictionary words without any modification as their passwords, the key logging module of the toolbar would have recorded a password reuse event (PRE) every time the user entered that word – also in regular online communication. Nevertheless, the fact that users are typically focused on their primary tasks, background logging helps to collect unbiased data with high ecological validity.

A final option to study passwords in the field is to collect and analyze them in an already deployed system (**in-situ evaluation**), which would be the ideal data source, according to Komanduri et al. [200]. Brostoff and Sasse utilized a coursework system to evaluate Pass-faces as alternative to passwords [41]. Similarly, Renaud et al. used a coursework tool to evaluate the effectiveness of different password nudges [267]. Mazurek et al. gained access to passwords of their University’s Single-Sign On (SSO) and were able to break down differences in password selection behavior by departments. As one of the few exceptions from the industry, Bonneau analyzed a private password data set at Yahoo [32] and Amazon [37]. The data is highly ecologically valid and diverse if it originates from a real product or service. However, if interventions are implemented as part of a password experiment, this might have negative consequences for both the service provider and the users. For instance, in an A/B setting one intervention to influence password selection might in fact lead to weaker passwords and put users at risk. Each user is a critical potential source of revenue for service providers, so tampering with the sign-up procedure might lead to higher bounce rates and consequently financial damage. High stakes like this make it difficult for researchers to convince stakeholders to cooperate on a study. In conclusion, it is unsurprising to see only rare instances of password studies carried out with deployed systems in public environments, although the insights gained might be invaluable.

### 3.1.4 The Bottom Line: Emerging Good Practices and Tools

Using one of the study methods above is the first step to get closer to answering the research methods. However, to get the full picture of the studied phenomenon, a **triangulated** approach appears to be the only option. For example, Wash et al. combined a survey with log analyses to study password reuse and self-report issues [363]. A multi-tiered approach like in Von Zeschwitz et al.’s study helps to identify themes first (formative stage) and quantify them later (summative stage) [355]. Similarly, Huh et al. were able to refine their concept of system-initiated user-replaceable passwords through triangulation [170]. Adams and Sasse conducted qualitative interviews to follow up web survey results [3]. If constraints allow for only one method, it is recommendable to consider how to collect both quantitative and qualitative data points. For instance, in online surveys that evaluate a novel password intervention, it is always feasible to collect quantitative metrics (e.g. usability and password strength) and qualitative data (reasoning, explanations, feedback) to put study results into context [3]. Those eager to find additional starting points for USEC experiments probably find essential aspects in Krol et al.’s principles for experimental design [205]. The methods described above can be drawn on to fulfill those principles, which we try to achieve in Part II and III of this thesis.

---

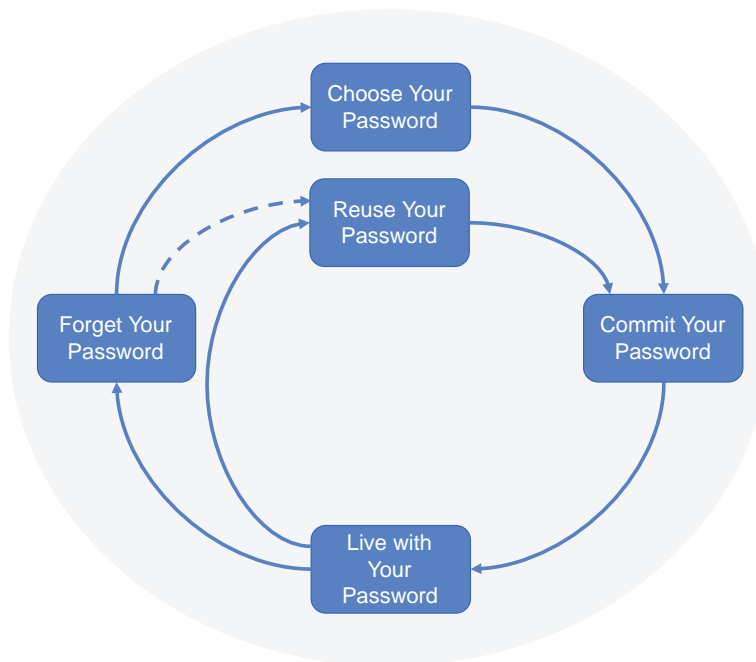
## 3.2 Password Coping Strategies and Risky Behaviors

Passwords are the cornerstone of *knowledge-based* authentication. And although “knowledge” can be stored inside and retrieved from computers, it is still a human capability to learn things and hereafter “know” them. So, humans are a large factor in the equation of knowledge-based authentication. Their actions and behavior to gain knowledge on passwords deserve to be studied in detail.

Some cybersecurity researchers started blaming system failures and vulnerabilities on users. For instance, Feldmeier et al. stated in 1990: “The main weakness in any password system is that users often choose easily guessable passwords: English words, names, trivial extensions to English words, etc., because they are easy to remember” [104]. It quickly became a dictum that users were the “weakest link” in the figurative authentication chain [276]. However, since the late 1990s, HCI advocates that systems take into account user capabilities and not the other way around [277]. Adams and Sasse postulated in 1999 that service providers acknowledge that “users are not the enemy”, which is one of the most influential position papers on the topic [3]. In that paper, they provide four central challenges in password authentication that users face: 1) Users have to deal with multiple passwords, 2) users do not intuitively create strong passwords 3) password procedures and work practices might conflict and 4) users develop a sub-par understanding of organizational security issues. Those challenges are often too hard to come by in everyday password authentication [87]. As a consequence, users develop coping strategies to reduce their task load. This early framework has since been fed with numerous research studies and is still valid today.

Stobert and Biddle formalized user challenges and behavior in the “Password Life Cycle” (see Figure 3.1), which they arrived at through qualitative interviews and coding the participants’ responses [316]. It starts out with the challenge to **choose a password**. Coping strategies at this point revolve around reducing effort, e.g. to memorize the password. Including personal or personally meaningful information comes natural to users. Others include pointers to the time they created it, or word-associations about the website-content. Mnemonics are found with some users, especially those aiming to secure their account. In essence, however, users often memorize their *coping strategy*, to recall their password. Consistent strategies reduce the effort effectively. Even if complex policies mandate a change to the first-choice password, users have a go-to strategy to deal with this situation, e.g. by appending a preferred symbol. When people create passwords, the most common action is to **reuse an old password**. This is not always possible, so users need to maintain and **commit to** a number of passwords. Hayashi et al. observed in a diary study that users categorize accounts [159] in different ways. There is also a mix of different password retrieval methods at the commitment stage. A survey in 2017 from Pew Research Center with N=926 participants found that the vast majority commits to their passwords by memorizing them in their heads (preferred strategy for 65% of the respondents) or by noting down the password (49% do this, and it is the preferred strategy for 18%) [247]. Some respondents also either

saved passwords in their browser (18%) or used a dedicated password manager (12%), but this appears to be a negligible go-to strategy (5% of the respondents). Once the user has committed to a password, they **live with it**, even if it produces difficulties in certain situations. For instance, the question “when is it time to change the password?” falls into that stage of the cycle. Finally, if passwords are not actively used on a regular basis, or were recently changed, it is very foreseeable that users **forget their passwords**. The password reset mechanism helps users cope with this situation. The two options at this point are either to create a completely new password (which increases the likelihood of forgetting it), or to reuse one (which potentially reduces the number of unique passwords). Then the cycle starts over. In the following we shed light on the actions and consequences at the different stages.



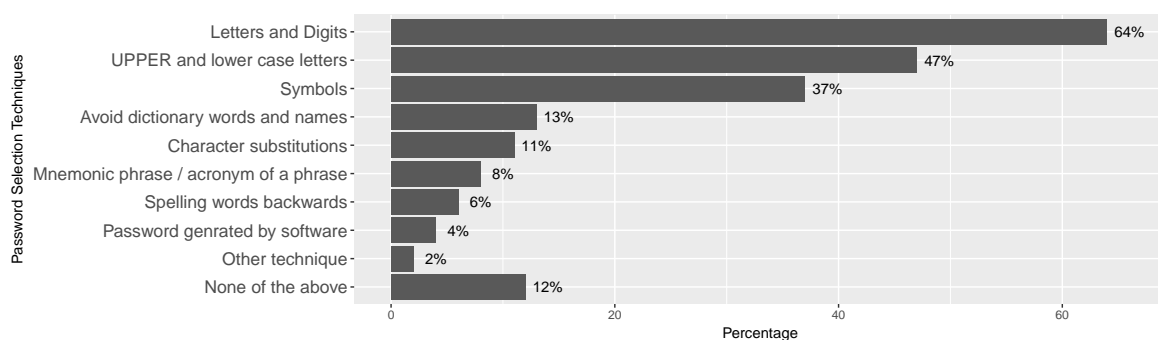
**Figure 3.1:** Stobert and Biddle’s “Password Life Cycle” [316] models typical stages of password behavior.

### 3.2.1 Weak Passwords

Why do users select weak passwords? First of all, selecting a strong password is hard for most people. Picking up the definition of a strong password (“something that is easy to remember, but difficult to guess” [25]), users do not struggle with the first part of the sentence, but the latter. Users do not intuitively know what makes a password *difficult to guess* [177], and not even security researchers have reached ultimate consensus on that matter.

Let us look at the first part: something that is easy to remember. Numerous studies have looked at what people do to make their passwords easy to remember. For instance, personal information is very memorable (“TobiasSeitz”), as is that of close ones (“LenaSeitz”)

and pets (“Fonsi&Alois”) [42, 209]. Veras et al. found that dates are very commonly found in passwords [349]. Looking at the top 25 most-used passwords<sup>3</sup>, a list published after each public data leak, we can easily spot more patterns. One group consists of “keyboard patterns” (qwerty, qazwsx) and “number sequences” (12345, 123456, 1234567489, 1234567, 123123). The remainder fall into the “likings” (football, monkey, iloveyou, starwars, dragon) and “password thematic” (Password, letmein, admin, welcome, login, passw0rd, master, hello, trustno1) categories. Figure 3.2 shows additional selection techniques. All of these passwords are particularly easy to remember, which was quantified by Chiasson et al. [52], but extremely predictable. However, they are often still allowed at many websites [288]. Interestingly, such password lists differ marginally across countries [351, 359]. Consequently, users’ **desire to create memorable passwords** naturally leads to more obvious selections.



**Figure 3.2:** Password selection techniques [186]

Another reason for the prevalence of weak passwords lies in incorrect *mental models*. Mental models are descriptions of how humans make sense of functions and system states [352]. Some users who aim to create stronger passwords still fail because their strategies to accomplish the task are predictable. Gaw et al. found that mangling is predictable [132]. Ur et al. pointed out that users understand a great deal of password security, but their modification behavior of existing passwords is sub-par [345]. Moreover, they provide evidence that users succeed to identify strong passwords, but certain characteristics are misleading [342]. For instance, including digits yielded significantly higher subjective strength ratings, but does not always effectively improve strength. What is more is that participants in their study showed a skewed understanding of password attacks. For instance, 34% of respondents thought that a strong password needs to withstand at least 50 guesses by an attacker. Therefore, trying to fend off a dictionary attack might be futile if users do not know that attackers attempt to crack the password trillions of times. Ur et al. conclude that feedback should thus inform users about attack scenarios which helps them assess the risk more realistically. Although it is sometimes argued that passphrases, i.e. a combination of multiple dictionary words, are often as secure as passwords from a richer character set [297], users fail to create strong passphrases, too. Bonneau studied the linguistic properties of passphrases at Amazon and

<sup>3</sup> SplashData publishes such a list each year, for 2017: <http://fortune.com/2017/12/19/the-25-most-used-hackable-passwords-2017-star-wars-freedom/> (last accessed 12.01.2018)



noted that an attacker could easily model user behavior to effectively crack passphrases [37]. So, in other words, the mental model “passphrase = secure” is also wrong and problematic. Finally, mental models also play a role for risk assessment in organizations. In many cases, employees underestimate the security threat that companies face day to day, which leads to insecure password practices [3, 368]. To summarize this point, users fail to create strong passwords, because in many cases **erroneous mental models** stand in the way.

What reports and news articles on password security often call “laziness” and “stupidity” [352] may in fact be the rational rejection of password security. Riley pointed out that users are well aware of “better” behavior, but they often ignore it by choice [269]. Florêncio et al. argue that this behavior is absolutely rational, because some accounts do not require strong protection and it would be impossible for users to follow all security advice given by experts [116, 264]. A case example of an account that may be valuable for some, but not for others is LinkedIn. The social-media platform which is focused on business connections suffered a severe data breach in 2012, but only became aware of a much larger leakage in 2016. Many hashed passwords had leaked and thus the affected users were prompted to reset their passwords. Huh et al. investigated the reasons (not) to reset the password and in many instances, people said that they do not use the service very often, did not want to, or were not really concerned about the risks [169]. Ur et al.’s results also indicate that users do know in theory what makes a strong password [342]. A survey commissioned by LastPass reported that one group of users often do not care about their accounts if they are not meaningful to them [208], while the other group is overly careful, but there does not appear to be a “just fine” area of behavior. The first group may believe that stronger passwords do not accomplish protection anyhow [114]. Herley and Pieters argue that it is also difficult to objectively falsify claims about security precautions, which would help to debunk unjustified security advice [164]. So, in essence, users choose weak passwords because they think **the account is unimportant** and therefore it is fine to ignore security advice.

Lastly, the literature lists a few other reasons that lead to weak passwords. Groß et al. looked at the association between cognitive depletion and selection behavior [143]. They reach the conclusion that if cognitively challenging tasks precede password selection, the resulting passwords tend to get weaker. Von Zezschwitz et al. point out that a user’s first passwords are created probably in teenage years when security precautions might be much less evident to people [354]. For important accounts, these early passwords are modified, but still persist for many years after they were first committed to. Mobile phones are another factor that steadily gains more importance. Under lab conditions, Yang et al. found that participants included more lowercase letters in passwords, if they created them on a smartphone [382]. Von Zezschwitz et al. corroborate the findings and argue that passwords created on smartphones are much less diverse than their desktop-counterparts, because they are shorter and contain mostly lowercase letters [355]. Two years later, Melicher et al. studied the interoperability of passwords on different platforms [232]. Interestingly, the passwords created on mobiles were only marginally weaker than those created on a desktop, but had more potential to lead to user frustration, especially if requirements were too bothersome. The individual keyboard on a mobile influences frustration levels, too [153]. In summary, contextual factors like

---

cognitive depletion and the device used during password creation have a notable impact on password strength.

We thus identify four main themes from related work that cause weak passwords: 1) Weak passwords are more memorable. 2) Hard-to-change mental models prevent the creation of stronger passwords. 3) In many situations, users rationally reject the effort to create a strong password. 4) Contextual factors notably influence selection strategies. In the following, we dissect another coping strategy, which is potentially even riskier than selecting weak passwords: password reuse.

### 3.2.2 Password Reuse

The primary reason for password reuse is the mere fact that users create new accounts on a regular basis, and it is logical to make sure to be able to log in later by choosing a secret they already know. “Password overload” essentially frames the problem [381]. “Memory interference” postulates reuse as a coping strategy [52]. In 2007, Florêncio and Herley conducted a large-scale study that empirically showed the challenges and coping strategies regarding this overload [111]: Users of the Windows Live Toolbar had 6.5 distinct passwords, each of which was used for 3.9 different websites. During the data collection period, users logged into 25 accounts on a regular basis, and typed around 8 passwords per day. Although keeping track of the multitude of username-password combinations is a tough challenge, most users still rely on their memory instead of other tools [247]. Users realize the challenge is hard, but Woods et al. argue that users underestimate their capabilities when it comes to memorizing passwords [377]. Consequently, many people do not try to create a richer portfolio of passwords than the numbers from the 2007 study showed. More recent numbers show that between 39% [247] and 76% [64] of users rely on reuse as coping strategy. Since social desirability bias could lead to dishonest survey responses, the “dark figure” might even be higher, because, as Inglesant and Sasse put it, “*Users see ‘good’ passwords (that are memorable and conform to the policy) as a ‘resource’, which they continue to use for new applications even if the original use is no longer allowed.*” [172]. What is more is that the user name is part of the authentication process, and users pick different aliases, pseudonyms, emails for different accounts [159].

All this is consequential for the overall online security of an individual. The more a user relies on reuse passwords, the more severe phishing attacks and data breaches become. The metaphorical “Domino Effect” describes the situation after a breach: When an attacker obtains a password from one user’s account, all the other accounts might fall with it [173]. It is enough to know even only one low-value password to crack a large part of high-value passwords via predictable mangling rules [67, 154]. Still, password reuse is difficult to mitigate, so Ives argues to look into understanding the specific approaches better [173].

One such approach is to categorize passwords by different criteria. Here, we can borrow the “Mental Accounting” theory from Behavioral Economics to describe users’ risk assessments [320, 330]. Users put their passwords into mental accounts that help them recall them later.



Wash et al. argues that frequently entered credentials are reused more likely than seldom used ones [363]. However, interviewees in Stobert and Biddle’s study reported the opposite behavior [316]. Florêncio and Herley noticed that strong passwords (in terms of entropy) are less frequently reused than weak passwords [111]. Here again, Wash et al. observed the opposite: participants in their study prioritized stronger passwords as reuse candidates [363]. If the latter is true, then this is another indicator for problematic mental models regarding risk assessment: users try to follow security advice and prefer strength over uniqueness which is typically also advised. Nonetheless, strategies (or at least results) appear to fluctuate throughout the years. Many users cluster their secrets regarding the usage purpose [154], e.g. banking, social media, communication, shopping, etc. [245, 315]. Bailey et al. argue that users do not really respect the usage purpose, but the value, importance, and meaningfulness of the account [17]. In that sense, users do not appear to care if the password ought to protect financial information if the website does not mean much to them. Radke et al. found users in their diary study tended to create unique passwords for important accounts, and reuse passwords for less important ones [260]. Still, probably all users have a “go-to password” that is tried first for an account whose value is uncertain in the beginning [316]. If password requirements disallow the preferred choice, the go-to password might not work anymore. Typically, users either pick another password from their portfolio in that situation, or they mangle their first choice until all requirements are met. As we show in Chapter 6, neither strategy is necessary if the go-to-password already shows certain features. But in case the policy mandates a change, Gaw and Felten have laid out that user-chosen mangling strategies are predictable, too [133]. Besides, policies might not be the only reason for password modifications. Over time, users sometimes are exposed to new security advice or other realizations that their previous password strategy is generally considered weak [354]. Taking the current (already memorized) credentials, and applying the recommendations to them is an obvious choice.

Florêncio et al. described reuse, categorization, memorable passwords, and mangling as “finite effort” [116]. They meticulously lay out that these strategies are not only common, but inevitable and necessary. Even experts in cybersecurity show similar patterns [216, 317]. The major difference between mainstream users and experts is that the latter are more articulate and considerate about their coping strategies.

In conclusion, we can hang on to the idea that password reuse is a necessary coping strategy. Zhang-Kennedy et al. motivate that it is not even “bad” or “risky” per se [389]. The challenge is to do it “right”. However, motivating users to alter this particular aspect is comparable to motivating a smoker to quit their guilty pleasure: Abandoning reuse does not show a visible immediate pay-off (nor does quitting smoking). In many cases, it does not cause harm, even though 16% of US-Americans have experiences someone else taking over their accounts [247] (a similar percentage of active smokers develop lung cancer<sup>4</sup>).

<sup>4</sup> <https://www.verywell.com/what-percentage-of-smokers-get-lung-cancer-2248868> (last accessed 13.01.2018)

---

### 3.2.3 External Storage

To avoid memory interference, many users resort to writing down their passwords. In its simplest form, a sheet of paper that holds user name and password suffices. Roughly half the users reportedly do this [247]. Many users also use a dedicated note-book that keeps their passwords in one place [202]. Interestingly, some manufacturers offer “password-logbooks” to help users organize their credentials. Kothari et al. collected customer reviews about the ten most reviewed logbooks and analyzed their content to derive a mental model of password security [202]. They were surprised how many people apparently use one of those logbooks and what their motivations are. For instance, customers often loved the inconspicuousness of the books and gave them away as presents. Age-related memorability challenges were also a central theme. Many people acknowledged the security risks but were unsure if a piece of software would be more secure than the book. Password logbooks can become a single point-of-failure. Digital files on the computer are often almost as accessible to local attackers (friends / spouses). In the workplace, writing passwords down on sticky notes [59] leaves the credentials wide open to anyone passing by and enforcing different behavior is difficult. Nevertheless, Herley and Van Oorschot generally advocate writing down as coping strategy, as long as the notes are stored in a fairly secure location [165].

Using a password manager (PWM) is a more sophisticated way of “externalizing” passwords. In essence, a password manager is a digital representation of the “logbooks” described above, but comes with many helpful extra features, like easy access, encryption with a master password etc. A plethora of services and tools exist in different flavors (e.g. built into browsers, third party programs, browser extensions, free vs. premium, cloud-based vs. local). Notable representatives include LastPass (freemium/subscription), 1Password (premium/subscription), KeePassX (free, one-off payment for apps), and Dashlane (freemium/subscription). Arias-Cabarcos et al. evaluated popular PWMs and suggest that Dashlane provides the the most feasible usability/security trade-off [12]. However, surveys have often revealed the low adoption rate of password managers [247], mostly because users feel secure enough with their current management habits, or due to financial hurdles and distrust [64, 102].

Lyastani et al. recently investigated the impact of using a PWM on password strength and reuse [217]. Those PWMs which included a generator had a positive effect on overall password strength and diversity in the large sample studied in-situ. But existing user strategies thwarted a boost in security, e.g. if a built-in PWM is solely used to store reused passwords. Users often benefit from automatically filled login-forms and do not have to type their passwords anymore, which is a huge usability plus. Autofill moreover mitigates most phishing attacks, because the PWM verifies the domain before filling the password field. As with analog notepads, PWMs are a single point of failure, especially in case the master password is weak. This is one of the few instances in which a strong password is recommended without restrictions. Password managers constitute a honey pot for attackers, who, e.g., exploit security vulnerabilities of the software to gain access to user passwords [34]. The situation is aggravated if passwords are synced to the manufacturer’s cloud storage, although this is rec-

ommended by Yee to enable seamless availability [384]. To summarize, password managers are a feasible solution for many users, but adoption rates are still fairly low.

### 3.2.4 Fallback Methods

Coping with a forgotten password shows more particularities of user behavior. If offered by service providers, the easiest way to handle the situation is to obtain a password reset link via email and create a new password. In the past, password resets had been responsible for a large portion of helpdesk calls [276], but self-service procedures have successfully mitigated the issue. Users tend to reuse passwords or mangle an old password when they forget and reset their credentials [316]. In some cases, however, password reset links are replaced by personal knowledge questions [33]. These often present a great risk to user accounts because of the statistical probability involved (e.g. “what was the make of your first car” has a predictable distribution), or the information is discoverable on social networks (e.g. “what is your city of birth”). Consequently, a strong primary password can be dominated by a weak fallback question and voids user efforts to secure their account. Perhaps the only strategy to maximize security within this scheme is to provide bogus answers to secret questions to fend-off statistical attacks<sup>5</sup>.

### 3.2.5 Account Sharing

Users often share their accounts, e.g. with close-ones, relatives, or co-workers [188, 298]. Security advice generally discourages sharing passwords because it increases the likelihood of leaked credentials. However, Singh et al. point out that this behavior is absolutely intentional, and does not originate from lack of understanding of the risks involved [308]. Password managers also integrate sharing features to give others quick and easy access to a set of passwords [210]. Account sharing plays a role in the overall security strategy, but personal passwords are usually less influenced by sharing considerations.

### 3.2.6 Summary

We can identify themes in coping strategies at each stage of the Password Life Cycle [316]. Coping is a natural reaction to “impossible demands” arising from Password overload [277]. All strategies can be justified from an economic point of view [116], although each of them generates security risks of varying severity. Password reuse is arguably the most severe problem, followed by selecting too obvious and weak passwords. It is critical to be consistent with one’s own reuse strategy and pick strong passwords in a select number of cases, e.g.

---

<sup>5</sup> <https://lifelhacker.com/use-fake-answers-to-online-security-questions-1821628011>  
(last accessed 14.01.2018)

---

master passwords for PWMs or central hubs like email accounts. In the following, we discuss how research has tried to influence risky behavior and support users in safe behaviors.

## 3.3 Guiding and Aiding Users

Since secure behavior does not come natural to users, there have been many attempts to make it more accessible to them and relieve the tensions between usability and security at the same time. Although systems can be changed in many ways, it is difficult to change the user. If the user needs to be kept in the loop, e.g. because constraints dictate so, we need to support them well and make careful decisions about how to support them [62]. In the following, we highlight approaches not only to make systems more usable, but also to influence user behavior regarding passwords.

### 3.3.1 Password Composition Policies

In terms of guiding users, combating weak passwords has received the most attention from research and practice. The idea of enforcing certain password requirements dates back to the 1970s, i.e. the early days of cybersecurity. Morris and Thompson acknowledged password authentication is inherently flawed in terms of usability. They suggested to make users either choose longer passwords, or systematically assign passwords to them to [237]. At the time, guessing attacks were not as powerful as they are today and it was obvious to shift responsibility to users [104]. When the Electronic Authentication guideline was published by NIST, password composition policies became the de-facto standard in attempting to make users select stronger passwords. The NIST guideline suggested passwords be at least eight characters long, include at least one upper case letter, one lower case letter, one digit, and one special symbol [47]. Moreover, passwords may not be taken from a dictionary with common words and not be permutations of the username. By looking at entropy estimates, it was argued that the resulting passwords would achieve at least 30 bits of entropy. Interestingly, this was not the only specification for policies. The guideline specifically says that passwords could also be graded with some other metric and be rejected based on their estimated entropy. At the time, there was not much evidence that user-selected passwords created under the NIST-policy were in fact strong, which set off a number of research studies, and made password composition policies one of the most studied topics in password security.

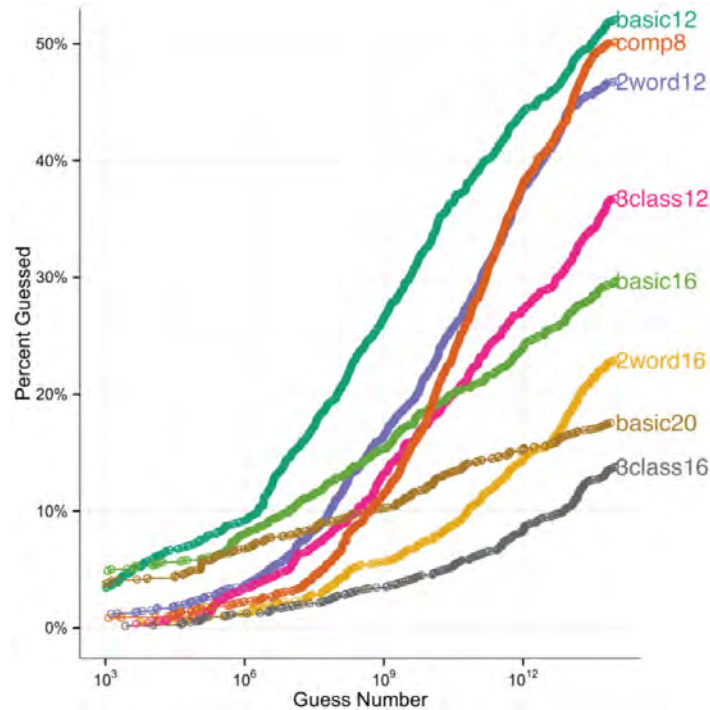
Proctor et al. found in 2002 that certain “proactive password restrictions” lead to stronger passwords [258]. In two laboratory experiments, they had participants create a new password for a university account. The policies differed in the required minimum length (five and eight) and complexity requirements like upper-/lowercase letters and digits. Requiring more complexity was comparable to increasing the minimum required length in terms of successfully cracked passwords. Interestingly, they concluded “Perhaps the most important message of this study is that restrictions on user-generated passwords may not accomplish

their intended goals.” Proctor et al.’s early hypothesis that strict policies are more or less ineffective is particularly surprising because in the fifteen years that followed, many research papers were written about such restrictions and many of them ultimately came to similar conclusions. In the following, some of the most influential works are summarized.

Inglesant and Sasse report on a diary study of password policies in corporate contexts [172]. They reached the disillusioning conclusion that password policies reduce employees’ productivity. Once the participants found a password that fulfilled the company’s policy, they used it as a resource to turn to when creating additional accounts. Similarly, Komanduri et al. found that users barely go beyond the minimum requirements of policies [200], but some policies yield better results than others. Weir et al. categorize policies into “explicit” and “implicit” policies [365]. Explicit policies have predefined rules about the password structure like the NIST guideline, which is explicitly based on lowercase, uppercase, digits, symbols (LUDS) [369]. On the other hand, implicit policies focus on strength estimation and are somewhat more volatile and intransparent to the users. For instance, if the policy uses a blacklist of forbidden words, the list is usually not displayed to users up front. The blacklist only becomes visible after the first attempt is made. Moreover, Weir et al. used subsets of the RockYou data set that fulfilled the NIST policy. Here, they provided early pointers that many of the seemingly complex passwords can be easily cracked, too.

Thus, the logical next step was to find replacements for the NIST policy with better usability and security. First attempts to find it were of theoretical nature. Blocki et al. modeled an “optimal” policy as algorithmically solvable challenge [28], which had been proposed by Shay [294]. However, while theoretically sound, empirical evidence was necessary to quantify the effects on user behavior.

During the last few years, Shay et al. have established a taxonomy for policies [200, 295, 299]: Basic policies that only mandate a certain length (e.g. basic12), character-class centric policies that require between 2 and 4 different character classes and a given length (e.g. 3class12), policies requiring phrase-like syntax and a given length (e.g. 2word16), and complex policies that have more than 4 specific requirements (comp8). Table 3.2 illustrates the specific differences as found in the taxonomy. In multiple studies they compared the guessability of passwords under different variants. They found that a 3class16 policy produced the least guessable passwords, while basic12 yielded the highest cracking success rates. For a limited attacker, who can make up to  $10^6$  guesses, basic16 and basic20 performed poorly, but they fared well at the cut-off threshold of  $10^{14}$ . The NIST-policy (comp8) performed well up to  $10^6$  guesses, but almost as many passwords had been cracked after  $10^{14}$  guesses as for the basic12 policy. This refutes the postulation that character diversity automatically leads to stronger passwords when users select them. Shay et al. point out that 28% of participants in the comp8 condition only fulfilled the criteria by adding an exclamation mark “!” at the end, which was later corroborated by Ur et al. [345]. So, as hypothesized, policies significantly influence the strength of user-generated passwords. In terms of usability, there are significant differences, too. Shay et al. examined typical usability metrics and participants’ subjective assessments. Basic12 passwords were the easiest to create, fastest to type and easiest to remember. Interestingly, the 3class12 policy was comparable in all dimensions. All in all,



**Figure 3.3:** Guess number graph for passwords created under different password composition policies. A higher percentage of cracked passwords (larger y-values) indicates that passwords were weaker. In this case, passwords adhering to a basic12 policy were the weakest, while 3class16 produced the strongest passwords.

basic16, 3class12, and 2word16 seem like the “winners” in terms of security and usability, but 2word16 passwords have high beta guess rates [299]. To combat this, Shay et al. suggest blacklisting specific sub-strings [295].

Both in the industry and public institutions, password expiration policies are commonplace [172, 53]. Forcing users to reset their users in predefined intervals was argued to mitigate threats arising from data breaches. If password leaks go unnoticed, at least the credentials expire at regular points. For users, an expiration policy drastically increases password overload and memory interference, for which the typical coping strategies (see Section 3.2) prevail: an expired “password1” quickly is reset to “password2” and so forth [388]. Thus, it has been argued that expiration is ineffective and causes users too much effort without graspable security benefits. Chiasson et al. set out to quantify the benefits of expiration and their mathematical model strongly indicated that the benefits are “marginal at best” [53]. If expiration is not enforced, security experts often advise users to manually change their password often. Zhang-Kennedy et al. reframe this rule to “change your password well” [389], i.e. users ought to change it once they suspect (or find out) a service had been compromised.

Looking at real-world policies, Wang et al. also found inconsistencies, i.e. not all web sites requiring the same password characteristics [361]. Users can get confused if multiple sites mandate different features for “security reasons”, because they might wonder who is



**Table 3.2:** Example policies from CMU taxonomy.

Policy	The password needs to ...	Example passwords
basic8 (1class8)	be at least 8 characters long	password monkey123 qwerasdf
3class12	be at least 12 characters long and include three different character classes (upper, lower, digits, symbol)	Password1234 2MonkeysBite NfJidl2kdils
2word16	be at least 16 characters long and include at least two letter sequences that are separated by a non-letter sequence.	password.unlocks 1-Monkey-Bites qwer.asdf.zxcvb.1234
comp8	be at least 8 characters long, include at least one character from each character class, and not include a dictionary word	P@ssw0rd !M0nkey1 LGtjj{Rd;w1u/

right. Florêncio and Herley compared policies of high-traffic websites and public institutions, mostly universities [112]. Surprisingly, the most influential companies enforced some of the loosest policies. The researchers argue that decisions in these companies are not only influenced by security officers (who evangelize password strength and expiration) but also by user experience experts (who point out the usability issues). In public institutions, security advisors outnumber human-factors experts, which is why policies represent an “overshoot” of security there, imposing considerable nuisance at marginal security benefits.

To summarize, policies have the power to influence password behavior. However, this influence arises from coercion and thus risks reduced usability. Users often cope by picking the easiest possible password that still fulfills the rules. This is why Florêncio et al. argue that we should not try to fix the user, but to fix the system first (in this case the password policies) [115]. Especially, service providers should be careful not to impose strict requirements and nonsensical policies<sup>6</sup>. However, if service providers feel the need to move to a stricter policy (and expire passwords at the time of policy change), Shay et al. at least provide evidence that users feel better protected afterwards, even though their password changes are well predictable [298]. Finally, NIST recently recognized that errors were made in the Electronic Authentication guideline and released an updated version<sup>7</sup>. William Burr, who was the lead author of the original policy recommendation, was recently quoted to regret his contribution. He told the Wall Street Journal “*It just drives people bananas and they don’t pick good passwords no matter what you do*”<sup>8</sup>. Shay et al.’s results, however, somewhat relieve Burr from his remorse, because some policies do help people pick “good passwords”.

<sup>6</sup> An amusing collection of nonsensical policies is collected on <https://twitter.com/PWTooStrong> (last accessed 14.01.2018)

<sup>7</sup> <https://pages.nist.gov/800-63-3/> (last accessed 16.04.2018)

<sup>8</sup> <http://fortune.com/2017/08/07/password-recommendation-special-characters/> (last accessed 14.01.2018)

---

### 3.3.2 User Education and Guidelines

Policies can be considered a means to “enforce recommendations”. Giving advice to users to educate them about password security is a softer approach. A Google search for “password recommendations” yields  $\approx 113,000,000$  results, some very brief<sup>9</sup>, others very elaborate and authoritative<sup>10</sup>. We summarized common advice and the characteristics of “bad passwords” in Section 2.4. The original NIST guideline also read more like a “how to” than a dictum for policies, although it was translated into policies after all. Sasse et al. expressed doubts about the effectiveness of user education, because it “will only work if users are motivated” [276]. We know by now that security is a secondary goal, and thus it is unlikely that people are motivated to educate themselves about it. Therefore, we have to focus on the restriction “if users are motivated” and present advice effectively in this opportune moment. Security can become the primary task, or on-par with the primary task. For instance, the moment users adopt a password manager and have to set-up their master password, they not only want to simplify password management, but also ensure that their central hub is safeguarded against attacks. Some web-accounts are of great value and users are potentially open to receive support to secure those.

We can identify four central problems with password advice. 1) There is no consensus about adequate password strength, because it always depends on the attack model. Thus, any guideline should differentiate between threat models and brief the users about them. However, this is rarely done in practice and most advice is opinionated [164]. Besides, password advice becomes outdated if new threat models prevail. 2) Reading a guide does not necessarily translate into action. Herley says users are rational in rejecting advice if it entails too much effort [162]. Forget et al. empirically showed that advice can lead to insecure behavior, too [123]. 3) Users misread password advice. Ur et al. argue that users misconstrue a statement like “adding digits, upper-case letters, lower-case letters and symbols add to the strength of the passwords” to “all passwords with digits, upper and lower case letters, and symbols are strong” [342]. Heterogeneous composition policies, and feedback can influence mental models, too. Forget et al. argue that one needs to understand users’ mental models of authentication first, before they can be effectively instructed [123]. 4) Lastly, if password advice achieves to change user behavior on a larger scale, password guessing attacks will be modeled around the recommendations, too [165]. Thus, in the long run, advice needs to be revised because attacks become too efficient.

To conclude, one needs to stay realistic about what can be achieved with password advice [117]. Zhang-Kennedy et al. provide one such realistic view on advice [389]. They revised the “character diversity” recommendation and suggest not using common passwords, predictable substitutions, or dictionary words. However, we still face the problem of com-

---

<sup>9</sup> [https://www.ibm.com/support/knowledgecenter/SS42VS\\_7.2.7/com.ibm.qradar.doc/c\\_qradar\\_niap\\_password\\_recommendations.html](https://www.ibm.com/support/knowledgecenter/SS42VS_7.2.7/com.ibm.qradar.doc/c_qradar_niap_password_recommendations.html) (last accessed 15.01.2018)

<sup>10</sup> <https://www.ncsc.gov.uk/guidance/helping-end-users-manage-their-passwords> (last accessed 22.12.2017)



municating to users what “common” passwords and “predictable substitutions” are. Zhang-Kennedy et al. propose users should come up with original mnemonics, which is a special kind of advice discussed in the following section.

### 3.3.3 Password Selection Algorithms & Memorization Techniques

Many researchers have proposed techniques and algorithms to help users create memorable and strong passwords. We have already discussed Haskett’s PassAlgorithms [157], where the user remembered how to respond to a challenge rather than a static password. Let us take a look at more such approaches.

**Mnemonics and Training** Barton and Barton were likely the first to propose mangling strategies in this context [18]. For instance, they suggested to use sentence-based mnemonics and mentally connect passwords to different cities. For instance, a sentence like “I love Paris in the Springtime” would translate into “lIPitS”. This work was seminal and some of its techniques persist in password recommendations today. Yan et al. empirically showed that there are benefits in terms of memorability and security of the resulting passwords [380]. However, Forget et al. observed that telling participants to generate phrase-based passwords can be misinterpreted and more guidance is necessary to achieve benefits [123]. Maqbali [220] and McEvoy [230] recommend using contextual or site-specific cues on websites as mnemonics.

Memorization by repetitive training has also been suggested. Bonneau and Schechter put forward a solution that is supposed to help users memorize “56 bit secrets” (for comparison, the original NIST guideline demanded 30 bit) [36]. Users first pick a self-selected password. Their system then displays a random code (or words) for each user at login-time which needs to be typed correctly into a separate field. The code becomes part of the password and was displayed with increasing delays. But participants could skip the delay by entering the code from memory. After a median of 36 log-ins 96% of participants had memorized a 56 bit secret. Despite the high success rates, it is questionable if this type of support is in the users’ interest, especially if it is deployed by more than one service. In a similar vein, Kroeze and Olivier proposed using gamification to make the training more enjoyable [204].

In terms of the security flaws, mnemonic passwords are predictably based on common phrases from movies, literature, songs, etc., but still stronger than regular passwords [206]. Nevertheless, Yang et al. demonstrate that even mnemonic phrase-based passwords can be attacked easily [381]. Thus, like Zhang-Kennedy, they highlight the importance of original, personal phrases and show how users can be instructed effectively.

**Passphrases** Another technique proposed to create strong, memorable passwords is to create passphrases. Rather than creating an acronym of a phrase, we understand them as a combination of words, e.g. the paragon “CorrectHorseBatteryStaple”. Passphrases are

---

usually longer than traditional *passwords* and thus increase password strength. Paradoxically, passphrases sometimes exceed character limits for passwords on certain websites [48]. The PGP system embraces the strength benefits and uses passphrases to encrypt private keys on the clients. Keith et al. showed that passphrases are more memorable than more complex passwords, especially if they include punctuation symbols [189]. However, participants in their study made significantly more errors, which is the biggest usability caveat of long passphrases. On devices where text-entry is cumbersome (e.g. on mobiles or smart TVs), refraining from masking entry can mitigate this problem [232].

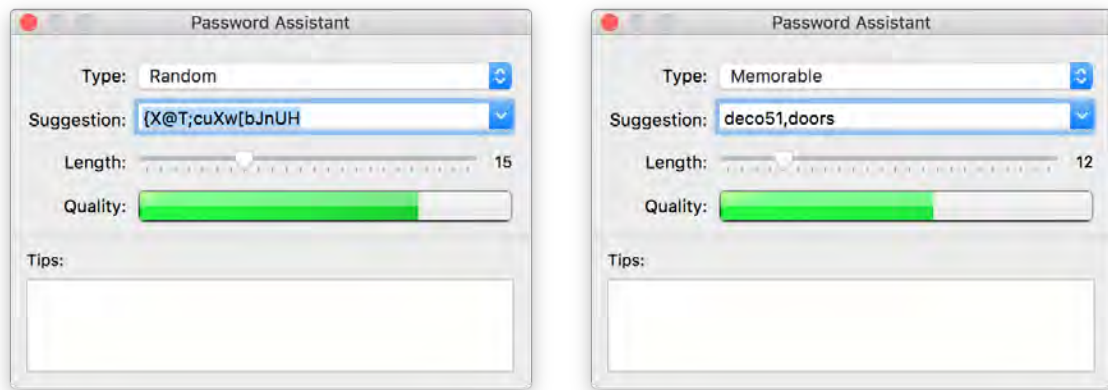
Moreover, Bonneau showed that user-selected passphrases are predictable [37]. To mitigate predictable word combinations, system-assigned passphrases have been evaluated. Shay et al. conclude that this boosts strength, but users dislike them [297]. Similarly, passphrase-centric policies (e.g. 2word16) lead to passwords with high guess numbers, but users struggle to create them [299]. The Diceware approach is often mentioned as a means to randomize word selection<sup>11</sup>. The user rolls a die five times and notes down the resulting numbers. Afterwards, they look up the word with the corresponding number from the Diceware wordlist. The process should be repeated at least twice to create a passphrase. Unfortunately, the process requires some dedication, time, and a dice. Nonetheless, passphrases usually do not require a change to the provider's system, which makes them a viable, memorable and secure option, e.g. if they are mostly entered with a physical keyboard.

### 3.3.4 Password Managers and Generators

Many of the problems around password security originate from the predictability of user choices and preferences. Therefore, systems that remove much of the decision-making process seem to remove weak passwords. Password generators can create (pseudo-) random strong passwords for the users and thus the element of human bias vanishes. There are two paradigms to generate passwords for users: easily memorable passwords, or passwords requiring external storage (see Section 3.2.3). In the first paradigm, the generator creates pseudo-random strings resembling user-selected passwords. **Pronounceable** password generators do not yield random characters but random phonetic segments [131]. This allows users to repeat passwords in their head or saying them aloud to memorize them. However, generating passwords that are consistently pronounceable is challenging [370, 135]. Passphrases can also be generated in the same way [207]. Generators implementing the second paradigm yield truly random character strings taken from an alphabet large enough to mitigate most guessing attacks (see example in Figure 3.4). Huh et al. propose letting users replace a certain number of characters to create a more memorable version of the random password [170]. However, memorization of random strings is still exceptionally arduous for most users, as is typing them. Therefore, such passwords need to be either be written down or stored inside a password manager. Most password managers (see Section 3.2.3), offer to generate passwords. Moreover, the Safari browser suggests a randomly generated password

---

<sup>11</sup> <http://world.std.com/~reinhold/diceware.html> (last accessed 16.01.2018)



**Figure 3.4:** On macOS, the Keychain application manages passwords for the user. If a new entry is manually created by the user, the password assistant can be used to generate different kinds of passwords: Random (image on the left), letters and numbers, numbers only, memorable (image on the right), and FIPS-181 compliant

to users and stores it into Apple’s cloud storage Keychain. Relying on a password manager gives users a number of usability benefits [384]. As mentioned above, auto-fill reduces interaction times and recall problems. If passwords are generated on the fly, the effort to create passwords is removed. It is easy to generate unique passwords for each account and thwart phishing attacks, so the PWM scales with the number of accounts. Some PWMs give the users feedback to assess their overall protection level. On the downside, users become dependent on the PWM, which makes it difficult to roam and use other devices. Even if users do not generate passwords, they lose muscle memory with auto-fill, so it is cognitively more challenging to log in manually.

HCI researchers have tried to solve these problems. Stobert and Biddle proposed VersiPass [315]. It uses graphical authentication and hints to avoid losing muscle memory, but it has not been empirically evaluated. Tapas is a decentralized password manager based on two-factor authentication, which was well received in two user studies [227, 228]. Yee’s PassPet is a browser extension that lets users pick a “pet name” (label) for each website they have signed up for [384]. The label is part of the password hashing and aims to increase both the security and memorability of the scheme, similar to the Password Multiplier system by Halderman et al. [147]. Fagan et al. investigated the reasons for (not) adopting a password manager [102]. They found that users more prominently appreciate usability benefits. Those who do not use a PWM distrusted the security, potentially due to a sub-par mental model. We can also hypothesize that many users want to stay independent and not give away control to a third party.

In summary, password managers aid users in password selection, and scaling the increasing number of accounts. Generated passwords are mostly a go-to method for more proficient, security-aware users who actively seek to strengthen their passwords as much as possible. While the academic research community has not been able to create PWM solutions with

---

widespread adoption, many commercial solutions exist. However, the adoption rates indicate that current systems have not been fully adapted to the masses.

## 3.4 Persuasive Interventions

Persuading users to behave differently is the last line of defense we shall discuss in this part. Using technology to persuade users can be traced back to Fogg’s seminal work on “captology”, which was later re coined under the umbrella term *persuasive technology* [120]. He defines it as “*interactive computing system designed to change people’s attitudes or behaviors*”. Persuasion itself is seen as “an attempt to change attitudes or behaviors or both (without using coercion or deception)”. The latter part further distinguishes persuasion from manipulation, where people are not aware of the manipulator’s intentions. Password composition policies exert authority and coerce users to follow the rules. The need for autonomy as part of the Self-Determination Theory [273] is thus undermined by policies. User education is voluntary and often misses its goal because users seldom decide to educate themselves on password security. Therefore, persuasion could fill the gap of providing transparent education by making alternative behavior more salient [124]. In the context of this thesis, we thus understand persuasive technology as an *enabling* technology that adequately supports users while respecting their preferences. Persuasion in HCI has become an essential topic with numerous papers published at top-tier conferences. Hekler already highlighted the rising interest already in 2013 [160]. What is more, persuasion is one of the central topics among the top-5 most cited CHI-papers of the past five years<sup>12</sup>. Consequently, we regard it as highly promising direction for password research.

### 3.4.1 Background

In the following, we explore how to use persuasion to create “soft paternalistic interventions that nudge users toward more beneficial choices” [1].

#### Terminology in Persuasion and Behavioral Economics

In the design of persuasive technology, we often encounter the concept of “nudging” people, i.e. figuratively giving them a small push to act in a certain way [332]. The “choice architect” decides on the direction of the push [333]. For instance, by setting clever *defaults*, people are relieved of making an active decision and can just accept the default. Nudging strategies as part of “soft paternalism” or “libertarian paternalism” stem from the field of behavioral economics, which studies “how individual, social, cognitive, and emotional biases influence

---

<sup>12</sup> [https://scholar.google.com/citations?&view\\_op=list\\_hcore&venue=6NNnGOq9\\_mAJ.2017](https://scholar.google.com/citations?&view_op=list_hcore&venue=6NNnGOq9_mAJ.2017)  
(last accessed 17.01.2018)

economic decisions” [2]. In other words, behavioral economists embrace the idea that people sometimes act irrationally when making economic decisions. For example, people are significantly more likely to purchase a glass of jam, if there are only six options instead of 24 – the so-called “choice paradox” [174]. As we have seen in Section 3.2, password coping strategies ultimately involve such economic decisions: Given a risk, one needs to assess the severity, likelihood, costs of mitigating the risk, and the effectiveness of the protective measure [283]. Thus, theories from behavioral economics are a useful resource that can explain user behavior.

Referring to the dual process theory, Kahneman argues that many sub-optimal decisions originate from System 1 which is responsible for intuitive and automatic thinking processes [184]. System 2, on the other hand, is the rational and effortful part in our thinking processes. He explains that most of our thinking is carried out by System 1, because it would be impossible to put the same amount of effort into every decision that we make (e.g. it is unnecessary to weigh the pro’s and con’s in answering “should I brush my teeth today?”). In password selection, both systems are involved, but depending on the context, one or the other is primarily responsible for the decision. For instance, users who have formed the habit to reuse one password for all accounts will do the same for the next account they create, thus the automatic System 1 is at play. However, if the composition policy forces the user to modify their password, the cognitive challenge rises and activates System 2. Persuasive technology aims to either facilitate decision-making (i.e. supporting System 1), or block automatic processes to help the user adopt a different behavior (i.e. activating System 2). For instance, intentionally introducing delays during password authentication [219] or in browser warnings [95] lead to users spending more time on the task and act more securely, probably because the interruption was enough to activate System 2.

### Cognitive Illusions, Biases, and Heuristics

Irrational decision-making, according to behavioral economics, shows patterns around *cognitive illusions* (also “cognitive distortions”) and *biases* [214]. Acquisiti et al. describe them as “systematic, and therefore predictable deviations from rational choice theory” [1]. The resulting behavior is neither erratic, nor irrational, but may strongly influence judgment under uncertainty [338]. To still be able to make decisions under uncertainty, people utilize *heuristics*, or mental shortcuts in decision-making. *Bounded rationality* explains the use of heuristics with the impracticality of assessing all possible options and outcomes. In other words, heuristics allow making good decisions under the circumstances [182]. Thus, like biases, heuristics are not necessarily bad, but even necessary to get through life [57]. The lack of certainty is prevalent in decisions in cybersecurity, especially for mainstream users. For example, it is hard for users to assess the risk of threats online, and make a decision on the required protection level. Therefore, heuristics in this domain have been investigated to better understand user behavior. The list of cognitive fallacies is long, Wikipedia mentions 109 decision-making biases alone<sup>13</sup> (see Figure 3.5). Let us discuss notable examples that

<sup>13</sup> [https://en.wikipedia.org/wiki/List\\_of\\_cognitive\\_biases](https://en.wikipedia.org/wiki/List_of_cognitive_biases) (last accessed 17.01.2018)





reluctant to move away to alternative schemes, although they might improve system security (see Section 2.5). Similarly, users who have committed to a password and reused it many times, might see the time spent as sunk cost. Thus, they do not switch to a more secure alternative even if they know there are plenty. Arguably, the **Status Quo bias** plays a role in both examples, too: people tend to favor a pre-existing state of affairs over potentially uncertain changes.

The **availability heuristic** is observed when people over- or underestimate the likelihood of events after being exposed to salient information. For instance, users might estimate the likelihood of being hacked as higher after someone in their social circle reports such an incident: the possibility has become more salient and available, thus future occurrences are seen as more probable. Das et al. partially confirmed such behavior through user interviews [68]. Certainly, **recency illusions** amplify the availability heuristic, i.e. a user who just found out about the “hacking threat” might think that this threat had just emerged.

The **anchoring** bias creates reference points (or baselines) for decisions, especially comparisons [124]. After a password breach, media reports typically mention some of the most-used passwords. A news article can anchor the reader on the strength of obvious passwords like 12345. In many cases, their own password(s) seem much stronger by comparison, but might not be seen as strong if assessed separately. Similarly, if many people show insecure password behavior, this may reinforce an individual’s practices through the **bandwagon effect**: “if so many people behave insecurely, why would I act differently?” [5].

People can also be biased by the way information is presented, which is known as **framing effect** [350]. The wording is key here and can generate **preference reversal** and other inconsistencies [167]. For password authentication, framing can be found in educating users about passwords. Zakaria et al. [387] suggest instructing users through real-world security comparisons. Framing consequences for users around losses may trigger **loss aversion** – a tendency to perceive losses as more valuable than equivalent gains [13]. Garg and Camp argue that security is currently framed as a *definite* loss for end users, while “the risk of not investing in security is a *probable*”, thus uncertain loss [130]. Cialidini shows the importance of crafting *normative* messages, that refrain from emphasizing that a socially unacceptable behavior is still widespread [56]. For instance, a normative message after a policy rejects the user’s first-chosen password might explain: “The password is weak, because it is easily guessable for hackers” (emphasis on the adversary) rather than “..., because it is something many people would choose” (emphasis on the social in-group). If normative messages are framed badly, this can lead to the **backfire effect**, i.e. people behaving in the opposite way as intended. Thus, Weirich and Sasse see framing the attacker effectively as important opportunity for persuasive password education [368, 367].

As a final example of behavioral biases, it has been observed that people’s motivation increases as they get closer to finishing a task. Hull coined this behavior the “**goal-gradient hypothesis**” [195]. Behavioral economists have found it to induce irrational behavior: Kivetz et al. handed out two different designs of “coffee cards”, that customers can fill up with each purchase [195]. Once the card is full, they can redeem it for a free coffee. One of

---

the designs had ten stamp-fields, while the other had twelve fields, two of which were already stamped. So, in both cases, customers received a free coffee after ten purchases. Surprisingly, more cards with twelve fields were redeemed after the trial period. People (wrongly) thought to be closer to goal, and thus were more motivated to achieve it. This “goal distance model” is sometimes used by password meters, i.e. a visualization of password strength (we discuss them in great detail in Section 3.4.3). Password strength visualization can give the user a “head start” to motivate them to fill up the entire bar and achieve a “strong password”, i.e. a full loyalty card.

### 3.4.2 Persuasive Design Patterns in Usable Security and Privacy

Phenomena involving biases and heuristics are often directly translated into a persuasive strategy. It is interesting that apparently the lines are blurred between the two concepts. Anders Toxboe collected a number of persuasive design patterns on his website<sup>15</sup>, which highlights this blend of psychology and persuasion. There is, however, no such list for patterns specifically for interventions in usable security and privacy. Therefore, we start with the discussion of general frameworks and put them into the context of security and passwords.

#### General Frameworks

Persuasive Design (PD) has been tried to formalize in several frameworks. Fogg’s Behavior Model posits that three elements must converge to achieve a target **B**ehavior: **m**otivation, **a**bility, and **t**riggers (B=mat) [119]. For passwords, this implies that people need to be motivated to protect themselves, at the same time they need to know how to achieve protection and be exposed to a trigger. Triggers work best when presented at the *opportune moment* (or *kairos*) [215]. Lockton et al. proposed the “Design with Intent” framework as a toolkit for persuasive interventions [215]. Jameson et al. show how to pick strategies from various toolkits to create a persuasive choice architecture [179]. Cialdini’s generic “**six weapons of influence**” have been used in the design of persuasive technology [57]. He lists authority, scarcity, liking, social proof, commitment & consistency, and reciprocity as the principles of persuasion. Let us walk through their potential usage in password authentication. The authority principle says that people are likely to follow the instructions of an authority, which explains why the NIST guidelines have been widely adopted despite their downsides. Scarcity drives motivation to act quickly to avoid losing an opportunity (see loss aversion), and framing effects are used to communicate the scarcity of a resource. For instance, the “419 scam” tactic in phishing emails frames time as a scarce resource: “your account has been intruded, if you don’t log in now, you lose access to it permanently” [312]. In this case, persuasion is used for malicious purposes. People prefer to comply if they *like* their counterpart. In theory, this would make password requirements less bothersome if used on a website

---

<sup>15</sup> <http://ui-patterns.com/patterns> (last accessed 21.01.2018)



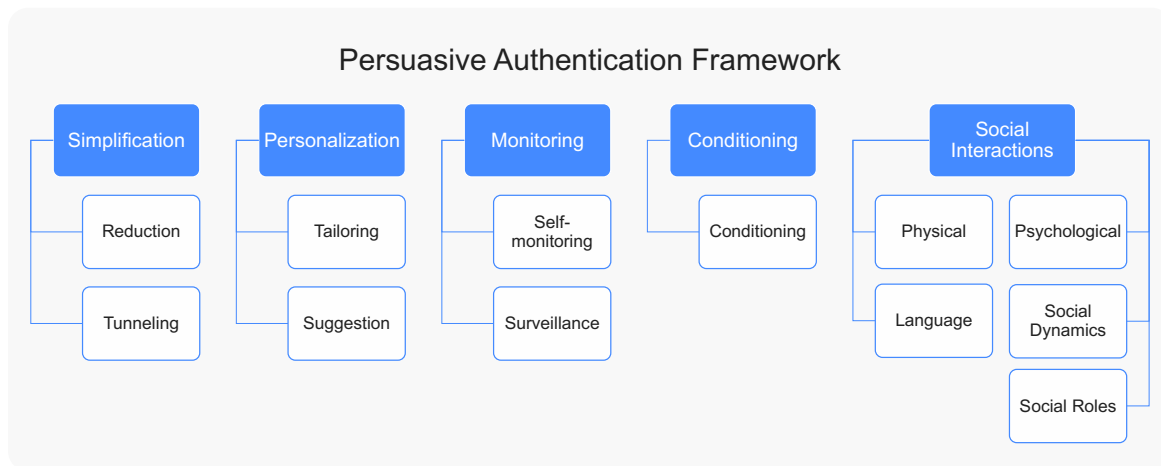
that users like, or in a more aesthetically pleasing way. Although there is no empirical evidence that the principle works for registration forms specifically, other HCI research points in this direction [121, 336]. The social proof strategy makes other people's behavior more *available* because in decision-making people tend to copy others' decisions. For instance, it might help to point out that millions of users are already using a password manager in a news article about a new password leak. We can also apply Fogg's behavior model here: Reading about a password leak *motivates* users to protect themselves, the social proof strategy acts as a *trigger*, and mentioning the ease-of-use of password managers gives users confidence about their ability to adopt a PWM. The commitment & consistency principle is an immediate part of the Password Life Cycle [316]. People try to be consistent with their past behavior and their past values. Thus, a small step towards breaking an insecure habit might, in fact, induce more secure behavior afterward. If a policy forced a user to stop using a common password, the effort to commit to the new password might spread over to other accounts, too. Stobert found that cybersecurity experts show higher signs of consistency [317]. Thus, if a persuasive strategy achieves that users commit to a new password selection scheme, their behavior might become more consistent and thus be elevated to the expert level. Finally, reciprocity describes the desire to return a favor. To use it as a persuasive design strategy, one has to do something favorable for the users first. Thus, offering users to store their passwords in a secure place and automatically logging them in is a good foundation to ask them later to turn on 2-factor authentication to help secure their account.

### The Persuasive Authentication Framework (PAF) by Forget et al.

Forget et al. embraced persuasion as technique to help users act more securely in password authentication [124]. In 2007, they put forward the Persuasive Authentication Framework (PAF) (see Figure 3.6), which breaks down the dimensions of persuasive design specifically for password support systems. In the following, we describe it on a high level and illustrate its components with results from empirical studies.

The **simplification** principle posits to reduce the number of tasks to achieve an overall goal. Thus, the hypothetical distance between the start state and the goal is shorter, and goal-gradient effects become visible. Password managers usually simplify authentication by reducing the best-case interaction from “recall-enter-submit” to “submit”. Fagan et al. showed that simplification is the primary reason to adopt a password manager [102]. But even without a PWM, both the “recall” and the “enter” sub-tasks can be simplified. Recall is facilitated with consistent personal password strategies. Entering is easier, if the primary password is replaced by a less complex one-time password.

**Personalization**, as understood in the PAF, encompasses tailoring the experience to an individual user, or to suggest actions. A recently emerged design pattern spreads out the log-in process across two separate steps for username and password, rather than having the two input fields visible at once. After submitting the username, the page is *tailored* to the user by showing them their profile picture and/or another piece of non-critical personal information. At the same time, this establishes a light-weight trusted path [383]. Wilkinson et al. suggest personalizing privacy by design by giving users different levels of control to depending on



**Figure 3.6:** Elements of the persuasive authentication framework [124]

their privacy profile [376]. Besides, this is a good example of the paternalism in persuasive interventions, because the designer suggests what option is “better” for the individual user. Similarly, it is conceivable to personalize password policies depending on the user’s “password profile” (I elaborate on this idea in Section 14.1.2). Moreover, personality constructs have been investigated to tailor user experiences, which perhaps exceeds the original proposition in the PAF. Recently, Egelman and Peer advocated the use of psychometric cues to contextualize privacy or security messaging, which they call the “next frontier in privacy and security research” [98]. Jeske et al. investigated user profiles for the susceptibility to security nudges [181]. In their experiment, they used *salience* to steer users away from insecure wireless networks. They found that participants with low impulse control were more susceptible to nudges. The finding can be interpreted that nudges directed at System 1 seem more effective. There is also preliminary evidence that susceptibility to phishing, i.e. a social nudge, is also related to personality [148]. It stands to reason that password selection is related to personality, too (see Chapter 7). If this were the case, a new spectrum of personalization strategies arises. Haque et al. have already proposed a psychometric construct to identify such associations [153]. Forget et al. list **suggestion** as part of the personalization dimension, because suggestions might be based on a personal context factors. They provided a number of cases studies on suggestive password alterations [122, 126, 127]. The “persuasive text passwords” (PTP) system takes a user’s password and either randomly inserts characters or replaces existing characters to increase strength. The high resemblance to the original password is supposed to maintain memorability [122]. Shay et al. later re-evaluated this personalization approach in an mTurk study, which yielded mixed results [300].

Arguably, looking over users’ shoulder while they authenticate increases the chances that they comply to security policies, given that users are aware of being observed. **Monitoring**, however, can entail reduced user experience (cf. need for autonomy [273]). Nevertheless, if used adequately, it can serve as *authoritative* strategy (cf. Cialdini’s six weapons of influence

[57]). Google’s password alert<sup>16</sup> is a moderate approach to secure users’ Google accounts. It is a browser extension that alerts the user if they type the password of their Google account on unrelated web pages. Hence, the idea is to make sure users use a unique password for their Google account, which is a central hub for various services. To work properly, the browser extension needs to access every keystroke. Some users might distrust it because, although it is open source, it is hard for them to tell if their private data is collected. Prospect theory tells us that in this judgment under uncertainty, losing private data is attributed a higher value than the potential security gain. A less privacy-invasive design element that might help in relieving uncertainty is *self-monitoring*. Self-monitoring through *feedback-loops* – another design pattern [211] – allows users to reflect on their past actions and derive alternatives to attain their goal. Some PWMs create a “security score” for each user and persuade them to improve it.

Users rarely develop habits to act securely. Thus, Forget et al. argue to use various forms of reinforcement to persuade people to develop such habits. Rewarding users for secure password behavior serves to **condition** them. Google Drive, for instance, offered users more cloud storage quota if they completed a two-minute security check-up<sup>17</sup>. Positive feedback during password selection can condition users, too. Other rewards, e.g. longer session expiration or faster system response, have not seen significant uptake.

Lastly, the PAF includes **social interactions** as persuasive design strategy. Forget et al. posit that authentication systems mimic, e.g., the users’ language to convey “team spirit” [127]. Weirich and Sasse [368], respectively Sasse and Flechais [277], similarly describe authentication as *socio-technical system* that follows a social protocol. DiGioia and Dourish formulated the *social navigation* pattern [86]. To perceive the system as capable communication partner, signs of previous interactions inspire trust. Egelman et al. tried to nudge users during password selection through the *social proof* strategy [99]. A visualization informed the study participants how well their password fared compared to other users, e.g. “your password is stronger than that of 85% of our users”. They did not find evidence that this persuasive strategy influenced people, but maybe the approach could have been more focused on the *proof* aspect, rather than *competition*. The normative message can be read as “other people’s passwords are bad, but many people act this way”. This could actually evoke backfire effects through social proof: although users see their password is stronger, they believe that the social norm is to pick weaker passwords, which makes them conform to the social norm. In fact, Weirich and Sasse have provided empirical evidence for such behavior [368]. Social interactions are perhaps one of the most powerful design elements to persuade users: Das et al. reported that radical behavior changes often occurred due to social processes [68]. They argue that it is critical for users to observe experts in their social circle to raise their awareness and motivation for cybersecurity. Thus, persuasive design could make expert behavior of known peers more visible

<sup>16</sup> <https://github.com/google/password-alert> (last accessed 20.01.2018)

<sup>17</sup> <https://twitter.com/googledrive/status/697104410296455168> (last accessed 20.01.2018)

---

In a number of ways, some ideas from the Persuasive Authentication Framework were rather optimistic. From today's point of view they appear questionable (e.g. conditioning users like animals). Much evidence from the ten years that followed its publication shows that persuasion does not always work as intended, so maybe a few suggestions were a bit naive. Renaud et al. evaluated eight nudges that were supposed to make students create stronger passwords [267]. They reached the disheartening conclusion that none of them work. Nevertheless, certain aspects have caught on and are actively used, like Google's Password Alert which is a direct implementation of Forget et al.'s ideas. We can conclude that the PAF has matured over time, with certain components receiving more weight through empirical evidence, and others becoming obsolete. Thus, a revision could incorporate our newly gained understanding of persuasive authentication.

## Dark Patterns

At the outset of this chapter, we discussed ethics as critical factor in password studies. Using persuasion inherently becomes ethically problematic if the intention of the influencer is concealed, either by poor design or by choice. Persuasive techniques are used to manipulate, too. Many social engineering scams on malicious websites use persuasion techniques. Muscanell et al. investigated the misuse of Cialdini's weapons of influence in cyber scams [238]. They found that scammers exploit all of them in social engineering attacks. Muscanell et al. also propose mitigation strategies, but do not lay them out in detail. Attacks that take advantage of "scarcity" as persuasive strategy often rely on *fear appeals*, e.g. "Your computer is infected, download this software *now* to remove all threats". Xu et al. proposed using similar fear appeals to nudge users towards anti-spyware measures [379], and Vance et al. utilized the "time to crack" as fear appeal during password selection [347]. There are a number of such "dark patterns", which Nodder dismantles in his book "Evil by Design" [243]. Some of them are relatively benign, others are highly manipulative. DarkPatterns.org summarizes them briefly, e.g. *misdirection* - where the design aims to focus the user's attention on a decoy to plant something on them<sup>18</sup>. However, Sasse strongly urges to resist them in the design of password persuasion because they wear off over time and often exaggerate the risks [275]. Hence, dark patterns most likely turn out counterproductive.

### 3.4.3 Password Meters: Persuasion at Play

The most prominent and widespread persuasive strategy directed at passwords are password strength meters (PSMs). Most commonly, they proactively estimate the strength of a password at entry time and visualize it. Often, verbal feedback accompanies the visualization. They have been used on a multitude of websites, in password managers, and as standalone tools. When they are used on a website or in a PWM, there are a number of persuasive patterns at play:

---

<sup>18</sup> <https://darkpatterns.org/types-of-dark-pattern> (last accessed 20.01.2017)

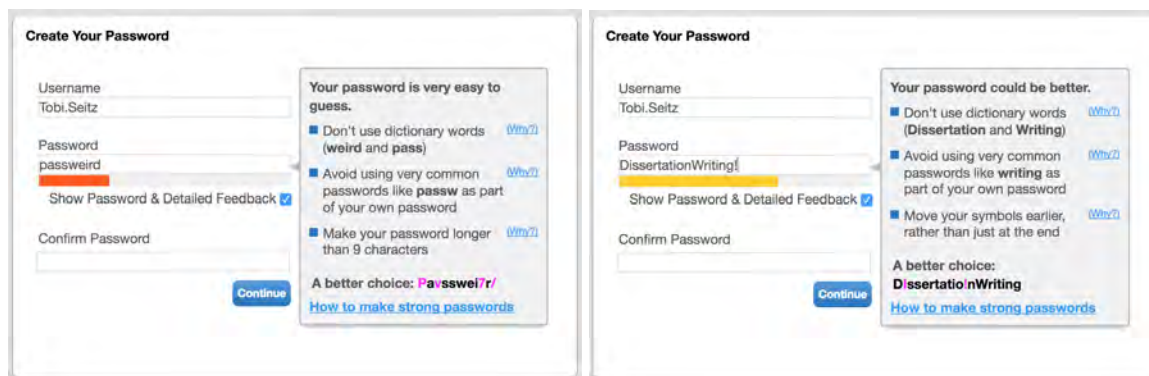
**Feedback and self monitoring:** Users are enabled to make an informed decision regarding the security of the password. **Kairos:** Meters are presented at the opportune moment of protecting an account. **Goal-gradient effect:** The closer users get to a password that is deemed strong the more motivated they become to achieve the strong rating. **Simplification:** A visual strength bar is universally understandable. **Personalization:** The dynamic visualization is user-dependent. **Suggestion:** Some meters suggest alterations if they detect weak passwords. **Reinforcement:** positive feedback about the user's password can reinforce secure behavior. **Authority:** If the service provider is a trustworthy entity, their feedback is more likely to influence the selection. **Social interaction:** Verbal feedback can speak the user's own language and open a dialog. **Loss aversion and scarcity:** The strength estimation is only shown during password entry – an opportunity to learn about one's own abilities which should not be missed. **Availability and salience:** The strength estimation makes weaknesses more salient and threats more available. **Framing:** verbal feedback and color coding allow framing the strength in a subtle manner.

Balancing all those strategies in a particular design has received much attention in HCI and security research. It is hard to trace back their origins, but proactive password checks have been part of policies since the late 1980s [369]. Bishop and Klein developed the pwcheck command-line tool that was used to give terminal-users feedback on their password selection [25]. A core challenge is the strength estimation proxy. Websites cannot implement a full-fledged cracking infrastructure just for the sake of password feedback. Thus, proxies and estimators are the go-to method.

### Effective or not?

A number of studies have been conducted on the effectiveness of password meters. Ur et al. explored 14 different designs, respectively settings, of password meters [343]: different variations of the “bar” visualization, different stringency parameters, suggestions, and nudges. They were able to identify significant differences in passwords strength depending on the meter. Very stringent meters led to stronger passwords. Any kind of visual meter resulted in longer passwords, so text-only feedback is less persuasive. Many participants in their mTurk study added another character at the end to receive a higher rating from the meter. Qualitative feedback and usability ratings showed that the stringent meters were unpleasant for many participants, reducing their persuasive power. Egelman et al. investigated context factors in the effectiveness of meters [99]. They found that password meters seemed to have a greater positive effect on strength, if they were displayed on a site with higher perceived value. Participants in that study did not “need” the password meter, but receiving feedback on highly values websites made their importance more available. At the same time they concluded, that the design of the meter plays an insignificant role and that social nudges do not affect strength either. Moreover, both Ur et al.'s and Egelman et al.'s study showed that memorability was not affected by the meters, which is a positive finding. The meter with the highest reported effects was recently presented by Ur et al. [341]. After carefully evaluating the design space for rhetorical framing of strength feedback [90], they isolated the components of password meters and compared their effectiveness. They found that textual





**Figure 3.7:** Ur et al.’s data driven password meter. It combines textual and visual strength feedback, explanations, and personalized suggestions.

feedback is recommendable to create more persuasive meters. The underlying policy also has an effect on the persuasiveness of the meter. Their final design (see Figure 3.7) uses a number of persuasive elements: A visual bar to simplify communicating password strength; text feedback about the strength; explanations about the scoring with clear calls to action (tunneling strategy); and lastly, a personalized suggestion of an alternative password.

## Beyond colorful bars

Visualizing strength is not the only way to design a password meter. Komanduri et al. crafted a sly prototype to tell users that their passwords are predictable: their *Telepathwords* system predicts the next character the user is about to type [199]. The rationale of the system is that users do not mindlessly enter a weak password anymore, i.e. that the feedback should activate System 2 thinking processes. Furthermore, realizing that the next character can be automatically guessed might evoke fear appeal and steer people away from predictable passwords. Komanduri et al. evaluated *Telepathwords* with an mTurk-study using a role-playing scenario. They found that password created in the *Telepathwords* groups outperformed traditional password policies in regard to strength and memorability. However, participants were significantly more annoyed by *Telepathwords*, perhaps because of the inconvenient truth and fear appeal. Communicating password strength with background information is another technique to persuade users. For example, Yee et al. displayed the estimated *time-to-crack* when users selected a master password for the PassPet password manager [384]. For users, it is a better call-to-action than a numeric score like “3/5”. Vance et al. found that such fear-appeal strategies persuade users to read password advice and modify weak passwords [347]. Khern-am-nuai et al. also measured the persuasiveness of context-based warning messages as part of the password meter [194]. They found that participants in their mTurk-study made significantly more changes to their initial choice when a warning was present, e.g. “Weak. We estimate that the password you chose is among the 30,000 weakest passwords”. However, the study suffered from a few important limitations (e.g. removing all data from users who were unable to log-in after 30 to 60 minutes). Kroeze and Olivier proposed evolving a Pokemon figure as users type to visualize the growth in strength [204]. Furnell and Esmael

evaluated feedback through emojis and found positive effects on the length of user-selected passwords [129]. Afjan et al. allowed users to interactively explore the visualization they had received from the password meter [11]. Ur et al. report that an animated dancing bunny that speeds up with increasing strength was unable to nudge users more effectively than less exciting meters [343]. Shay et al. tried to help users select a stronger password through a wizard that explained mnemonic phrase-based passwords [300]. Besides, they compared this strategy to the insertion approach we saw in Forget et al.’s persuasive text passwords (PTP) [126]. Both approaches were generally disliked by the participants.

Apart from pure strength visualization and persuasive messages, real-time feedback can accompany a password policy. The user sees a list of requirements and as they enter their password, they get feedback on the aspects that have already been fulfilled, which was originally proposed by Proctor et al. [258]. Although we know by now that the resulting passwords are not necessarily strong, Shay et al. found that this kind of checklist can reduce user frustration with policies in general [300]. Feedback is crucial if the policy utilizes blacklists to ban too easily guessed passwords [299]. Habib et al. evaluated Ur et al.’s data driven password meter for situations where blacklists are present [146]. They found that text feedback mitigates insecure selection especially for those users who intended to use a blacklisted password.

## Light and Shadow

Evidence about the effectiveness of password meters is mixed, but Ur et al.’s strategy might be one of the most persuasive ones, because it combines many techniques that have proven successful in different areas of persuasion [341]. So, in certain contexts, password meters can definitely nudge users towards more secure behavior.

However, where there is light, there is also shadow. Meters influence users’ mental models of password strength in similar ways as policies. However, as with policies, it is nearly impossible to reach consensus on all parameters of pro-active measures and metrics for all contexts. De Carné de Carnavalet and Mannan showed that the estimation algorithms in real-world password meters largely differ, which is a result of the natural constraints of pro-active checks [73]. Such inconsistencies, which were also observed by Ur et al. [343], have big potential to confuse users: if their password is rated “strong” on one site, and “weak” on another, a lot of explanation is going to be necessary to let the users find out why the ratings differ. Persuasive interventions probably fail at that moment. The authors suggest zxcvbn as the most robust password meter for websites and the KeePass meter as an alternative for offline tools. Others have addressed the shortcomings of industry password meters. Wang et al. developed *fuzzyPSM* based on an optimized version of PCFG [360]. Melicher et al. implemented their neural network strength estimation in password meters, too [233]. Tupsamudre et al. demonstrated that a sudden surge of n-gram scores can be used to proactively detect modifications of common passwords [337]. Improvements will continue to surface, but in recent years we can at least observe a trend towards more homogeneous strength estimation due to the gained knowledge about guessing attacks.

---

### 3.4.4 Summary

Persuasive interventions help to shape user behavior and facilitate decision-making processes. Many influence strategies have been empirically shown to persuade people across a variety of domains [150]. Thus, I believe in its feasibility for the design of secure and systems. Naturally, not all interventions work in the same ways for all users. Still, persuasive technology asserts the claim to be in the users “best interest” and wants to enable them to make “better” decisions. Thus, before any intervention, we have to be sure to do the right thing, if changing user behavior and attitudes is the ultimate goal. I follow Acquisiti et al.’s definition, and find that good decisions “minimize adverse outcomes or are less likely to be regretted” [1]. With this goal in mind, we can set out to better understand the constraints and create novel persuasive strategies which are the focus of this thesis.



# 4

## Related Work Summary

### Main Take-Aways

The current landscape of password research deals with both technical (Chapter 2) and human factors (Chapter 3). From a system-design perspective, we can take away the following state of the world:

1. There have been numerous attempts to replace passwords, but none of them are a panacea, because they entail a number of disadvantages that thwart their adoption. Any authentication scheme needs to withstand a plethora of attacks, but passwords seem to offer the best trade-off regarding usability, deployability and security compared to other schemes. Single-Sign On, multi-factor, as well as biometric authentication have gained importance and are likely to be further adopted in the future. Especially implicit biometric approaches can complement password-based authentication.
2. Entropy as a password strength proxy only works for system generated passwords. Thus, the results of a number of studies would need to be reassessed or replicated through new studies. As of now, the most reliable metric is performing actual guessing attacks on passwords. Configuring attacks is not trivial but crucial for the results. Currently, the `crackmapexec` is seen as the most robust tool. If cracking is impossible to determine password strength (e.g. for proactive password checks), the best proxy is to estimate a guess number for skillful, informed, and resourceful attackers. Neural networks and `zxcvbn` are among the most useful strength proxies at this point, but it is worthwhile to triangulate to obtain the full picture.
3. Password strength is difficult to agree upon and highly context dependent. This makes it hard to give homogeneous, consistent advice to users.

The technical aspects are contrasted and complemented by these human factors:

- 
1. Users need to manage many password-protected accounts, which is one of the drivers to come up with individual coping strategies. Although these strategies are often intuitively developed, they show commonalities among larger user groups. Passwords go through a life-cycle no matter what strategy is used. A large part of the users' password practices can be classified as risky, but in many ways they remain rational.
  2. Users tend to select predictable passwords, reuse them often, and write them down in physical or digital notes. While there are many hypotheses to explain user behavior, the problem is probably too nuanced to reach a final consensus regarding its origins. It appears to be an assessment we need to make on a per-user level.
  3. Many solutions have been proposed to guide and support users. Policies enforce requirements on users to mitigate strength issues. However, the design space of password policies is large and the best parameters are context dependent. Policies do not guarantee "better" passwords, but real time feedback can make the restrictions more bearable. Password managers can take some responsibility from users which had been shifted to them in previous years. Explicit user education was once seen as primary tool to combat risky behavior, but users continued to dismiss it, which has led to the understanding that education alone does not solve all problems.
  4. Persuasive design has been a well-studied area in password authentication to aid user education. It leverages cognitive biases and heuristics to achieve more secure user behavior. Password meters encompass many persuasive strategies at once and offer an interesting design space to influence password selection. However, their real-world implementations have brought inconsistencies to light, which reduced their persuasive power.
  5. Studying passwords has a multitude of facets and the method toolkit is large. Most studies rely on online-surveys (especially through mTurk), usability tests in the lab, or qualitative interviews. Several principles for the design of valid studies in usable security and privacy have emerged in the past two decades.

## Open Questions

Based on the review of the related literature we can identify a number of questions that have not been answered to their full extent.

1. Since results regarding user behavior and knowledge about password strength are mixed, we still have not uncovered all context-factors and exploratory variables for password selection. However, we need to have this kind of understanding if we aim to design better support systems to either reinforce or break certain habits. Since user education has been going on since decades, we ask: What have users learned about password security in the meantime? What are their mental models of password

strength and how do they change over time? Can we trace back certain behavior to failed mitigations, or are psychological factors more decisive?

2. The design space for persuasive interventions has not been fully exhausted. The effectiveness of persuasion is still somewhat low. What are novel, radical approaches to steer users towards stronger passwords? How do we find them? Which cognitive biases might be most suitable for the design of persuasive password interventions? Can we find new horizons to empower users to create stronger passwords that are still memorable? What would a holistic solution look like?
3. Password reuse is rampant and still a great risk for users, primarily in the form of identity theft following social engineering attacks. However, this aspect has not been addressed by the design of password policies. If policies cannot always effectively combat weak passwords, do they prevent password reuse? Can persuasive design aid the secure reuse of passwords?

I try to answer these questions with empirical data and reasoning in the remainder of this thesis.

---

# II

## EXPLORING THE CONTEXT FACTORS



# 5

## Mental Models of Password Strength

We start our discussion of empirical research with an investigation of psychological context factors to explain the prevalence of weak passwords. The number of extremely incautious passwords in breached databases is alarming [366]. It could be interpreted such that users might not realize that their passwords are insufficient to protect accounts. I challenge this notion in this chapter. If users are able to judge the strength of a given password, it is fair to assume they can select an appropriately strong one depending on the situation. There is first empirical evidence that users can assess passwords more accurately than one might think: Ur et al. showed that, with a few exceptions, users are able to gauge the strength of a given password [342]. In this chapter, we follow up this result by extending the range of password topologies, i.e. the composition and types of passwords. Understanding the users' mental models of password strength allows us to craft feedback more persuasively, because it can take into account what users already know about the strength of their own passwords. In particular, we address the following research questions:

RQ1: How well can users identify weak and strong passwords?

RQ2: What types of password topologies lead to inaccurate perceptions?

Beside the results, this chapter presents a novel method to answer our research questions. In the following, we discuss the background of the research and provide new insights into mental models of password strength.



---

## Publication Statement

Some insights of this project have been previously published at OzCHI 2017, with Heinrich Hußmann as co-author [289]. The paper builds the foundation of this chapter. However, the chapter extends the publication in three important aspects:

- We show another design iteration for the password game that introduced a new study condition and improvements of the password generation algorithm.
- The dataset reported in the extended analysis is 85% larger than the one in the OzCHI publication.
- We perform new analyses, provide new insights, and dive deeper into the discussions of the findings.

## 5.1 Background and Context

Users create passwords on a regular basis. As discussed in Chapter 3, previous research indicates that users most commonly resort to weak passwords that are easy to remember. In case the account value is higher, however, it is suspected that users invest more effort into creating a stronger password. Password meters are helpful in this context [99]. Yet, even strong passwords are often ineffective against sophisticated guessing attacks, so it stands to reason that users have a subpar mental model of what makes a strong password. This has been a commonly accepted assumption [35] but it oversimplifies the state of affairs.

Mental models change over time – both on a micro- and a macro-scale. At micro-scale, it is evident that people select their first passwords in their teenage years and without much care for security [354]. Over time, users are exposed to password advice and educational nudges: popular news portals regularly publish new articles that report on data breaches or warn about risky online behavior. Furthermore, incidents in one’s own social network [316] raise awareness about the topic and can spark plans to behave differently in the future. Moreover, by creating many accounts, users encounter different password policies and feedback tools. Users need to make sense of these constraints and might reflect on their password choice. Consequently, an individual’s mental model of password strength is impacted if feedback and policy instructions are well-designed [300, 341]. The literature already has some evidence that, overall, password behavior has positively changed in the past few decades [269, 282].

Zooming out to the macro level, efforts to educate users may have already paid off. In the early days of research in Usable Security, often the user was seen as the “weakest link” in a secure system [4, 277]. When Florêncio and Herley conducted their large-scale study in 2006/2007, they noticed that the entropy of the passwords was generally low, but high-value accounts were protected a bit better. In 2015, Ur et al. found that users’ mental models had

become more accurate which allowed them to create stronger and memorable passwords [345]. A year later, they presented an in-depth analysis of the perceptions of password strength [342]. They found that for the most part, participants in their online-study were capable of identifying the factors that add to password strength. Shay et al. already used “perceived strength” as a proxy metric [300], and it seems this is a better approach than one might think. For a few notable exceptions, though, certain password characteristics fooled participants. Ur et al. argue that errors in mental models about strength arise from false understandings of attackers. Consequently, it would be necessary to shift the focus in user education from password strength towards attack models, because users already have a fairly accurate understanding of strength.

Florêncio et al. put forward a formal model as to why users behave insecurely. Their central argument is that it is inevitable to choose weak passwords for some accounts and that users are well aware of their behavior. Yet, if users have a faulty understanding of password strength then the selection process is biased. If users also show the overconfidence bias, where people are subjectively more confident in their abilities and judgments than the objective accuracy of the judgments [307], this could indeed lead to considerable risks. Password meters could counteract these biases, but only if they achieve to inform – and potentially disrupt – the user’s mental model of password strength.

In summary, having a clear mental model of what makes for a strong password is essential to make the decision whether to expend the effort and create one or not. While there are initial results, the perceptions of these factors are still understudied to this point.

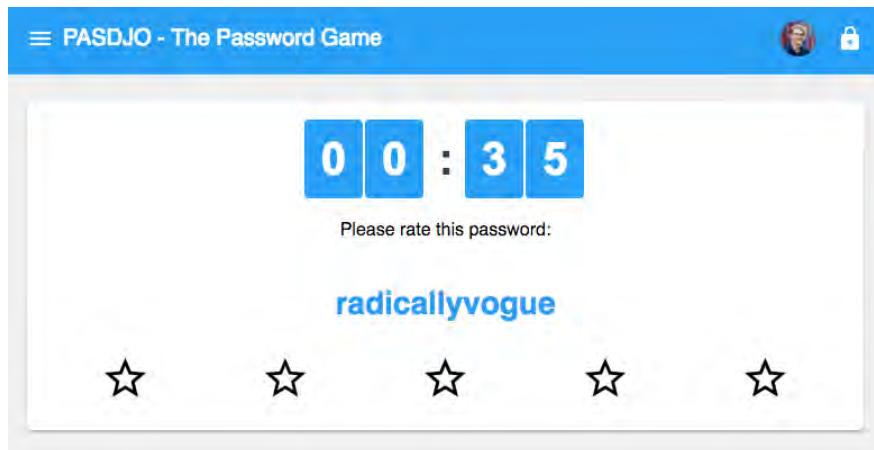
## 5.2 Approach: PASDJO - The Password Game

To evaluate (in)accuracies in mental models about password strength, I chose to have users rate passwords similarly to the study task in Ur et al.’s study [342]. In their presentation at CHI 2016, they challenged the audience to assess the strength of a couple of passwords on the slides. This sparked the idea of making a game out of this study topic. The game is called PASDJO (pronounced “Pass Joe”<sup>1</sup>). To open the game to a large audience, I decided to implement it as a web-application that runs in any web browser on various platform.

### 5.2.1 Game Mechanics and Design Elements

The game is relatively simple: Players judge how strong or weak a given password is. They receive points by accurately estimating the strength of a given password on a scale from 1 (weak) to 5 (strong). The game follows similar design strategies for the password topologies as Ur et al.’s online study, but passwords are either randomly taken from large dictionaries

<sup>1</sup> The name PASDJO was inspired by the Bavarian “passt scho”, which translates as “it’s alright”. Players need to provide an assessment if the given password *is alright*.



**Figure 5.1:** Screenshot of PASDJO during gameplay. The user clicks the stars to rate the password. One star = weak, five stars = strong.

or generated on the fly. To induce intuitive estimations, a time-limit is enforced while a “highscore” acts as incentive to estimate as accurately as possible. To reach higher scores, one has to judge as many passwords as possible in 60 seconds.

## Scoring and Metrics

A crucial point of the game is how the passwords are rated objectively. Here, we rely on the `zxcvbn` library<sup>2</sup> (described in detail in Section 2.3) because it is highly reliable and straightforward to use on a web page. It also comes with a number of word lists that are helpful to implement different scenarios, respectively conditions. Beside the guess-number metric, `zxcvbn` also scores passwords on a scale from zero (weak) to four (strong). We translate this scale to one star (weak) to five stars (strong) in the game.

For a correct estimation, i.e. a match between the `zxcvbn` score and the player’s rating, a player is awarded 100 points. The difference between the user’s rating and the `zxcvbn` score is the *deviation* ( $D$ ). A player’s rating can deviate by at most four stars, e.g. if they rate a five-star password with a score of one and vice versa. In that case, the player should not get any points, but in all other cases, the player is still awarded a few points. For each integral deviation in either direction there is a penalty of 25 minus-points (maximum of 100 points, at most 4 errors, which leads to  $100/4 = 25$ ). The points of all estimations are summed up and build the *achieved* score ( $A$ ). As an overall accuracy measure, at the end of the game we calculate the ratio of achieved and possible points and display it as percentage ( $P$ ). The game thus implements the following scoring function, where  $U$  is the user’s estimation on a scale from 1 to 5,  $Z$  is the `zxcvbn` score on the same scale, and  $n$  is the number of passwords a user has rated within the time limit.

<sup>2</sup> <https://github.com/dropbox/zxcvbn>

The scoring function takes an array of user ratings and zxcvbn scores for  $n$  passwords and returns a vector consisting of the achieved points  $A$  and accuracy metric  $P$ .

$$f([(U|Z)_1, \dots, (U|Z)_n]) = (A|P)$$

For each estimation the deviation from the zxcvbn score is calculated.

$$D_k = U_k - Z_k$$

The total achieved points are the sum of the achieved points per round. Each round takes the 25 point penalty into account. If  $|D_k| = 0$ , the player gets the full 100 points per round.

$$A = \sum_k^n 100 - (|D_k| * 25)$$

Finally, the accuracy is the fraction of achieved and possible points.

$$P = \frac{A}{n * 100}$$

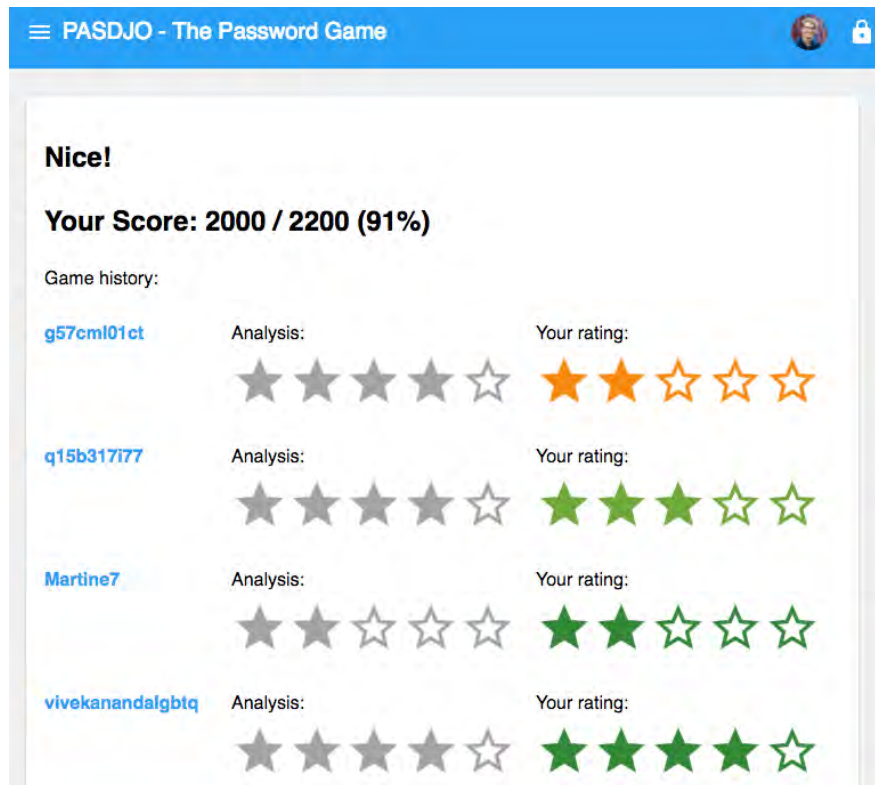
The score and accuracy are displayed to the users once the game is finished, see Figure 5.2.

## Persuasive Design Elements

We used a number of persuasive techniques (see Section 3.4) in the design of the game. Most of them relate to the Persuasive Authentication Framework [124]. First of all, we lower the barrier to play the game through *simplification*: the task is immediately clear and does not require special skills. Since one round only takes sixty seconds, it is easier for users to *commit* to finish the game. The feedback screen uses positive *language* to create the notion of a *social interaction*, which also serves *self-monitoring*. Also, feedback and points serve as *reinforcement* for correct ratings, i.e. a subtle form of *conditioning*. Moreover, the user interface is *tailored* to the user, as they can see their own previous scores. Players can also create an account through Google-federated SSO. If they log in, their profile picture and a personal greeting *personalize* their experience. Through the player's eyes, the game's strength ratings are "right", which can be attributed to the *authority* principle. Finally, the game leverages the *goal-gradient effect* by giving the player small amounts of points even for inaccurate estimations. At the end, the player receives a score that seems improvable through playing another round.

### 5.2.2 Password Generation and Study Conditions

PASDJO initially had four different password types that act as levels of the independent variable: Common passwords, mangled passwords, passphrases, and random passwords. During gameplay, the condition for the next password is selected at random. The following paragraphs depict the conditions in detail.



**Figure 5.2:** Feedback screen. The player can review their performance after the time is up. Each assessment is contrasted with zxcvbn’s score.

**Common passwords:** We take the word lists that come with the zxcvbn library. One of these lists contains 47023 leaked passwords ordered by frequency, from which we randomly pick one for this condition. The data stems from breaches of user databases at RockYou, Yahoo and Xato [369]. All passwords are lower-case and can be considered weak, because they are usually amongst the first attempts in a guessing attack [346] unless the adversary launches a targeted attack where personal information plays a more important role. Zxcvbn rates the top 1000 passwords with a score of one (e.g. “12345”, “password”, “monkey”), and the remaining passwords are scored with two stars (e.g. “iloveyou2”, “skywalker”, “apollo13”). It is worth noting that many of these passwords would not be accepted anymore by websites as common policies demand at least 8 characters (see Chapter 6), which RockYou and Xato did not enforce at the time.

**Mangled passwords:** For the mangled password condition, we take the same list of the top 47023 passwords, but we algorithmically substitute certain characters. The substitutions look like “leet” / “l33t” speak, which is a typical way to try to increase password strength [67, 226]. For instance, an “a” is replaced by an “@”, or an “s” is substituted with the dollar sign “\$”. Also, random characters are transformed to uppercase. To allow recognizing the original word, we only mangle up to 30% of the characters of the password. Since we only use substitutions that zxcvbn recognizes, mangled passwords mostly receive

a score of two. However, in rare occasions, they get higher or lower scores depending on the specific character constellation, e.g., “p@ssw0rd” (1), “b0n3he@d” (2), “fireFI9hter” (3), “123qaz456w\$x” (4), “123Q@z4s6w\$x” (5).

**Random passwords:** We implemented a simple string generation algorithm to create random passwords. They are lowercase alphanumeric passwords containing letters from the German alphabet, i.e. [a-z0-9äöü]. Zxcvbn consistently gives them a score of four, which makes them easy to rate. In a real-world attack they can only be brute-forced [115, 369].

**Passphrases:** We combine two entries from the English Wikipedia index to create a passphrase (also shipped with zxcvbn). The words were required to be between 4 and 11 characters long. This restriction leads to a dictionary size of 27202 entries. Thus, there are  $27202^2 \approx 10^9$  possible combinations, which is unpredictable enough to withstand online guessing attacks. This is reflected by the rather high scores: zxcvbn gives passphrases mostly a score of three or four, e.g. “armedtamils” (3), “boostedeuros” (4). If the words appear in other dictionaries, e.g. the “TV subtitles” word list, they are more likely to result in a score of three. However, for a user it is not straight-forward to tell whether a passphrase scores three or four points, which makes this condition harder to get right. The game is expected to benefit from this element of *unpredictability*, which is one of the eight cornerstones of gamification according to Chou [55].

For the remainder of the chapter we refer to these four condition as “Common”, “Mangled”, “Random”, and “Passphrase” (mind the capitalization later).

### 5.2.3 Benefits and Shortcomings of the Game-based Approach

Since the game was deployed publicly, the method can be denoted as an unsupervised in-the-wild study. Ur et al.’s study is closely linked to ours, but they used an online survey and recruited participants through mTurk. In comparison to an online survey, our approach features certain benefits and shortcomings as discussed in the following.

#### Benefits of the Game-based Approach

**Easy collection of multiple data points per user** The game can be played over and over. A survey could also be taken multiple times, but this is usually impractical because there is a fixed set of options, and reassessing the same passwords is unnecessary. Besides, if people could repeatedly participate via MTurk this would mean that they would be paid for each completion, which drives costs.

**Possibility to provide feedback** After the game, the players can review their ratings and find out how they performed. Implementing such a feedback loop is much more complicated and often impossible with current survey tools. Survey usually debrief participants about the



---

general purpose of study, but it is difficult to tailor the debrief to the individual participant, i.e. tell them how they performed.

**Randomization and password space** Ur et al.’s study was well-designed because it tested a wide range of password characteristics and there were multiple options per condition. However, the options were predefined by the researchers and limited in that sense. In our case, we can pick passwords from much larger word-lists at random and use algorithms to randomly adjust certain characteristics. This allows for higher internal validity of the data collection.

**Intrinsic Motivation** Participants in an online survey often receive an incentive to participate. In an mTurk-study each participant receives a small amount of money. Contrarily, there is no extrinsic incentive to play PASDJO and we can leverage the players’ intrinsic motivation.

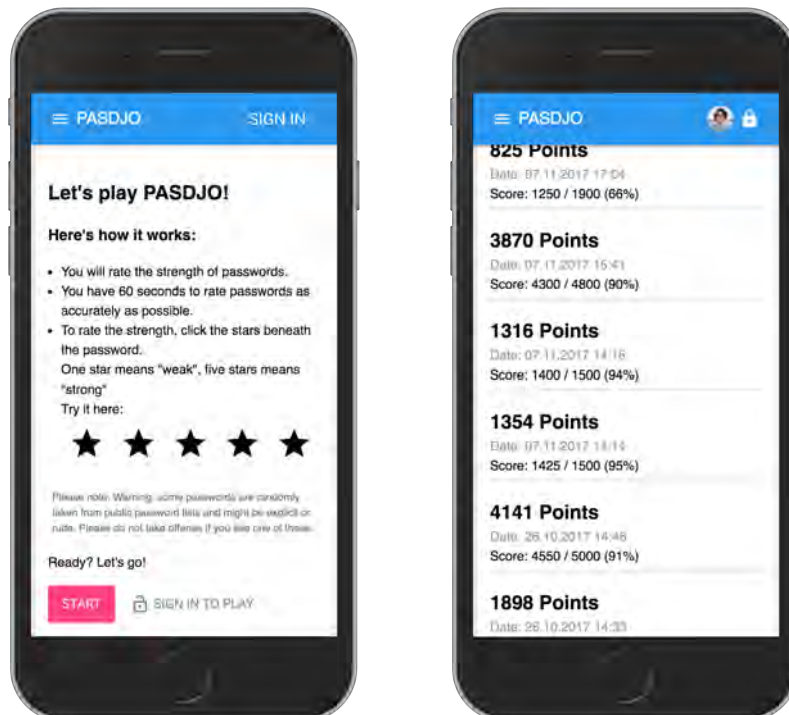
**Intuition** The game uses time pressure to induce intuitive responses. Thus, if playing fast enough, we expect to measure System 1’s responses instead of System 2 [183]. This is vital because password selection is often a System 1 process, i.e. intuitive and automatic. In an online survey, the participants can take as much time as they’d like to reason about their response. However, this is a hypothetical benefit that needs further empirical evidence.

## **Drawbacks of the Game-based Approach**

The game approach towards collecting password strength perceptions naturally entails drawbacks, too. First and foremost, we do not collect demographic data. In a survey, it is common practice to include questions about basic demographic information. If we included such questions as part of our game, we might have scared off users and damaged persuasive strategies. Thus, the collected data stems from an unknown population, which is a limitation. In the future, we might address this issue by iterating on designs that ask people to provide some basic demographic information, e.g. after playing three times.

Furthermore, the lack of demographic data also makes it hard to track usage across multiple devices. The same player could first get in touch on a desktop and then later decide to keep playing on their mobile. Allowing users to log in and synchronize their game history can mitigate this problem to some degree.

Last, ethically running password studies (see Section 3.1) demands that participants be debriefed appropriately. In our case, we provide a separate “About” page that is accessible from anywhere in the app. The information on the about page points out that users should refrain from using the generated passwords as their own and that our strength estimation is only an estimate. Nevertheless, it requires that users seek this kind of information, which – realistically – only few people might do. Therefore, as a small encouragement to read it, we require users to confirm a consent-dialog that also points them to the about page. So far, we have no data on how many people looked at this page, but this is an easy fix for the future, e.g. by adding an analytics library.



(a) Welcome screen for anonymous users. The game is briefly explained and the users can try the rating interface (stars).

(b) Users can browse through past games. The updated version visualizes the scores in a line chart (not displayed).

**Figure 5.3:** Key screens in the user interface.

## 5.2.4 Implementation

For the implementation, we used modern web technologies to create a state-of-the-art user experience. The front-end uses the polymer library<sup>3</sup>. *Custom elements*, which were recently added to the WebComponents standard<sup>4</sup>, serve to separate concerns through encapsulation. On the back-end, we rely on the Firebase platform<sup>5</sup>. It allows setting up an easy and secure authentication, respectively authorization, workflow. Moreover, the data is stored in a *real-time-database* that uses fast web-socket connections to synchronize data between client and servers.

We leveraged user experience heuristics to design the game for a large audience. Among others, we based our solution on the Progressive Web-App heuristics<sup>6</sup>, which we hoped

<sup>3</sup> <https://www.polymer-project.org/> (last accessed 22.01.2018)

<sup>4</sup> <https://www.webcomponents.org/specs> (last accessed 22.11.2018)

<sup>5</sup> <https://firebase.google.com/> (last accessed 22.01.2018)



---

would lower the barrier to engage with the game. Since we expected many people from Germany would play the game, the interface is fully localized. Depending on the browser language, it automatically picks the preferred language. Moreover, the interface is responsive to different screen sizes to provide a seamless experience. We chose a short URL [pasdjo.de](https://pasdjo.de) to facilitate visiting the website from mobile devices where typing is cumbersome. Upon visiting the web page, a welcome screen briefly explains the rules (see Figure 5.3a). As part of our onboarding process, we give users the opportunity to try out the rating interface, so they can get comfortable even before they play for the first time. For mobile users, we had to ensure that the game loads quickly. This is challenging, because we have to transfer a number of word lists to generate the passwords randomly. Therefore, we used a lazy-loading design pattern to render the UI first and load data in the background. At the same time, we minimize and compress the source files to reduce data transfers. The first page load consumes 1.6 MB of data. To ensure subsequent visits are faster and do not stress the user's mobile quota, a *service-worker* caches resources in the background. It intercepts requests to the remote server in the future and instead serves local resources<sup>7</sup>. This strategy reduces the transfer size for subsequent requests to a mere 10.3 KB, which are mostly consumed by authentication and incremental synchronization.

## 5.3 Log Analysis

We deployed the game and made it publicly accessible. We ran the first analyses after four months and a second analysis after one year of public deployment. The first sample served as the basis for the full-paper at OzCHI [289], while the second has not been previously published.

### 5.3.1 First Sample

Data collection started in December 2016 and the first data dump was created in March 2017. The game was first presented to the public at an open lab day in our research group. We set up a booth and invited visitors to play the game on their own devices, although we also had two iPads that used a shared, known account. Posters at our institution kept advertising the game, and it was later shown at student orientation days. Thus, it is very likely that the sample stems from a group of mostly younger adults who are interested in technical innovation. After removing log data from the known demo accounts, 115 users remained in the data set who had played at least one game on a device unknown to us. Only two users chose to log in to keep their history synchronized across devices.

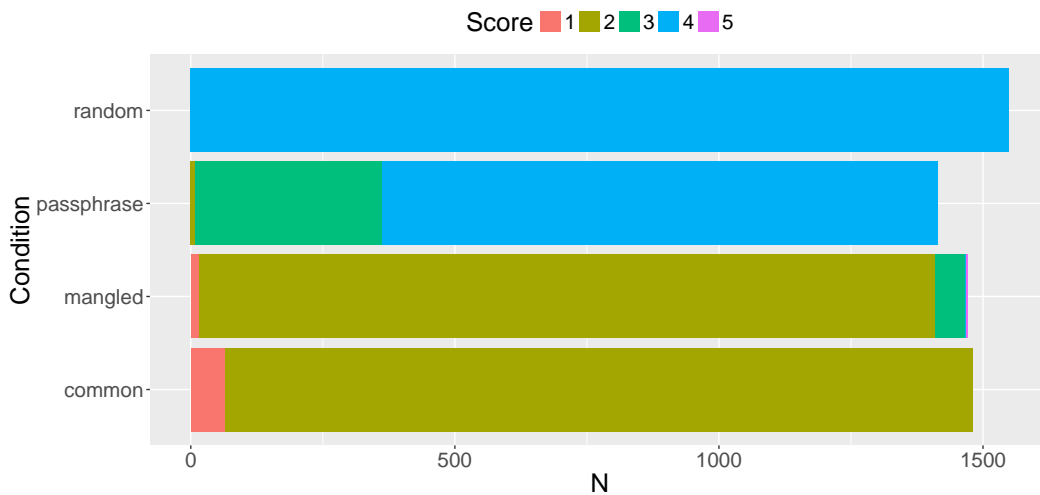
---

<sup>6</sup> <https://developers.google.com/web/progressive-web-apps/checklist> (last accessed 22.01.2018)

<sup>7</sup> This is an erratum in our original publication, where we reported using IndexedDBs

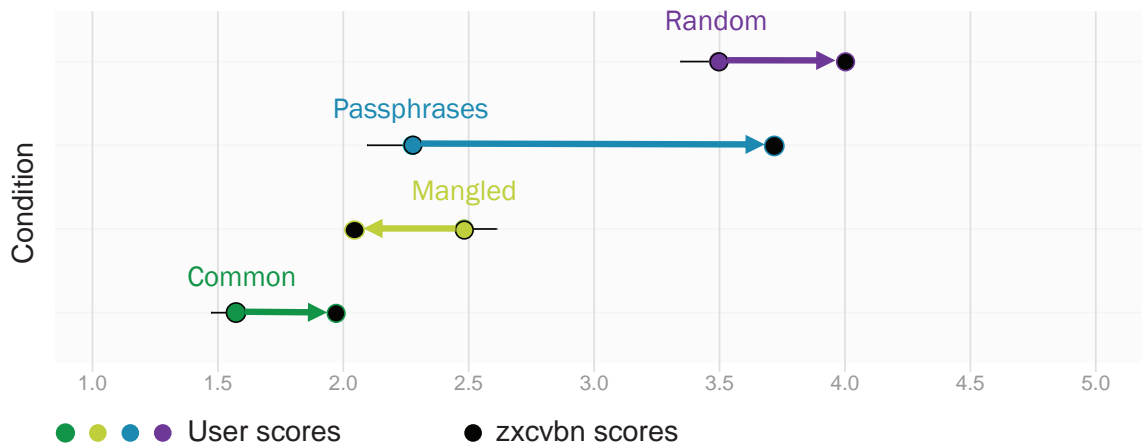
## Score Distribution

The 115 users played 242 full games in total. Thus, on average, a user played 2.1 full games (SD=2.3, Min=1, Max=16). The time pressure motivated users to rate as many passwords as possible. We found that an average of 24.44 passwords (SD=10.39) were rated per game, which amounts to 5915 passwords rated. Users spent  $\approx 2.45$  seconds per estimation on average. The zxcvbn strength estimates were fairly consistent within each condition (see Figure 5.4). Random passwords consistently received a zxcvbn score of 4 out of 5 stars. Common passwords were either rated with 1 or 2. Passwords in the Mangled condition filled the full spectrum of ratings, but the majority had a score of 2. Interestingly, passphrases were the least predictable condition. From  $n = 1414$  passphrases, 25.03% received a score of 3, and 74.54% a score of 4. The remaining eight passphrases were deemed weaker with a score of 2, e.g. *soilwithin* or *augsburgtime*, but it is not clear why this was the case (perhaps because one of the words had a very low guess number). However, the vast majority of passphrase-scores are explicable with the number of characters. For the users, the length of the passphrase is then the only criterion to decide between a score of 3 and 4.



**Figure 5.4:** Distribution of zxcvbn scores for  $N=5915$  passwords collected in the first four months of deployment (1 = weak, 5 = strong). Random passwords were the most stable condition with consistent score of 4. Mangled passwords show a range of 1 to 5, but passphrases were least predictable.

Moreover, the distribution in Figure 5.4 shows that in most cases, the maximum deviation  $D_{max}$  was less than 4:  $D_{max} = 3$  in 5420 cases, and  $D_{max} = 2$  in 410 cases. In other words, the user's estimate deviates by *at most* 3 score points in 98.56% of cases. At the same time, it is difficult to *underestimate* passwords that had already received a low score. This narrows the interval of the expectancy value for common and mangled passwords, i.e. users are more likely to *overestimate* these – if at all. These two findings motivate taking the absolute value of the deviation and create an overall accuracy score for each game played, just like we did.



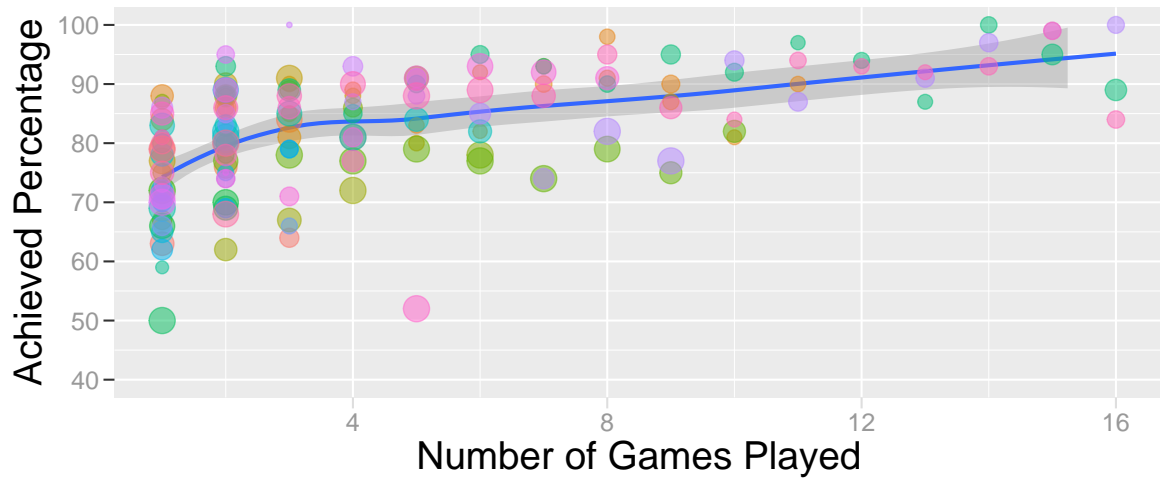
**Figure 5.5:** Average user ratings in each of the four conditions, plotted alongside mean zxcvbn scores. Users’ estimations were least accurate for passphrases, which they had rated 1.4 points lower than zxcvbn. (N=111)

### “First Game” Statistics

A player’s first game is vital for the assessment of their pre-existing strength perceptions in different conditions. After the first game, learning effects induced by our feedback are likely to change their subsequent judgments. Therefore, the first game is the most valuable indicator of mental models of password strength. In their first game, players achieved an average score of 2010 points (SD=653.16, Min=350, Max=3675), which translated to an average accuracy of 74.58 % of the achievable points. Moreover, players rated 27.21 passwords in their first game (SD=8.34), which is more than in subsequent games, albeit not significantly ( $t_{32}(2) = 1.13, p > 0.1, 95\% - CI = [-2.04; 7.12]$ ). The first games consisted of 6.9 Common (25.3%), 6.8 Mangled (25%), 7.2 Random passwords (26.5%), and 6.3 Passphrases (23.2%). The conditions were thus evenly distributed ( $F_3 = 2.36, p > 0.05$ ).

As a next step, we can evaluate the achieved accuracy for each player in each condition. Since randomization led to four users not rating at least two passwords in each condition, we remove their data and maintain a sample of N=111 users. Non-parametric tests are used to account for the coarse password ratings (1 through 5).

In their first game, players showed fairly consistent inaccuracies (see Figure 5.5). There was no notable correlation between the number of passwords rated during a game and the resulting accuracy ( $\rho = -0.1, p > 0.1$ ). In other words, fast players were about as accurate as slow players. Mangled passwords were the only condition that was overestimated (estimated median deviation  $Md = 0.5$ ). Interestingly, common passwords were underestimated slightly, although there was little room for underestimation ( $Md = -0.5$ ). Random passwords were also underestimated by about half a point ( $Md = -0.5$ ). The users in this data set rated passphrases exceptionally low ( $Md = -1.6$ ). A Friedman rank-sum test showed significant differences for the deviations across the four conditions ( $F_3 = 187.84, p < 0.001$ ). We followed up this finding with Bonferroni-corrected post-hoc tests, i.e.  $\alpha_{Bonf} = 0.008$ . Wilcoxon

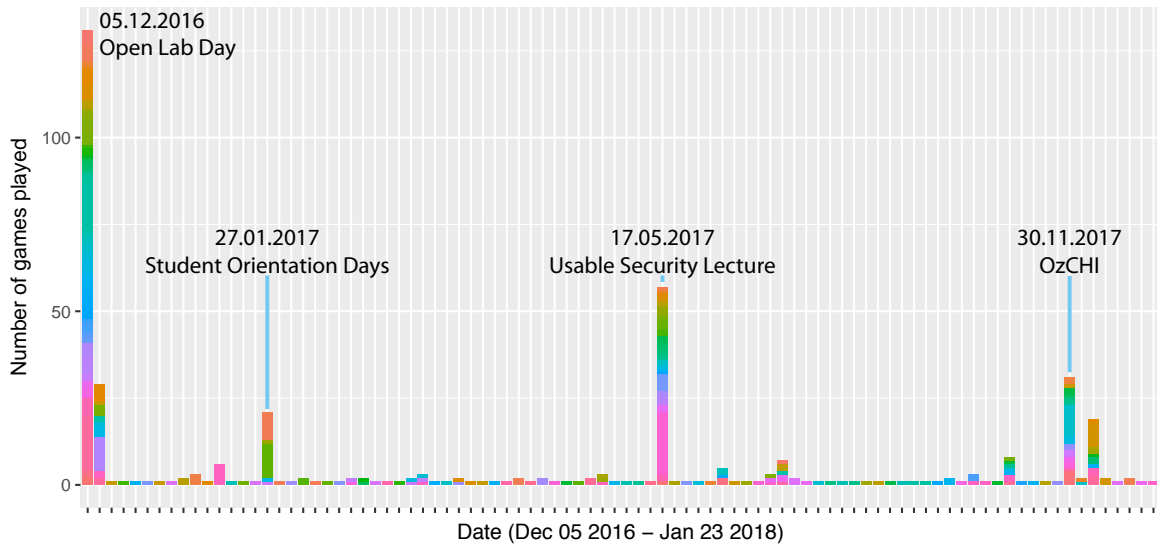


**Figure 5.6:** Achieved percentage (accuracy) evolution depending on the number of games played. Each dot represents a full game of a user, the size of the dot indicates how many passwords were rated in that particular game. Although a linear model works, we used a locally weighted polynomial regression (LOESS) to fit a line through the data points. Users clearly become better the more they play.

paired sample tests revealed significantly different deviations between all pairs of conditions, except for the common/random pair. In other words, common and random passwords were equally inaccurately assessed by the users.

### Score Development

From the 115 users who completed a full round of PASDJO, 33 kept on playing for at least one more game. We tried to model their progress through a linear regression with `gameIndex` as predictor and achieved percentage as dependent variable. To account for the fact that some users played more rounds per game, we weighted the percentage with the number of passwords rated in that game. The model shows that players were able to improve their accuracy by playing multiple times ( $F(1)=49.37$ ,  $p < 0.001$ ,  $\beta = 0.54$ ,  $R^2_{adj} = 0.23$ ). The increased accuracy is mostly due to more accurate estimation of passphrases: performing linear regressions with average deviation of each condition as dependent variable, with `gameIndex` as predictor, we observed that *only passphrases* have a non-trivial model fit ( $R^2_{adj} = 0.19$ ). For all other dependents, `gameIndex` had virtually no effect on the tendency. Players therefore primarily learned to estimate passphrases. Figure 5.6 visualizes the progress of users who played at least twice. In there, it is also evident that the slope is not stringent, i.e. an individual user's accuracy does not necessarily increase steadily. We attribute this to the element of randomness in the game, which makes it more difficult in some cases to rate the passwords, while it might be easier in a subsequent game.



**Figure 5.7:** The game was played on 90 distinct days (days with 0 games are not shown in the graph). There were four major drivers for the adoption (see annotations), that indicate that the sample mostly consists of students, their peers, and other academics.

### 5.3.2 Extended Sample and Results

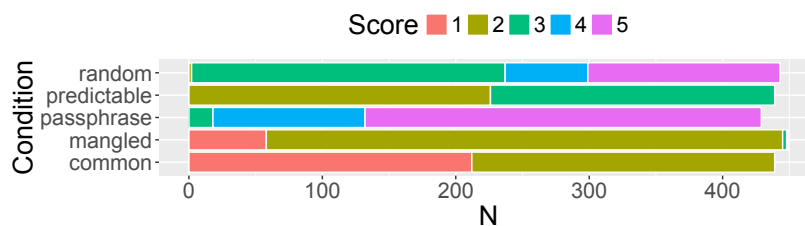
On the 23rd of January 2018, I created a new data dump and re-ran the analyses to see whether there were any differences to the first publication. The data set at this point consisted of 10,965 played rounds from 342 games played by 190 users on 90 distinct days (see Figure 5.7). Thus, there was an increase in 41.32% in terms of games, and 85.38% of played rounds. Before we look at the logs, I briefly present the changes introduced in PASDJO version 2, which was deployed on November 7th 2017 (i.e. prior to the OzCHI conference, where the game was demoed).

#### Version 2

After publishing the first results, I had iterated the design and implementation of the game. There were a number of changes that might affect the players' performance. Splitting the data between version 1 and version 2 allows us to compare the players' performance before and after the adjustments to measure their effectiveness.

**New Condition: Predictable** The mangled passwords were algorithmically altered versions of common passwords. This often made it hard to figure out the original password. Perhaps, the mangled passwords sometimes seemed more like random passwords which is why they were the only overestimated condition in the first sample. The new condition “Predictable” mimicks the most predictable user alterations. The resulting passwords consist of a capitalized dictionary word with a predictable suffix, e.g. ‘123’ or ‘!’. There were 12 predefined suffixes, and they were randomly appended to the dictionary word.

**Tuning Password Generation** After realizing that virtually no passwords had received a score of 5 in the first sample, I fine-tuned the generation of passphrases and random passwords. Passphrases were now either two or three words. Separators (e.g. a dash “-”) were inserted in between the words. Random passwords used to be ten characters long in version 1. In version two their length is also randomized between 7 and 15 characters. This leads to a higher variety of zxcvbn scores. The resulting passwords sometimes resemble the mangled condition, which is a greater challenge for the players. Moreover, I increased the likelihood of common passwords from the top 100 by a factor of two. All measures resulted in a much more diverse spectrum of scores as shown in Figure 5.8.



**Figure 5.8:** Zxcvbn scores in version 2. The scores within the conditions had become much more diverse, which should make it more difficult for players to judge individual passwords.

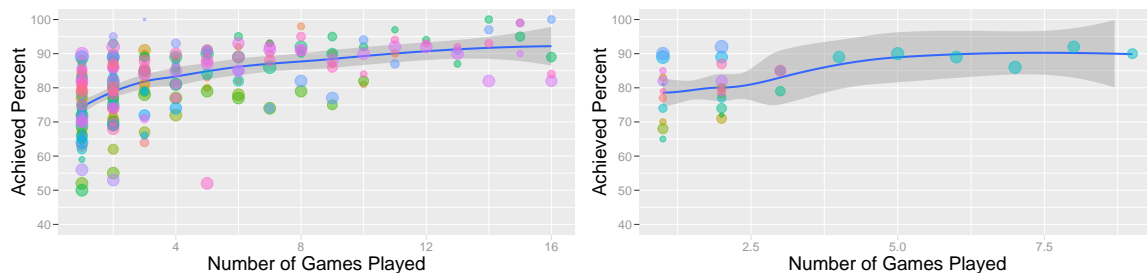
**Scoring and Feedback** The players’ achieved score (A) did not entirely reflect their performance. For instance, if a player rated as many passwords as possible without spending much thought on the strength, they still achieved a high score, albeit not a high accuracy. Therefore, the accuracy is now used as a “penalty” by multiplying the achieved score with the accuracy (e.g.  $3000A * 0.5P = 1500$  final score), which represents the performance much more adequately. Furthermore, the players had to figure out why their ratings deviated from the zxcvbn scores on their own. Now, the feedback screen contains a small indication as to how the score was calculated (see Figure 5.9). This enables players to improve their score in the next game.



**Figure 5.9:** Updated feedback in version 2 informs the player about the password category and provides pointers to the origin of the objective rating.

**Leaderboard and History** Finally, we introduced a new game-design element, namely leaderboards. Players can pick an alias and submit their score after each game. The main purpose of the leaderboard is to encourage *competition*, which is another persuasive design strategy.

## Score Development



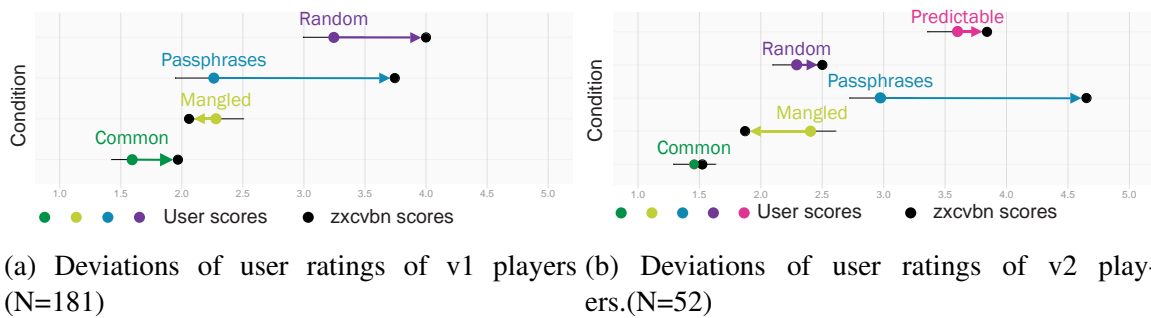
(a) Performance of v1 players (Dec 2016 - Nov 2017) (b) Performance of v2 players (Nov 2017 - Jan 2018)

**Figure 5.10:** Performance development during the first 13 months of public deployment (5.10a). In November 2017 version 2 was released, which roughly shows a similar learning curve.

Among the players who played at least two games, the overall score development has not changed visibly (see Figure 5.10a), with no notable differences in the regression model ( $F_1 = 67.12$ ,  $p < 0.001$   $\beta = 0.53$ ,  $R^2_{adj} = 0.23$ ). However, if we look at the players who played version 2, we can observe a slight decrease in the model fit ( $F_1 = 8.83$ ,  $p < 0.01$   $\beta = 0.43$ ,  $R^2_{adj} = 0.20$ ): There were 12 players who played version 2 more than once ( $M = 2.9$ ,  $SD = 2.29$ ). Their progress is shown in Figure 5.10b. The decrease can be explained by the fact that only one player played more than three times in this reduced data set.

## Strength Perception in the First Game

We also re-calculated the users' average accuracy in each condition, again only using the players that had generated data in all conditions in their first game (see Figure 5.11). As before, the deviations from the zxcvbn scores differed significantly across conditions in version 1 ( $F_3 = 316.4$ ,  $p < 0.001$ ). The same goes for version 2 of the game ( $F_4 = 108.7$ ,  $p < 0.001$ ). All post-hoc pairwise comparisons showed significant differences  $p < 0.008$  for version 1. For version 2, the situation is not as clear, because the sample is smaller (724 observations). The significantly different pairs were Mangled-{Common|Passphrase|Predictable|Random}, and Passphrase-{Common|Predictable|Random} ( $p < 0.005$ ). The lack of significant pairwise comparisons is also visible in Figure 5.11b: The users' ratings did not deviate much from the zxcvbn score in both the Common and the Random condition, which explains this result. In other words, mangled and predictable passwords, as well as passphrases caused the players the most trouble, so their accuracy in these three conditions differed significantly.



**Figure 5.11:** Users’ average ratings, for PASDJO version 1 on the left and version 2 on the right. The version switch appears to boost accuracy for common and random passwords, but not for mangled passwords and passphrases.

## 5.4 Discussion

In the following, the results are put into context, which also allows us to derive implications for future support during password authentication.

### 5.4.1 Players’ Overall Performance

Overall, we can attest players a fairly good performance. This assessment resounds to Ur et al.’s results [342]. Players performed best in the “Common” condition, i.e. they can identify commonly used passwords with high certainty. Algorithmically mangled – but common! – passwords caused a bit more trouble. The fact that this was the only condition where people judged strength as too high reveals the users’ mental model: *if one substitutes individual letters with digits and symbols, the password must become stronger*. The additional data for version 1 helped narrow down the confidence intervals and moving closer to the “actual truth”. After about one year, the average rating for mangled passwords had moved closer to the zxcvbn score (compare Figures 5.5 and 5.11a). Perhaps this means that the players that followed ( $\Delta(n) = 70$ ) had a better mental model of the implications of character substitutions. However, since none of the other average ratings changed significantly, the shift might be due to chance.

### 5.4.2 Strength Perception vs. Selection

Users rarely rate the strength of passwords. The only notable instance where this happens is when users see passwords that are shared with them. There is growing evidence that shared passwords are typically weak [154, 298, 308, 351, 368, 389]: Sharing passwords is easier if it is a regular dictionary word, with predictable suffixes, like Ferrari123. It would be extremely cumbersome to dictate a password like 8zJ,uwD’ dXBTUVub over the phone,



---

especially if the other person needs it urgently. As a consequence, passwords shared among users might have affected players' strength ratings. We did not observe this in our data: The conditions that contained regular words were deemed weak. Ur et al.'s findings point in the same direction.

We can safely assume that users in our sample know that common passwords are weak, and that they are misguided by character substitutions. We believe that this has immediate implications on their selection behavior: it can be hypothesized that users intentionally pick weak passwords – as data breaches reveal time and time again. If they think a stronger password is necessary, they mangle weak passwords and overestimate the security benefits. Therefore, there is a lesson we can learn here: users do not need feedback on obviously weak passwords. They *do* need feedback if they select a predictably altered password. Ur et al. mention this already in their recent work on password meters [341]. However, many password meters and feedback mechanisms do not yet discourage predictable mangling.

### 5.4.3 Intuition or Deliberation?

PASDJO players spent 2.45 seconds on average to rate the strength of each password. Which kind of thinking processes were responsible here? Is it probable that users acted intuitively (System 1) or did they ponder enough to activate System 2 processes (see Section 3.4.1)? We can find evidence for both. In their first game, fast players were about as accurate as slow players. For those players who played multiple games, we used the number of passwords they managed to rate as weight in regression models. We did this for validity reasons, but, in fact, the effect was negligible: We ran the tests without the weight and model fit did not change notably. All this is interesting because fast players were most likely to use *heuristics*, or as Acquisiti et al. calls them: *mental shortcuts* to come to an assessment. Otherwise they would have been less accurate or slower. This is one piece of evidence that the time pressure can induce System 1 thinking processes. The slower players on the other hand obviously had to think harder to estimate password strength; System 2 had to be activated. In PASDJO v2, *zxcvbn* scores for passphrases and random passwords were much more diverse (see Figure 5.8). So, if a fast player wanted to be accurate in these cases, they needed to rely on two heuristics: *are there dictionary words or does it look random?* and *how many characters are in the password?* Only then does one have a chance to assess strength correctly. For the *length*-heuristic, it is even necessary to count the characters. We can safely assume that 2.45 seconds are not enough to make the distinction between a 17 character (average score = 3.9) and a 19 character passphrase (average score: 5.0). For random passwords, it could be feasible to count characters, because scores are much more predictable ( $\text{length}(7,8) \rightarrow 3$ ,  $\text{length}(9,10) \rightarrow 4$ ,  $\text{length}( > 10) \rightarrow 5$ ). Nonetheless, we maintain the assumption that rough heuristics helped more than analyzing all features.

We can only wonder if the speed of password perception is also associated with selection time – are skilled PASDJO-players capable of *choosing* a password quickly? If users take less time to sign up to a given service, they might be using mental shortcuts to accomplish

the tasks. There is much evidence that contextual cues, the preferred password, or the composition policy are indeed used as mental shortcuts. Investigating perception speed with selection speed poses new research questions that have not been addressed in the literature as such.

#### 5.4.4 Changing users' mental models with PASDJO

The learning curves in Figures 5.6 and 5.10 clearly indicate that players learned assessing password strength after playing PASDJO. Most notably, players learned to estimate passphrases. Thus, if the *authority* principle succeeded as a persuasive strategy, this might have influenced the users' mental model around passphrases. More generally, this finding confirms that mental models of password strength are not static. Therefore, PASDJO could be actively utilized to educate users about password strength, especially passphrases. Persuasive interventions could challenge users to rate the strength of a passphrase in the opportune moment and then debrief users to correct their mental model. We created such an intervention with our studies on the Decoy effect (see Chapter 10), which indeed show the predicted influence on mental models.

#### 5.4.5 Limitations

We discussed the most important limitations regarding the methodology in Section 5.2.3. Therefore, we focus on the limitations in the data and test results in this section.

From Figure 5.7, it is evident that most games were played at special events at the university or at conferences. Therefore, we can safely assume that our sample consists mostly of students, their peers, and academics. Thus, we must be careful not to expect the same strength assessment from other user groups or larger populations. Nonetheless, many players in our one-year trial might not have a background in cybersecurity and thus, we are confident that the sample still represents a useful cross-section of a diverse enough population.

Moreover, version 2 introduced a number of improvements that would have been nice-to-have already in the earlier version. Still, in the first version, the different conditions were easier to tell apart. Thus, causal effects are easier to isolate.

Finally, the strength ratings in PASDJO are solely based on zxcbn scores. While it is a state-of-the-art metric (see Section 2.3.3), other metrics might model real-world attacks better in some situations. For instance, we could have chosen Melicher's JavaScript implementation of neural network strength estimation [233]. However, it is more challenging to integrate and also does not map password strength to a scale from 1 to 5, but to a guess number. The same goes for other estimators. Therefore, it is also not immediately possible to conduct secondary analyses because players' coarse ratings are hard to map to fine-grained guess numbers. Nonetheless, zxcvbn's assessments correlate strongly with the output from PGS. Therefore, we are confident that zxcvbn is the best tool for the constraints of our game. In

---

the future, it might be worthwhile to explore useful solutions to include multiple strength metrics to obtain a more differentiated strength assessment.

## 5.5 Summary

In this project, we explored password strength perceptions in a novel approach. Earlier surveys had shown that users do not always have an accurate understanding of what contributes to the strength of a password. Rather than conducting a survey, we implemented a novel password game, which allows for better randomization of the measurements. Moreover, the game has unique potential to measure learning effects of the players. The source code of PASDJO is openly available<sup>8</sup>, so other researchers can adjust the parameters to their own needs. For instance, PASDJO was customized for a competition as part of a Google-internal event in October 2017.

We contribute insights from a longitudinal field study. Our sample comprises more than ten thousand strength estimations, which allows for precise analyses. PASDJO players accurately recognized the strength of commonly used passwords (RQ1). Furthermore, players largely underestimated the strength of multi-word passphrases, so this was the most error-prone password topology (RQ2). However, if they went on to play multiple games, they adjusted their mental model and rated such passphrases higher. This clearly demonstrates learning effects induced by playing PASDJO. Users were also misled by leet (l33t) substitutions in otherwise predictable passwords, so they slightly overestimated their strength. However, all in all, players performed better than anticipated which might be due to the constellation of the sample. On the other hand, educational efforts from the past decades might have somewhat paid off, which brings us back to our initial motivation and research objectives: The folk model of a “naive” user seems wrong. Current password meters and similar feedback systems thus might fail to persuade users because the information they provide is redundant or well-known. Therefore, we posit that feedback on password accomodate the fact that people often act deliberately and rationally. For instance, if service providers aim to persuade users to create a unique passphrase, the benefits need to be more *salient* and made *available*. Passphrases are a feasible alternative for master passwords in password managers, therefore we believe that persuasive techniques are viable for them.

### Future Work

The game presented in this chapter can be used as-is to answer additional research questions. For example, it would be very interesting to measure how much playing PASDJO influences actual password selection. To measure this, participants could be invited to a user study in the lab with a cover story. Once arrived, they are asked to sit in the waiting area until the experimenter picks them up again. PASDJO is installed as a demo in the waiting area and

---

<sup>8</sup> <https://github.com/TobiasSeitz/pasdjo> (last accessed 24.01.2018)

there could be a monetary incentive to “beat the highscore”. Afterwards, they complete a usability test where the focus is steered away from the sign-up form. In such a study setting, it should be possible to measure password selection under PASDJO’s short term influence. To study the influence in the wild, the feature set of the game could be reduced and made part of an onboarding process to a password manager. If necessary, one could think of another cover story, e.g. *please wait while we set up your system, you can distract yourself with this tiny game while you wait*.

The game can also evolve in the near future. For instance, players could be taken through a few trial passwords to make sure everyone has the same understanding of the task. To incentivize participation, additional game elements like challenges and quests are thinkable. Allowing players to set up profiles serves the *personalization* principle and also could be used to collect basic demographic information.

### Take Aways

- Users can identify weak passwords, but overestimate the strength of mangled passwords.
- Passphrases are seen as weak, but it is possible to influence users’ mental models with persuasive design.
- A game-based approach is feasible to collect data about psychological context factors inexpensively.

---

# 6

## Password Policies and Reuse

In the previous chapter, we reached the conclusion that users' mental models of password strength are fairly accurate with some exceptions. We can thus expect that users at least try to create strong passwords if they deem the account worth protecting. However, this is only one side of the coin: Even if users succeed in creating a very strong password, they still have to manage a multitude of accounts. Therefore, password reuse is rampant. Although it is necessary to some degree [116, 389], it remains hard to defend against. At the same time, password reuse might expose users to an even greater risk than weak passwords. In case an attacker obtains a user's plain text password, they gain access to all accounts that share this strong password. Studies have shown that users tend to underestimate the risks generated by password reuse [186]. In fact, password reuse can render the security advantages of picking a very strong password void, which is not immediately understood by users.

As explained in Section 3.3.1, password composition policies are one of the interventions targeted at weak passwords. However, in many cases users fulfill requirements in predictable ways, so policies often fail their primary goal. If password policies do not always benefit password strength, could they at least prevent password reuse? Heterogeneous policies across different web sites with mutually exclusive requirements could effectively disallow users to reuse passwords like they tend to do. In this chapter, we investigate password policies of one hundred of the most-visited web-services in Germany. This extensive audit should answer our research question "*Do password policies prevent password reuse?*" We shed light on the findings and discuss them in the context of supporting password authentication.

### Publication Statement

The dataset and the analysis have been previously published at CHI 2017 together with Manuel Hartman, Jakob Pfab, and Samuel Souque [288]. For the most part, the chapter picks up the content of that paper and puts it into the larger context of this thesis.

---

## 6.1 Background and Context

Password reuse is a major threat because it is easy for attackers to compromise many accounts at once. Even if users try to slightly modify their base password, attackers are still able to crack a large portion of the resulting passwords [67, 175]. There is mixed evidence about the subset of passwords that are reused more often, but generally one can identify a “go-to password” for regular sites, “high-value passwords” for important sites, and a “don’t care” password for the rest [17, 111, 154, 316, 317, 345, 363].

Password policies were originally designed to combat weak passwords, but some of them try to steer users away from reused passwords. Usually, this is done through black lists that block passwords that have already been exposed after a data breach. If a user tried to reuse a password which has been leaked, the system can detect this and enforce the creation of a new one. However, Habib et al. showed that users perform predictable alterations to circumvent the black-list filter [146]. In total, they identified 13 modification techniques. For instance, participants in the study added digits, symbols, words or letters. Habib et al. conclude that blacklists are only useful, if a user’s password modifications are not obviously based on a blacklisted word. Segreti et al. evaluated a different approach to combat reuse, known as the “Popularity is Everything” system [285]. Here, a password becomes blacklisted after a certain number of users have used the same password. Reuse in this case means reused by many users, instead of a single individual reusing the credentials multiple times.

In most cases, it is impossible to display the full list of blacklisted words in the user interface, so many service providers cannot do this. Thus, reverse engineering by testing different passwords is required to draw conclusions on a provider’s policy. Florêncio and Herley audited policies of public institutions and high-traffic website this way [112]. They found that on-line retailers have much looser password policies than government or university sites. Wang and Wang similarly checked the policies of 50 representative websites [361] to estimate the resulting entropy of the passwords. They took passwords from leaked datasets and picked 16 passwords with varying hypothetical strength. Unlike us, they did not aim to identify black-lists or special forbidden character types. Carnavalet and Mannan managed to automate dictionary checks by leveraging keep-alive connections [48]. However, they focused on server-side strength estimations rather than blacklists per se.

Reverse engineering password policies is tedious and might even violate the terms and conditions of a website. Thus, it would be favorable to compile a repository that contains all policies. This way, double efforts can be avoided, and service providers could be more transparent by specifying their own policies. The “policy-repository” would also be useful for a password generator that takes the site’s policy into account to avoid rejected passwords. In an effort to create such a repository, Steves et al. proposed to crowd-source the data and defined a formal language (based on XML) to describe policies [313]. The idea was later picked up and extended by Horsch et al. [166].

In summary, we conclude that certain policies based on blacklists can help in mitigating password reuse to some degree. However, blacklists entail usability issues because users



have no way of finding out which passwords are actually on the blacklist of a given website. For our project, this lack of transparency makes it hard to find out and compare policies of different websites, which demands a careful development of a suitable methodology.

## 6.2 Method: Reverse-Engineering Password Policies

To create a rich useful data set for studying password policies, we need to address two central aspects: selecting representative web sites and determining appropriate passwords. The first part is relatively straight-forward. We took Alexa.com rankings as indicator for the popularity of a website. To accomplish the analysis in a reasonable time frame, we took the 100 most visited web sites in Germany, which is 100% more than Wang and Wang, and 33% more than Florêncio and Herley. As of May 2016, out of these 100 services, 83 allowed public online registrations. The remaining 17 sites were banks, mobile carriers, or pay-tv providers who require offline registration and verification as a security measure. The websites and their policies are listed in Table 14.1 (Appendix). Although we took the most-visited websites in Germany, the results have implications for an international audience, because many of the audited sites operate globally.

The second challenge is finding suitable passwords to reverse-engineer password requirements. We approached this task in two separate stages to find a good candidate set.

### First Stage: Identification of Suitable Passwords

Similar to Wang and Wang, we crafted 15 passwords showing typical password characteristics. The passwords followed common policy categories as proposed by Shay et al. [295]. For instance, some passwords met a *3class12* policy by including three different character classes (lower-/uppercase letters, digits) and a minimum length of 12. Next, we tried to register new accounts with all of these 15 passwords at all 83 websites. In case the site rejected the password, we investigated the reasons and modified the password until the policy was fulfilled, e.g. by adding or removing characters. This new password was then added to our test set. During this stage, a number of sites revealed blacklisted symbols. If this was the case, we intentionally crafted a password with the blacklisted characters and added it to our set, so that we could see if other websites implicitly utilize the same blacklist. Similarly, if there were maximum length enforcements, we added a password longer than that to our list. This process resulted in a test set of **46** diverse passwords (see Figure 14.4, Appendix). The list was structured by the following criteria:

---

<b>length</b>	minimum and maximum length restrictions
<b>character classes</b>	the presence of enforced character classes, i.e. mandatory, forbidden, and allowed characters
<b>complexity</b>	the most stringent policy that the password would fulfill, as classified by Shay et al. [295]. The categories were <i>basic</i> , <i>2class</i> , <i>3class</i> , and <i>complex</i>
<b>dictionary</b>	the presence of a pro-active dictionary check, including common passwords.
<b>additional requirements</b>	black-/whitelisted symbols instead of a whole character class, additional requirements like large enough edit-distance from username

## Second Stage: Re-Evaluation with Extended Test Set

After identifying suitable passwords, we tried to use all of them on each website, i.e. we performed a total of  $46 * 83 = 3813$  checks. To avoid re-creating accounts with new email addresses, we tried to reset passwords wherever this was offered. We treat top-level domains that implement a Single-Sign On scheme (e.g. live.com, msn.com, microsoft.com) as separate services, because users might not know that the accounts belong together.

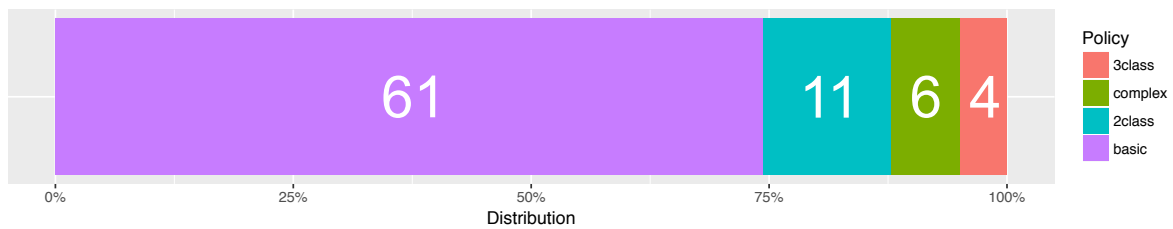
## 6.3 Results

We found it was possible to create passwords that would meet 82 of the 83 policies (~98.88%). In the following, we illustrate why this is possible.

### 6.3.1 Complexity

Most policies fell into the “basic” category. This means that their sole requirement was a minimum (or maximum) length. As shown in Figure 6.1, around three quarters of the sites used a basic policy<sup>1</sup>. Eleven web sites (13.3%) specifically require at least two different character types (*2class*). However, ikea.de requires letters and digits, but would not count a symbol towards the two character classes. We still opted to categorize this as a *2class* policy (see remark in footnote 1). *Complex* policies in the wild have further restrictions, e.g. dictionary checks or special rules. 23 websites (27.7%) used a dictionary check. Bahn.de demanded three *different* characters, i.e. a password like annnna would be rejected, but banana would not. Paypal.com disallowed using the same character three times in a row, which rejects a couple of German compounds like Schiffahrt. Nevertheless, it is easy to find a password that fulfills the complexity requirements of all 83 websites: “D.ssertation18” is a 4class password that would pass all complexity requirements.

---



**Figure 6.1:** Distribution of minimum complexity of the policies. Most sites only require a given length.

## Length requirements

The length requirements and restrictions were fairly inconsistent in the test set. The average minimum required length was  $M = 6.3$  ( $SD = 1.9$ , see Figure 6.2). No website had a minimum length greater than nine characters. Among the top 10 most-visited sites, facebook.com, amazon.de, ebay.de allowed six-character passwords. Security-wise, this is alarming, because even system-generated 6-character passwords can be brute-forced in a matter of hours in an offline attack<sup>2</sup>. Wikipedia.org, which is also in the top 10, had a minimum length of one character. Interestingly, two tech-oriented websites allowed the same (heise.de and chip.de). Perhaps the service providers expect their technical audience to create stronger passwords anyhow, and they also do not store much personal information. We were surprised that 40 sites (48.2%) imposed a *maximum* length restriction, which is counterproductive in terms of password security. The average maximum length was  $M = 43$  characters ( $SD = 32$ ). Ten websites rejected passwords longer than 20 characters, so a considerable number of passphrases would be excluded ( $\text{length}(\text{correcthorsebatterystaple})=25$ ).

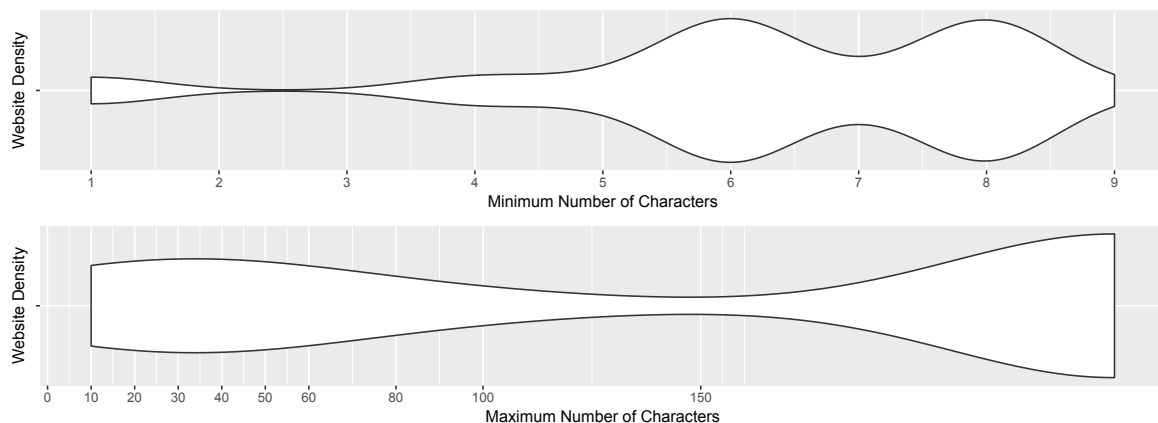
In order to effectively prevent password re-use, the maximum length on one site would need to be lower than the minimum length of another site. This was not the case in any permutation of policy pairs. The closest difference was the maximum length at ikea.de (10 characters) and the minimum length at yahoo.com (9 characters). Thus, only a nine or ten character password can be reused on all the tested websites.

## Character Sets

Although the websites at the top of the table, i.e. Google, Facebook, Amazon, all use a *basic* policy, there is still a high chance that they reject passwords containing certain characters. We found it was common to disallow non-ASCII symbols (see Table 6.1). For instance, Google rejected any passwords with non-ASCII characters. So, even dictionary words that include letters from non-English alphabets, e.g. the German umlauts ä, ö, and ü, will be

<sup>1</sup> In the CHI publication, we reported slightly different numbers: 57 *basic*, 11 *2class*, 3 *3class*, 1 *complex*, 10 *other*. It is in fact possible to put the *other* policies in the remaining found categories, which explains the updated distribution

<sup>2</sup> <https://arstechnica.com/information-technology/2012/12/25-gpu-cluster-cracks-every-standard-windows-password-in-6-hours/> (last accessed 27.01.2018)



**Figure 6.2:** Density distribution of password length rules. We excluded maximum lengths beyond 245 characters, which explains the hard cut-off in the bottom plot.

rejected in this case. Some sites already provide a list of characters that are either allowed or disallowed at registration time. In the case of mobile.de, the list of allowed symbols is even smaller. On the positive side of the spectrum, twitter.com accepted even extended Unicode characters like emojis.

A few websites did not reveal the list of allowed non-ASCII characters; some of our passwords were blocked if they contained certain characters. Thus, we had to remove character by character to find out which of them was blacklisted. For regular users, this process would be exceptionally tedious if their password contained a character from an unknown blacklist. Our audit shows that spaces and the tilde character ‘~’ are the most likely culprits in this situation, which was the case at netflix.com, spiegel.de or welt.de. Finally, there were mutually exclusive policies: Lidl.de proactively shows a list of symbols from which the user has to choose at least one to create an account. However, all those symbols were disallowed on at least one other website. In that sense, if a user’s go-to password is immediately accepted at lidl.de, chances are that it is rejected at a couple of other websites. So, the policy at lidl.de prevents password reuse, but perhaps at the cost of frustrating users with an exotic rule.

## 6.4 Discussion and Implications

In the following, we shed light on what the results mean in terms of password reuse.

### 6.4.1 Policies are Mostly Homogenous

Overall we can state that in-the-wild policies are largely similar. However, the devil lies in the details. It is not quite clear why almost half the websites enforce a maximum length restriction, albeit a high one in most cases. Shorter passwords are easier to guess in an

Web site	Whitelisted symbols	Blacklisted symbols
ebay.de	!@#%*^*_-_+=	
web.de	!#%&()*+,-./:;<=>@[\\]^_{ }~\$ÄäÖöÜüß	
gmx.net	!#%&()*+,-./:;<=>@[\\]^_{ }~\$ÄäÖöÜüß	
t-online.de	!#%&()*+,-./:;<=>@[\\]^_{ }~	
live.com	@#%*^*_-_+=	
mobile.de	!\$%&*?-_+#	
pornhub.com	/_	
lund1.de	@#%*^*_-_+=	
chefkoch.de	äöüÄÖÜß, _ , -, !?&.	
zeit.de	äöüÄÖÜß ,.!?:;#&* ()_+<=>-	
lidl.de	@#%*^&+<=>.-!?	
spiegel.de		'space' 'new line'
outbrain.com		'space'
welt.de		'space'
netflix.com		~

**Table 6.1:** Many websites only allow ASCII characters. This table shows the sites that restrict the ASCII set even further. Some sites are mutually exclusive regarding the usage of certain symbols only by looking at a small subset of the tested sites, e.g. netflix.com and t-online.de. The full rules can be found in Appendix 14.1

offline attack. It is therefore almost necessary to lock out users after a number of failed login attempts to at least mitigate targeted online-guessing (see Section 2.2). Blocking non-ASCII characters further impedes users' password selection. Although the reasons for doing so are anything but obvious – allowing more characters increases the theoretical password space and thus potentially security – we interpret this restriction as a usability precaution. ASCII characters are part of virtually all physical keyboard layouts, but extended alphabets are not. For instance, if a German user's email password were "DieÄrzte123", they would be troubled to log into their account from a PC while abroad, because the umlauts are missing on the keyboard. Of course, there are ways to enter such characters, but they require much more effort under these circumstances. The user experience of the entire login process would suffer, which is the reason for blocking the characters in the first place. However, with the increasing number of personal mobile devices that also work abroad, the motivation to block non-ASCII characters crumbles. Artificially blocking non-alphanumeric ASCII symbols, like \$ or ;, in any case creates unnecessary burden for users. As we have seen in the previous chapter, users attribute password strength to such symbols and forbidding those can lead to confusion and erroneous mental models. Perhaps the restriction was introduced to prevent site vulnerabilities, e.g. SQL injections. But this is a prime example for a misguided security approach that shifts effort to millions of users rather than to a couple of security engineers.

---

## 6.4.2 Policies do not Prevent Password-reuse

Neither length, nor complexity requirements were heterogeneous enough to prevent password reuse. In the end, only artificial character-class restrictions narrowed down the list of passwords that can be used everywhere. We found that a nine or ten character password that includes at least one uppercase letter, one lowercase letter, and one digit would have been accepted by 82 of the 83 tested web sites. To circumvent dictionary checks, it is recommendable to intersperse digits in the middle. Thus, if users pick a password like `s1lverPWD`, current in-the-wild policies do not stand in their way. With an edit distance of 1, many rejected passwords can be turned into an accepted password if the rejection was caused by a special character. Hence, the criteria for the most-reusable password (what we paraphrase with “golden password”) are very well-defined and narrow. Any password that does not meet these criteria will generate usability issues for users, e.g. if they try to use longer passwords or ones with a richer character set. Automatic password generation is hampered in many cases by length limitations. In that sense, policies help to prevent re-use of “extreme” passwords that are either exceptionally weak or stronger than average. Reuse of “normal” passwords that work everywhere is not prevented.

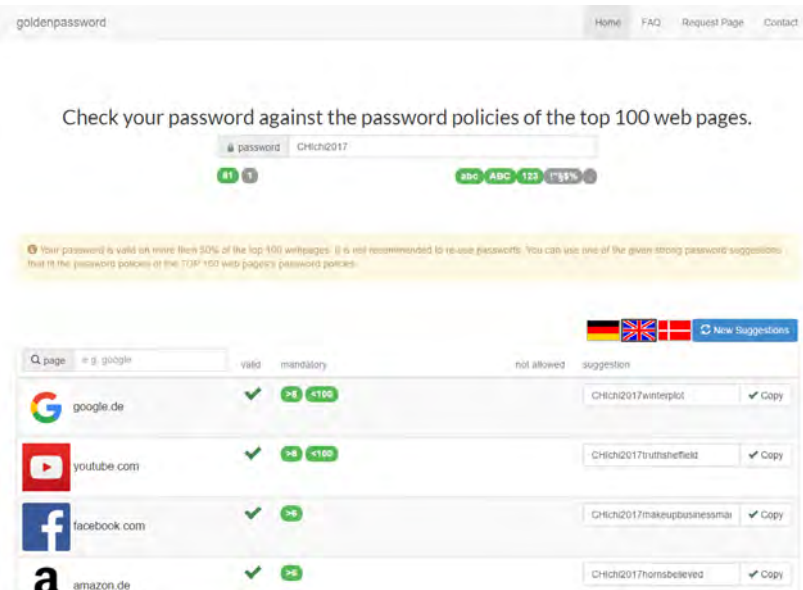
As a consequence, a policy could be adjusted dynamically if the user signs up with a “golden password”. It is very likely that a password showing the above described characteristics is reused across many web sites. As Florêncio et al. pointed out, this is not necessarily a bad thing [116]. However, if a high-value password is reused for an unimportant account, this interference could lead to problems. Stobert showed that experts are less prone to this threat [317], but regular users cannot always estimate the importance of an account up front. As a consequence, dynamically adjusting the policy could be a solution. Alternatively, it might be feasible to more prominently warn users about reuse and explain the implications. As shown by Ur et al., displaying specific solutions can help in this situation [341].

## 6.4.3 Smarter Password Generation

Our data clearly shows that using software to generate random passwords can become troublesome for users who opt for this kind of selection support: many policies restrict the length and allowed characters. Although many password generators enable the users to adjust some parameters of password creation (like Apple’s system shown in Figure 3.4), more effort is required to find the right parameters. Thus, a system designed to take away cognitive effort becomes effortful once more. A better solution is to use contextual information for password generation. Typically, password managers have built-in generators. To avoid that users have to adjust parameters, a password manager could automatically retrieve the policy for a given website (context) and generate a strong password that fulfills it. As a proof of concept, we built a web-based prototype that demonstrates how contextual policy information can be used for password generation<sup>3</sup> (see Figure 6.3).

---

<sup>3</sup> <http://jakob-p.github.io/goldenpassword/> (last accessed 26.01.2018)



**Figure 6.3:** We built a small web app to demonstrate how contextual policy information can be used to a) generate suitable passwords and b) find out which web sites accept a given password.

## 6.4.4 Limitations

Although the data can be very useful for password generators and for the design of future composition policies, it is limited in some ways. First, the list of web sites is not comprehensive, we merely observed a tiny snapshot of the high-traffic websites, which already required days of work for the entire team. We also could have investigated the top websites in different categories to get an even more representative sample. However, the current dataset already depicts the state of affairs in reasonable detail, and is comparable to similar publications [112, 361]. Furthermore, the longest password in our test set had a length of 246 characters. We concluded that there is no length restriction if this password was accepted by a given website, but there might just be a higher restriction. Nevertheless, since only a fraction of users use passwords this long, this limitation has almost no consequences.

Moreover, for several reasons we did not include emojis in all our tests although they are in fact part of the unicode character set. First, emojis break some input fields, because most of them are encoded with 4 bytes instead of 2. This means that they show up as two characters in the input field, which distorts length requirements. Furthermore, not all websites were encoded in the extended UTF32 standard and hence failed to submit the data.

Lastly, password policies change over time. Since carrying out our research in May 2016, we found changes in several of the assessed policies when we randomly re-sampled them. For instance, idealo.de used to enforce a *complex* policy, but they have switched to a *basic6* as of January 2018. Therefore, our data has a limited lifespan. The big players on the list, like Facebook, Google, or Amazon, are slower to enforce new policies due to the even larger



---

user base and business impact. An automated process similar to the one from Carnavalet and Mannan [48] could help continually validate the data in the future.

## 6.5 Conclusion and Future Work

In this project, we extensively audited password policies of the top 100 web sites in Germany. The data set is published for further analysis on GitHub<sup>4</sup>. 83 of the websites offered public registration, and we could reuse a single password on 82 of them. Hence, we were able to answer our main research question “*Do password policies prevent password reuse?*” with a resounding “**no**”. Thus, we further question the use of strict password policies as a means to influence password selection. It has been shown that they do not necessarily lead to stronger passwords. Now we have shown that they also do not prevent reuse. As a consequence, it would be easier for users if restrictions on password selection were loosened. Restrictions enforcing a maximum length or a certain set of symbols should be abolished to ensure universal password generators integrate with the websites. A *basic8* policy appears to work for the major players, so more service providers can follow them. To account for the potential security vulnerabilities, blacklisted common passwords and adaptive blacklists are promising solutions [146, 285]. Future research should thus evaluate the specific blacklists and user support to help them find alternatives in case their password was blacklisted.

### Take Aways

- Most password policies in the wild used a basic policy with length as the only requirement.
- About half of the policies enforced a maximum length, which appears counter-productive in terms of both usability and security.
- It was possible to reuse a nine or ten character password that includes uppercase and lowercase letters, and at least one digit on 98% of websites in the test set.
- Only the requirement of particular symbols generated conflicts between policies.
- The above characteristics also tell us that any user with a different selection strategy faces a usability backlash. Acting more securely by generating passwords automatically is hampered because some policies reject such passwords.
- All bullets above support the demand for simplified password policies to remove important pain points of password authentication.

---

<sup>4</sup> <https://github.com/mimuc/password-policy-dataset> (last accessed 25.01.2018)

# 7

## Understanding Password Practices through the Lens of Personality Traits

Although there are certain general patterns, password selection is a task that each individual handles in their own way. We have shown in Chapter 5 that password strength is *subjectively evaluated* depending on certain characteristics whose benefits users interpret in different ways. Again, while there was an overall tendency, we were unable to break down the strength ratings by such individual differences: was there a special user group who performed better than others in the game? What characterizes this user group? Moreover, our previous chapter shed light on password policies in the wild. In a few notable cases, the rules were challenging and reject a substantial part of passwords. Users have developed strategies to cope with rejected passwords, but it would be interesting to know the exact factors that contribute to their behavior in these circumstances. Egelman and Peer make a strong argument that there is no “average user” so it is necessary to look at individual differences to understand user behavior [98].

Demographic background is one of the external factors that influences password selection [226, 351, 359]. Personality traits have been brought into the discussion to explain user preferences, actions, and behaviors in security questions [42, 143, 304, 387]. Especially in research about phishing susceptibility we find evidence that personality traits have the potential to explain behavior [148, 149, 252, 339]. Empirical results from password studies have been discussed and explained with different personality traits, too [153, 368]. It is evident that a user’s personality shines through when they select a word with personal meaning. Petrie classified users in distinct password personalities: family-oriented, fans, fantasists, and cryptics<sup>1</sup>. A LastPass report more roughly divides password usage into two groups [208]: Type A users want to stay in control and are driven to act securely, so they developed an elaborate system that they perceive as suitable. Type B users do not believe that their accounts are valuable to attackers, so they do not prioritize security over usability.

---

<sup>1</sup> The original survey is not available online anymore. In a personal inquiry with Ms Petrie, she said that the original data is with the firm who commissioned the survey. The aggregated statistics are available at <http://passwordresearch.com/stats/statistic130.html>, (last accessed 29.01.2018)

---

Those two taxonomies stem from analysis of user-selected passwords, i.e. a retrospective evaluation. However, predictive approaches are under-explored. For instance, if a user is generally an emotional person, does this impact their password selection strategies? If a user is diligent in real life, do they invest effort to diligently craft passwords, too?

In a series of user studies we explored the associations between personality traits and passwords. If such associations exist, they open a new range of support systems that are tailored to a user's personality. Current one-fits-all approaches could be re-designed radically. The research was carried out in cooperation with three students. In each separate project we focused on a different stage of the password life-cycle. Timo Erdelt investigated personality as predictor for the usability of composition policies [101]. Paul Huber explored correlations between strength perceptions and personality traits [168]. Finally, Aline Neumann examined personality factors in password selection [241]. In total, 440 individuals participated in three separate online studies. In the following sections, the projects are put into context and their findings are discussed on a bigger picture.

## Research Objectives

Our primary goal was to find ways to predict password behavior from personality traits. At this point, the discourse about risky passwords included personality factors as hypothetical explanatory variables. However, only few empirical studies had been carried out to challenge the assumptions. We aimed to provide such empirical data and a discussion of the implications on the design of password policies and password authentication systems. For instance, adjusting requirements of password policies depending on the user's personality promises to reduce frustration of password selection. Thus, our over-arching research question can be framed as *“Does personality influence a user's mental models of password strength and consequently selection and coping strategies?”*.

## 7.1 Background and Related Work

In this section we give a brief overview about the characteristics of strong passwords and how users go about creating them. Moreover, we portray projects in usable security and privacy research in which the users' psyche has been the focus.

### 7.1.1 Sociodemographic and Cognitive Factors

Apart from such conscious behavior, there may be other preconditions that make some users pick stronger passwords than others. In a large field study, Mazurek et al. found that computer science and engineering students created passwords that were less guessable than those from business or politics students [226]. Beyond demographic background, context factors like the emotional state during password selection have also been investigated. Gulenko

examined the effect of presenting positive textual messages and icons during password selection and found benefits for the adoption of passphrases [144]. In contrast, putting users in a state of cognitive distress or depletion made participants choose weaker passwords in a large lab study [143]. Social pressure as another type of psychological leverage was investigated by Egelman et al. [99]. While they argue that account value plays a superior role for the effectiveness of password meters, others have shown that the *design* of a password meter does have a measurable impact on the effort users put into creating a password [343]. In summary, the literature shows that password selection depends on context factors beyond education and experience.

### 7.1.2 Personality Factors in Cyber Security

In our work, we are interested in context factors of password strength originating from psychological variables like personality. One of the most commonly used models to characterize personality are the Big-Five traits (B5), also known as the five-factor model. Costa and McCrae [60] refer to the personality traits as *openness to experience*, *conscientiousness*, *extraversion*, *agreeableness*, and *neuroticism* (OCEAN). The traits can be described with these exemplary adjectives [229]: **Openness:** imaginative, creative, curious, independent, liberal **Conscientiousness:** careful, reliable, ambitious, scrupulous, neat, punctual **Extraversion:** sociable, talkative, passionate, warm **Agreeableness:** selfless, helpful, forgiving, cheerful, humble **Neuroticism:** worrying, emotional, insecure, impatient, vulnerable, subjective

Most frequently, the influence of these personality traits have been explored for privacy-concerns, where the openness trait was associated with privacy attitudes [96, 235]. Other inquiries have shown that personality traits like neuroticism [148] or openness [339] might be associated with the response to phishing attacks. The likelihood of employees adhering to security policies is potentially influenced by the manifestation of agreeableness and conscientiousness [304, 305]. These investigations show that personality models are a considerable factor in security and privacy. Yet, our understanding of the influence of personality on password perception and consequently password selection is still low. Our work tries to improve our understanding about the origin of the differences in users' judgments of password strength.

## 7.2 Study 1: Policies

We start out with the exploration of psychological factors for the design of password policies. We were motivated by the fact that, at this point, policies are a one-fits-all solution that evidently does not work in the same ways for all users: Shay et al. observed that subjective usability ratings for policies differed among participants [297, 299]. For instance, about 40% of their participants found it difficult to create a password under a “3class16” policy, but another 40% found it easy [299]. Following the general discourse and related results

---

from privacy research, we hypothesized that an individual's personality might be responsible for their attitudes towards one policy or another. Therefore, our goal in this project was to explore such associations between personality traits and policy preference. At this point, we leave out analyses on password strength.

## 7.2.1 Method

Our study was completely exploratory, because the literature did not allow us to derive narrow hypotheses. Since personality traits are nuanced, we opted for an online survey to collect a large sample. Personality was assessed based on the Five-Factor Model. We opted for the very reliable BFI-K construct by Rammstedt and John [261], which is also freely available in German. Moreover, with its 21 items, the time to fill out the questionnaire is kept reasonably low. Participants were asked to create several passwords in a row, i.e. the study followed a within-groups protocol. Here, we evaluated three different password policies: a traditional (3class12), an uncommon (2word12), and a novel policy (emoji12) that required the selection of at least one emoji through a graphical user interface (more on emojis in Chapter 11). The reason for this choice was that the policies are different enough to serve as characteristic levels of the independent variable “policy”. Also, creating passwords that exceed the length requirements of typical policies (see Chapter 6) is more difficult, so we could better measure the differences of perceived difficulty. Participants assessed the “difficulty to create” of a password for each policy. Moreover, we had them rank the policies by their personal preference, so the distinctiveness of 3class12, 2word12, and emoji12 would help them spot and judge the differences easily, which makes the data more reliable.

### Structure and Tasks

The study was divided into 3 overall parts. In the first part, participants were briefed about privacy details of the study and they provided demographic background information. Then they proceeded to the personality questionnaire before they were asked to perform three experimental tasks. Each consisted of creating a password and assessing the difficulty with agreement levels on the three items “*It was difficult to create a password that meets the requirements*”, “*I found the password requirements bothersome*”, and “*It was easy to create a new password*”. Agreement was measured on a five-point scale ranging from “Strongly disagree” to “Strongly agree”. Inversely keying the items as well double encoding makes the data more robust against implausible responses. The resulting difficulty-to-create score thus ranges from 3 to 15 (3 = very easy, 15 = very difficult).

The order of the policies was counterbalanced during the experiment to mitigate order effects. For each participant, we recorded the resulting order as a control variable. We chose an online-banking scenario for all three selection tasks. The first prompt was to create a password to protect an online banking account. Secondly, participants were told that someone had gained access to their account and the bank locked them out. As a security precaution, they had to reset their first password. The last task description explained that their password

had expired after one year and they need to reset it again. This storyline was designed to fulfill the *realistic threat* principle proposed by Krol et al. (see Section 3.1.1) [205].

We used SosciSurvey, a standard survey tool, to collect the responses. The dynamic parts involving password selection were embedded in iframes. To match the data from the survey tool and the iframe we used URL query parameters containing the response ID. We asked participants to only use a desktop browser to avoid styling glitches and unexpected behavior from the prototypes.

## Recruiting and Demographic Background

We recruited participants through posts on social networks and by sending out the invitation link in a university-wide newsletter (more than 5000 recipients), which was due to time and budget constraints. To incentivize participation, we announced a raffle of five shopping vouchers with a value of 20€ each. At this point, 222 people had started participation. After drop-out and plausibility checks, the remaining sample size was  $N = 164$ . As expected, the age distribution was narrow: our sample consisted mostly of students in their mid-twenties (average age 24 (SD=5). 79 respondents were female, 83 male, and 2 preferred not to answer. In the background screener, 65 people (40%) indicated to possess formal training in computer science or information technology. We also requested self-reported assessments on password practices. Here we found that 40% reuse passwords without modification, 32% reuse them with modifications or with a mnemonic technique. 17% often create new passwords. In terms of management strategy, the majority (53%) tries to memorize passwords. 11% use a password manager or generator. Written cues served as aid for 10% of respondents, and 16% write passwords down on analog media, while 21% use electronic files.

## Statistical Analyses

For statistical analyses, we consulted the StabLab<sup>2</sup> to identify suitable methods. After a revision of the collected data and the necessary assumption checks, we analyzed associations by fitting generalized additive models (GAMs) to the dataset. Their advantage over linear regression is that they are more flexible for non-linear associations<sup>3</sup>. The *mgcv* package for R was used to calculate the models. GAMs can be primarily interpreted through residual/smooth plots – the steeper the fitted regression line, the stronger the association (The box in the results section (7.2.2) explains how to read GAM plots in great detail.)

Scores on the Big-Five sub-dimensions served as independent variables, i.e. the predictors in the regression models. *Openness* is coded with five items, while the remaining four dimensions were assessed with four items each. The agreement level for every item was mapped to numeric values from 1 to 5. The score on each sub-dimension is the sum of agreement levels. To better estimate effect sizes, we control for gender, age and IT proficiency in the regression models.

<sup>2</sup> <http://www.stablab.stat.uni-muenchen.de/> (last accessed 30.01.2018)

<sup>3</sup> [https://en.wikipedia.org/wiki/Additive\\_model](https://en.wikipedia.org/wiki/Additive_model) (last accessed 30.01.2018)

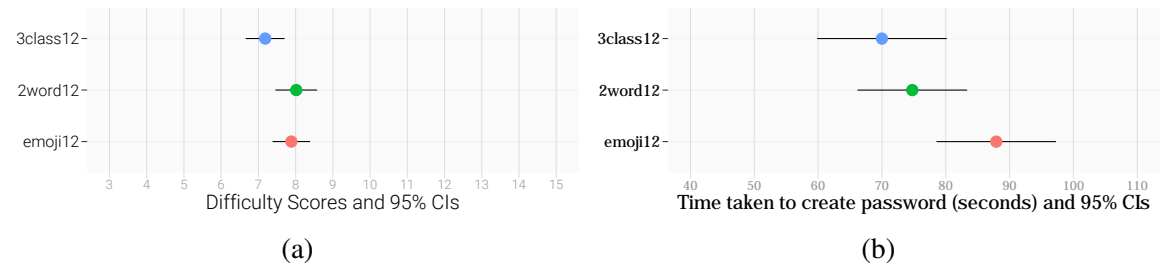
## Method Limitations

The method, albeit carefully executed, faces a few limitations regarding the interpretability of the data. First, the sample was fairly homogeneous, because participants were mostly between 20 and 28 years old and have an academic background. This might reduce statistical power in detecting effects on personality [310], but on the other hand, this constellation resolves age-related confounding effects. Moreover, our study was strongly focused on individual preferences and usability perceptions of different policies, so only a within-groups design was feasible. However, in real-life password selection, users rarely select three passwords in a row. The choice of our storyline still makes us confident about the ecological validity [103]. The repeated measures design did not allow us to measure the policies' influence on password memorability, which we have to postpone to another study. At this stage, the subjective preference was more valuable for our exploration than memorability effects. Besides, we briefed participants to fill out the survey on a desktop PC or a similar device. Since it is always possible to circumvent user-agent detection by requesting the desktop version, we cannot guarantee that all participants followed this instruction, which might have had an effect on their password selection [232].

Finally, we unfortunately made a mistake in the deployment of the emoji-based policy. Instead of 12 characters, it required participants to select 16 characters beside the emoji. We realized this fact by looking at descriptive statistics during the course of the study, because the policy performed significantly worse than the other two. We re-deployed the emoji-based policy immediately after we had realized the error. Consequently, we had to remove the data for the creation difficulty and ranking in cases 1-61, reducing the overall sample size to 103. Nonetheless, the sample size is sufficiently large to investigate medium to strong effects.

## 7.2.2 Results

Overall, associations between personality and policies were moderate. In the following, we only describe non-trivial and interesting associations.



**Figure 7.1:** Confidence Intervals for a) Difficulty to create and b) time to create passwords in each condition. The traditional policy (3class12) was the easiest and fastest overall, but not on a statistically significant level (also visible in the charts due to the overlapping confidence intervals)



## Descriptives and Independent Variables

Users rated the difficulty to create a password very similarly in all conditions (averages of scores in range [3;15]: 3class12 = 7.18, 2word12 = 8.02, emoji12 = 7.88). A linear mixed model ANOVA did not show significant differences ( $F_{2+2} = 5.28, p > 0.1$ ). Figure 7.1 shows the confidence intervals for these two usability metrics “difficulty to create” and “time to create”. Observing no significant differences overall is interesting because it means that individual ratings could be explained by personality traits.

## Creation Difficulty Models

The general additive models showed associations between the predictors and creation difficulty scores (see Table 7.1).

	<i>emoji12</i>		<i>2word12</i>		<i>3class12</i>	
(Intercept)	9.99		7.04		6.26	
Predictor	$\beta$	$\sigma_n$	$\beta$	$\sigma_n$	$\beta$	$\sigma_n$
Age	0.02	0.06			−0.05	0.06
Gender (female)	0.91	0.58	1.35	0.66	−0.46	0.61
No IT Background	−0.10	0.59	0.60	0.66	0.21	0.63
Extraversion	−0.05	0.08			−0.01	0.08
Conscientiousness	0.01	0.09			0.08	0.10
Neuroticism	−0.22	0.09			0.05	0.09
<i>emoji12</i> Position 2	0.52	0.65				
<i>emoji12</i> Position 3	0.23	0.63				
<i>2word12</i> Position 2			−0.38	0.74		
<i>2word12</i> Position 3			−0.07	0.73		
<i>3class12</i> Position 2					1.19	0.66
<i>3class12</i> Position 3					0.77	0.69
Explained Deviance	0.15		0.21		0.10	
Num. Observations.	119		119		119	
Num. Smooth terms	2		6		2	

*Description:*

$\beta$  = Correlation coefficient

$\sigma_n$  = standard error

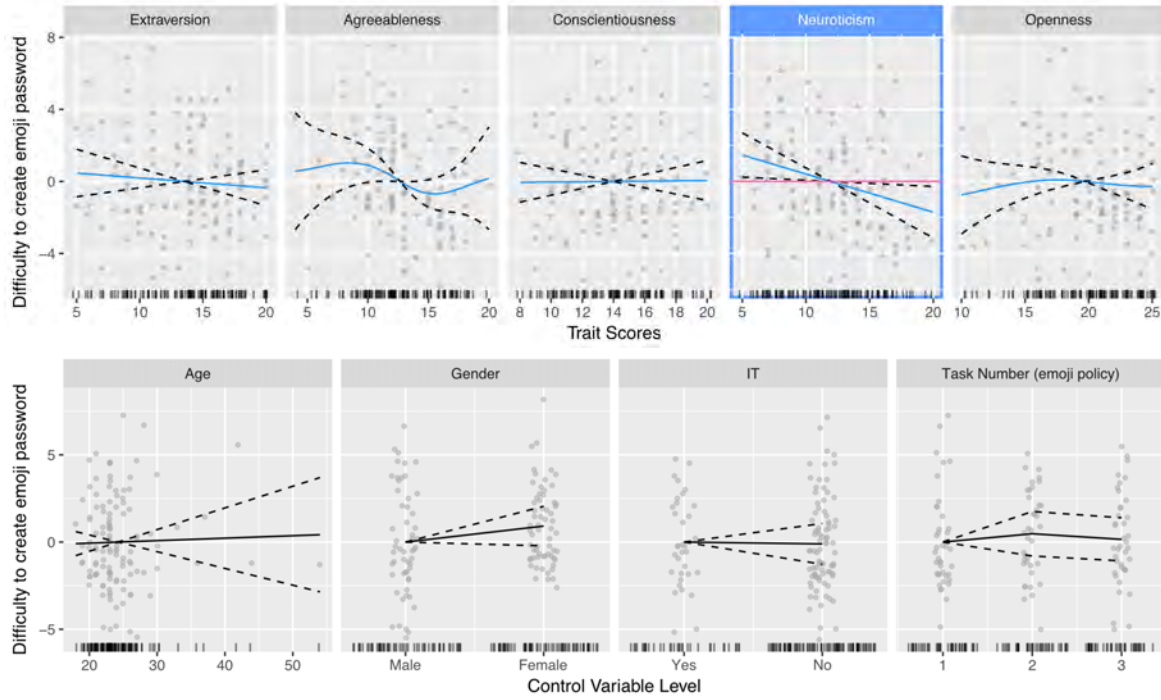
edf = estimated degrees of freedom by non-linear effects

(„estimated degrees of freedom“)

Smooth terms = non-linear effects in the model

**Table 7.1:** Additive regression models for the difficulty to create passwords under the three policies.

**Control Variables** The GAM allowed us to model associations linearly for the control variables (example for emoji12 in Figure 7.2). Although we have to be careful not to generalize too strongly with our sample, linear associations at least enable us to use correlation



**Figure 7.2:** Associations between predictors and **difficulty** to create a password under the emoji12 policy. Charts visualize the functions derived from the Generalized Additive Models (GAMs). Neuroticism was significantly negatively associated with difficulty (highlighted in blue), i.e. it was easier to create emoji-passwords if participants scored high on neuroticism.

coefficients  $B$  as basis for discussion. Female participants assessed it slightly more difficult to create passwords under the emoji12 ( $B = 0.91$ ) and 2word12 ( $B = 1.35, p < 0.05$ ) policies than male participants. For 3class12, the correlation was smaller ( $B = -0.46$ ) and pointed in the opposite direction. Having a background in IT positively showed medium correlations with difficulty in the 2word12 policy, too ( $B = 0.61$ ). This policy, albeit alphanumeric, is uncommon in the wild and ignores the verdict of high complexity – IT people might be skeptical about the “words” requirement, while others are less concerned about it. The task order also showed medium-strong influence on creation difficulty. If emoji12 or 3class12 were part of the second task, creation was seen as more difficult ( $B_{emoji12-pos2} = 0.52$ ), ( $B_{3class12-pos2} = 1.19$ ).

## How to Read Smoothed Regression Plots

Figure 7.2 shows the first example of regression plots that will be used throughout the thesis where appropriate. There is a separate plot for each marginal association with a given predictor. The solid curve/line represents the estimated regression curve. If it is a straight line, the association can be modeled linearly (e.g. neuroticism in Figure 7.2). If it is a “wiggly” curve, the association is modeled with polynomial terms of different degrees (e.g. Agreeableness in Figure 7.2). The model intercept is at  $y = 0$ . In many cases, residuals are plotted at their respective  $(x, y)$  position to give a sense of clusters. At the bottom border of the plots, we often add “rugs” to show the number of observations/residuals along the  $x$ -axis.

Interpreting the effect size and significant contribution to the model fit is visible in two ways. First, the slope of the fitted line shows estimated strength of associations. Second, there are two dashed curves surrounding the fitted curve/line that visualize 95% confidence intervals. Both curves are entirely above, respectively entirely below, the intersection between the fitted curve and the intercept at  $y = 0$ , the association is significant at the 0.05 alpha level.

In Figure 7.2, we highlighted this for the neuroticism plot. Left to the point of intersection, both dashed lines are above the intercept (in pink color); analogously, the dashed lines are below the intercept on the right-hand side, indicating a significant contribution to the model.

**Personality Traits** Contrarily to the modeling functions for control variables, we mostly observed non-linear associations between trait scores and creation difficulty (see Figure 7.2). One exception is the weak linear association in the emoji12 ( $B = -0.22$ ) condition. It tells us that it was slightly easier to create emoji-passwords for people with higher neuroticism scores. The neuroticism trait is also referred to as “emotional stability” [60], i.e. neurotic people are usually more emotional. Expressing *emotions* with emojis in passwords seems to support this trait and come easier to users scoring high on neuroticism.

In the models for the 2word12 policy, all associations were non-linear and thus inconclusive at this point. For the 3class12 policy, associations were linear but trivial for the extraversion, conscientiousness, and neuroticism factors. In summary, scores of the “Creation Difficulty” scale are difficult to explain with personality traits, but control variables appear to have stronger influence. Only neuroticism is associated notably with difficulty to create an emoji-password.

## Policy Preference

Table 7.2 shows the overall preference for the three policies. It is evident that participants generally preferred the 3class12 policy (68%). The emoji policy was best ranked by 18 participants, and 2word12 by 12 participants. Using logistic additive regression, we can

**Table 7.2:** Distribution of binary rankings of the three available policies. Evidently, 3class12 was ranked best in most cases.

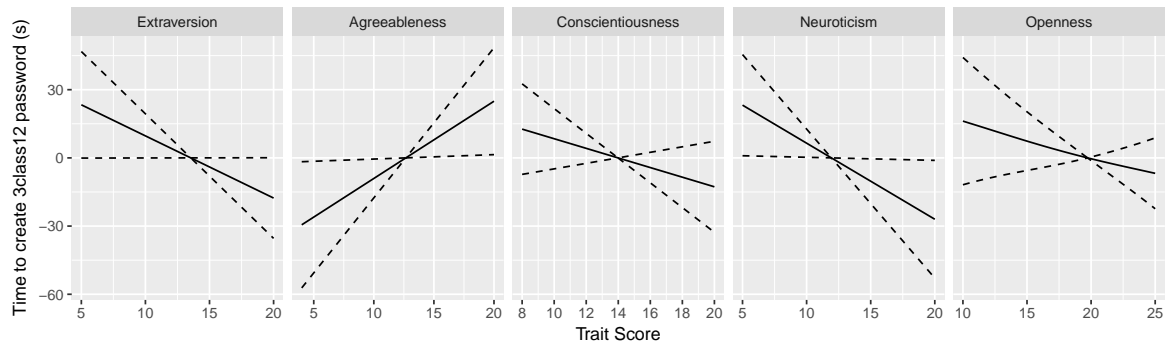
	emoji12	2word12	3class12
1st rank	18	12	67
other rank	80	85	31
n	98	98	98

determine the factors that contribute to these preferences. In essence, the model gives us the likelihood of voting a given policy to the top, which is a binary decision (1 = preferred, 0 = not preferred). Including demographic control variables as predictors, we observed strong effects for IT-background. The probability of putting emoji12 at the top is  $\exp(B_{\text{emoji} - \text{IT}}) = 9.87$  for participants without technical background, so around ten times higher. Only one respondent with an IT background ranked emoji12 on the top. The order in which policies were displayed also produced a notable effect. If emoji12 was part of the second  $\exp(B_{\text{emoji-pos2}}) = 0.27$  or third task  $\exp(B_{\text{emoji-pos3}}) = 0.76$ , the likelihood to rate it the best policy slightly decreases. Other task orders, as well as age-related preferences, were inconclusive.

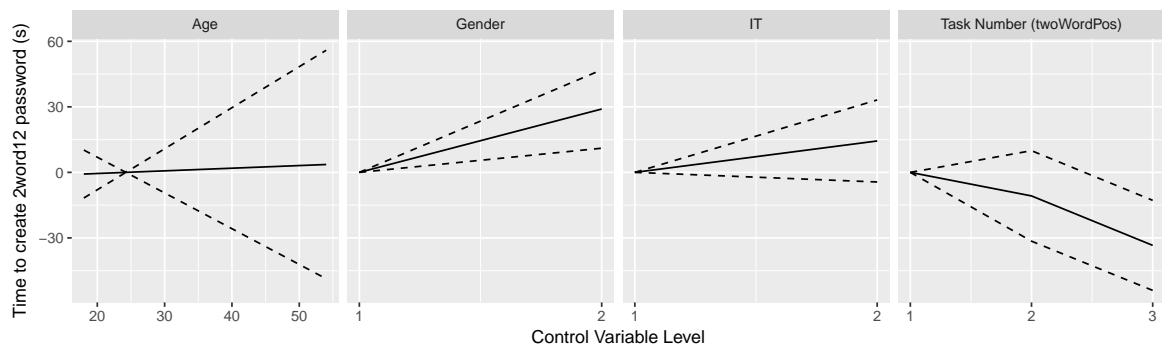
As with creation difficulty, rank-associations with personality traits were generally weak. The likelihood to prefer 2word12 decreased with higher extraversion scores  $\exp(B_{2\text{word12-E}}) = 0.82$ . High agreeableness scores entailed higher chances to vote for emoji12  $\exp(B_{\text{emoji12-A}}) = 1.28$ . Interestingly, only very high neuroticism scores ( $>17$ ) caused a stark increase in favoring the emoji policy. However, the sample is thin in this area so the model is more unstable at this boundary. Other associations were negligible or inconclusive.

## Time to Create Passwords

The second usability metric was the time the participants took to create passwords. It can also be interpreted as *effort* put into the task. The most interesting associations were visible for the 3class12 policy, where all personality traits could be modeled as linear predictors (see Figure 7.3), and all showed medium-strong correlations (details in the Appendix, Table 14.3). One might have suspected that conscientious people would invest more time, because one of their common attributes is diligence. However, this was not the case: conscientiousness was negatively associated with creation time for all policies ( $B_{\text{emoji12-C}} = -3.58, B_{2\text{word12-C}} = -3.4, B_{3\text{class12-C}} = -2.09$ ). Neuroticism was significantly negatively associated with creation time, while agreeableness was marginally positively associated in 3class12. Demographic control factors indicated that, on average, women spent more time creating passwords in all conditions ( $W = 1166, p < 0.01$ ). The biggest effects were visible for task number as predictor: our participants took significantly less time to select passwords for the second and third tasks (example shown in 7.4). Most likely, this is a learning effect or due to people modifying their previous passwords.



**Figure 7.3:** Associations between personality traits and time to create a 3class12 password. All traits could be modeled as linear predictors.



**Figure 7.4:** Associations between control variables and time to create a 2word12 password. Especially the task number and gender were showed strong associations.

### 7.2.3 Finding Summary

Overall, the usability metrics were comparable for all policies and the ranking was fairly homogeneous. So, any deviance from the mean could potentially be explained by personality traits and other confounding factors. In terms of personality, the most important finding was that scores on the neuroticism dimension seem to be positively associated with the usability of the emoji policy. As indicated above, emojis in passwords appear to fulfill their purpose in that they help some people express their emotions. Participants who did not have a background in IT were much more likely to prefer the emoji policy, and their neuroticism score played a subordinate role. Perhaps, a lack of expertise does not allow those participants to consider the switch from keyboard to mouse and vice versa (*homing*). Moreover, the 3class12 policy received the most votes which might be due to the *status quo bias* or *familiarity bias*. 3class12 is the only commonplace policy among the tested ones, so participants were already used to it and voted it first. The timings were mostly affected by the task order – participants spent less time on the second and third creation task. So although personality traits serve as predictors for timings, these effects were weaker by comparison.

---

In summary, control factors had a greater effect on password usability and approval of policies than personality traits. Nonetheless, medium to strong effects are more likely to be detected with our sample size to begin with. So, the fact that we even observed small associations indicates that personality is a contributing factor to the perception of password policies. At this stage of the research we were not able to look into the selection behavior because the repeated measures design stood in the way. Thus, we designed and conducted two further studies to investigate the effect of personality on password strength perception and selection.

## 7.3 Study 2: Strength Perceptions

With an observational online study, we explored the associations between psychological variables and password strength perception. As outlined in Chapter 5, we regard the perception of strength as an implicit driver for behavior, which is more difficult to observe. Hence, we explored associations between subjective password strength ratings and scores on well-established psychometric scales. The task resembled the PASDJO game: participants were shown a password, and they had to rate it on a seven-point scale. We chose seven point scales to make the approach more comparable to Ur et al. [342]. Moreover, we gave participants two passwords, and they had to decide which one was stronger.

Before we ran the study, we pre-registered the experiment with the open-science framework (OSF)<sup>4</sup> and planned to conduct all analyses as predicted to mitigate confirmation bias. However, the envisioned statistical tests were not always applicable, which forced us to consider more appropriate methods and move away from the pre-registration. Since we consider our research efforts mostly exploratory, we approached the study without specific hypotheses regarding the influence of certain personality traits on the perception of password strength.

### 7.3.1 Structure

The study was divided into six parts. Two parts were standard psychometric tests. We describe all parts, to give the reader the full picture about the participants' tasks. However, we have to omit a few less important results for the sake of clarity. After a brief introduction where the participants were informed about the background of the study, the first step was to provide basic demographic information regarding gender, age, educational and professional background.

**Meta Password** The second part elicited characteristics about the passwords that our participants used on real online accounts. Here, we asked about typical password attributes, like LUDS (lower-, uppercase, digits, symbols), length and the inclusion of dictionary words.

---

<sup>4</sup> <https://osf.io/>, last accessed 11.09.2016

The collection of such password descriptions is an ethically reasonable way to study actual behavior that does not directly involve creating and disclosing an entire password [354]. Participants could select from a list of accounts that they used on a regular basis, e.g. Facebook, YouTube, Netflix, Google. If they did not have any of the selectable accounts they could provide another. We call the description of a participant’s password on any of these sites “*meta password*” in the remainder of the chapter. We included this part in the survey to explore additional covariates for strength perception.

**Table 7.3:** Set of passwords that we divided into different length and strength categories (as measured with zxcvbn). Features: U = uppercase, D = digits, S = symbols

Password	Categories		Features	guesses (log10)
	Length	Strength		
hagrqqqthhbbe	Long	Strong	-	12.48
etuhcarap	Short	Weak	-	4
AbWxCdYz	Short	Medium	U	8
1qaz2wsx3edc	Long	Weak	D	3
a6a4ba8a	Short	Medium	D	8
ieatkale88	Short	Medium	D	10
thedzfhg123	Short	Medium	D	10
11Nd1sPPut8ble99	Long	Strong	U,D	16
bicycles-peaches-cold	Long	Strong	S	13.69
AatIcs,ijayl-t	Long	Strong	U,S	13.95
p@ssw0rd	Short	Weak	D,S	0.95
ocean4 Size !beer Car	Long	Strong	U,D,S	20.39
F@m1Ly07%	Short	Medium	U,D,S	6.88

**Standalone Strength Rating** In the third part of the study, the *rating part*, the participants assessed the strength of one password at a time in random order, similar to playing PASDJO. The passwords had to be rated on 7-point scales ranging from 1 = *very weak* to 7 = *very strong*. We picked a set that was comparable to related work [342] and that we carefully designed around certain attributes. Table 7.3 shows the selected set of passwords and their features. The “length” category distinguishes between short passwords with nine characters or less and long passwords with ten characters or more. The distinction is inspired by real-world policies that most commonly require up to nine characters (see Chapter 6). The “strength” category groups passwords on three levels by their guess number as determined by zxcvbn. Weak passwords require less than  $10^6$  guessing attempts, strong passwords at least  $10^{12}$  guesses, and medium passwords anything in between. This classification is in concordance with related work [115, 369]. To fully counterbalance all category combinations, one would require 120 items, which we could not implement for the sake of brevity. Hence, we kept the number of items small to mitigate fatigue during the study sessions. In the chosen set of 13 passwords, there were at least three items for each distinct category, i.e. three short, long, weak and strong passwords, and at least three passwords pertaining to different LUDS policies (lowercase, uppercase, digits and symbols).





**Figure 7.5:** Simplified item of the comparison task. The passwords differ in length, strength, and the usage of uppercase letters and digits. Here would have scored the importance of length and digits with +2 while the importance of strength and uppercase letters was scored with -2.

**Comparison** Following a similar procedure as Ur et al. [342], participants moved on to compare the strength of passwords pairs (*comparision part*). The 7-point scale ranged from “<left password> is much stronger” to “<right password> is much stronger”. The pairs were constructed such that the passwords differed, e.g., in the existence and positions of digits and uppercase letters. Figure 7.5 illustrates our scoring schema.

If the passwords measurably differ in strength as in Figure 7.5, the ratings show if the participants’ perceptions match reality. In total, ten comparisons had to be made in random order, in which we permuted the combinations. For this task, we also added an attention check where the passwords on both sides of the scale matched, allowing us to exclude responses where the answer differed from “both passwords are equally strong”.

**SeBIS** Next, we requested self-assessment about security-related behavior using the Security Behavior Intentions Scale (SeBIS) [97]. This scale comprises the dimensions *securement*, *passwords*, *awareness*, and *updating*. For each dimension, a score is calculated with four items totaling up to 16 additional items in our study. However, in discussions after the experiment we received hints that it would have been better to include a gap of a couple of days before collecting the SeBIS data to ensure its validity. Doing so would mitigate priming and social desirability biases. Since we failed to take this into account beforehand, we do not report the results further, but it is important to mention that the questionnaire included those 16 items.

**Big Five** The study concluded with two psychometric tests. In the *Big-Five part*, we utilized a set of 50-items from the International Personality Item Pool (IPIP), which is a representation of Costa and McCrae’s NEO-PI-R domains [60]<sup>5</sup>. In this personality test, participants rate how accurately a certain statement portraying a certain personality characteristic describes themselves. Each item is a 5-point scale with the labels *very inaccurate*, *moderately inaccurate*, *neither accurate nor inaccurate*, *moderately accurate*, *very accurate*. Every personality trait is tested with five positively and five negatively keyed items. It was shown that the 50-item version of the test shows high correlation with more exhaustive tests ( $r > 0.75$  in all dimensions) and is thus a sufficiently reliable test. We randomized the order of the items.

<sup>5</sup> All items of the personality test can be found here: <http://ipip.ori.org/newNEODomainsKey.htm>, psychometric properties: [http://ipip.ori.org/newNEO\\_DomainsTable.htm](http://ipip.ori.org/newNEO_DomainsTable.htm)

**GDMS** Egelman and Peer found that the general decision-making style had higher predictive power than the Big-Five traits for privacy-related behavior [98]. Thus, we wanted to test the feasibility of both psychometric tests and finished the study with the *GDMS part*. This scale uses 25 positively keyed items to measure the five decision-making styles *rational*, *intuitive*, *dependent*, *avoidant* and *spontaneous*.

### 7.3.2 Quantitative Analysis

Since at least three passwords showed a certain characteristic, e.g. uppercase letters, we averaged the ratings for them accordingly and used them as dependent variables. Moreover, in the psychometric tests we accounted for negatively keyed items, i.e. those items that were phrased with negations like “*I don’t talk a lot*”. We inverted the ratings where necessary and afterwards calculated the sum of agreement levels for each dimension (*trait score*).

As in Section 7.2, we repeatedly fit a GAM to our data, i.e. a more flexible and interpretable form of regression<sup>6</sup>. Subjective password strength assessments, respectively comparisons, serve as dependent/response variables. We calculate one score per participant and password category by adding up the corresponding ratings. For instance, if they rated all eight passwords containing digits with seven points, their score for “G\_Digits” (Group of passwords with digits) is 56. We average participants’ ratings for models that require means instead of total scores.

Psychometric scores on all several sub-dimensions served as independent variables (covariates). We always control the regression models for gender, technical background and age to contrast effects. Wherever possible, we model covariates as linear, if the GAM indicates that smoothing is unnecessary. For this data-set, we also conducted principal component analysis followed by factor analysis. The resulting factors are then used to fit additional models for comparison. This allows us to evaluate the suitability of the Big Five inventory for our exploration.

### 7.3.3 Qualitative Analysis

To better understand the reasoning behind the ratings and comparisons, we also inquired how the respondents approached the rating task. They could enter free-text answers after all ratings were done. The answers were then coded independently by two members of the team. The first coding step was to find categories and propose the code book. Afterwards, the proposed codes were handed over to the second coder, who sorted answers into the categories and amended new ones where necessary. Interrater agreement between the two coders was satisfactory (78%, Krippendorff’s  $\alpha = 0.55$ ) and the final the code book could be created after discussing the discrepancies. We report how many participants mentioned a particular theme in their response regarding their rating strategies.

<sup>6</sup> <https://multithreaded.stitchfix.com/assets/files/gam.pdf> (last accessed 06.02.2018)

---

### 7.3.4 Recruitment

We utilized the online research platform Prolific<sup>7</sup> to administer our survey. Participants received \$2.65 upon successful completion, which took 20 minutes in average. This compensation level is suggested as part of the ethical reward guidelines on the platform. Only an English-speaking audience was eligible to participate. From the 178 people who started the survey, 104 finished it. To prevent low quality answers, we introduced an attention check during the comparison part of the experiment.

### 7.3.5 Ethical Considerations

There is no institutional review board for this kind of studies at our institution. However, we designed the questionnaire to respect the participants' privacy and did our best to minimize the level of disclosure of sensitive data. The metrics we collected to characterize the participants' passwords are most likely insufficient to reconstruct the passwords in a straightforward way and can thus be considered ethically acceptable.

### 7.3.6 Limitations

Like most studies involving personality assessment, the result is only a rough model of a person's personality and does not include all facets. We chose a test with 50 items to assess the Big-Five traits. While such psychometric tests exist with item counts between 10 [137] and 240 items [60], the 50-item version has high internal reliability and does not fatigue the respondents as much as more exhaustive tests. Additionally, with a sample size of 100 participants, power-analysis tells us, that only strong and medium interactions are likely to be found for with our regression models (cf. [303] or [108, p. 223]). At this stage of the exploration, however, this is what we aimed for. If we do find effects with such a small sample, then they must be large enough to justify follow-up investigations with larger samples. Moreover, statistical analyses can be done very differently. We traded off model complexity and interpretability to draw first conclusions. Thus, the reported associations can never be seen **causal effects**, because we would have to use different experimental setups, and carry out the experiments many times on larger samples. In the scope of our personality studies, we therefore provide pointers and possible explanations, but do not claim that the results are highly generalizable. This is especially important, because the sample stems from a technically savvy audience. Users registering for tools like Prolific or Amazon Mechanical Turk may also have stronger financial motivation to do so than the rest of the population [271].

Furthermore, the methodology relies on self-assessment and honest answers, which are difficult to control. We introduced an attention check to mitigate the problem, by asking people

---

<sup>7</sup> <https://prolific.ac>, accessed 01.09.2016

to compare two identical passwords. For the meta-password, we do not know whether it was created on a mobile or desktop device. Passwords created on mobiles are usually less complex than those created on desktops [232, 355].

Finally, the study set-up and procedure may also influence the interpretability of the outcome. We decided not to randomize the order of the question blocks to maintain full control over the general procedure. When we measure dependent variables, the order of questions is still randomized in the question groups. This way, the potential fatigue effects are the same for all participants at the different stages, while the important questions are in random order. Moreover, the items for password pairs were not fully counterbalanced on all levels to prevent fatigue when answering the entire questionnaire. A more exhaustive set of tested passwords would increase the generalizability.

### 7.3.7 Results

In this section, we first describe the participants and meta password characteristics before we proceed to the regression analyses. Since we created plenty of regression models, we only report those who showed notable associations – mostly for “Overall” and “Digits” in the rating part, and “Symbols” and “Digits” in the comparison part. We omit results from other psychometric measures (GDMS, SeBIS) for the sake of stringency. The final part of this section shows qualitative findings and a brief synthesis of the results.

#### Participants

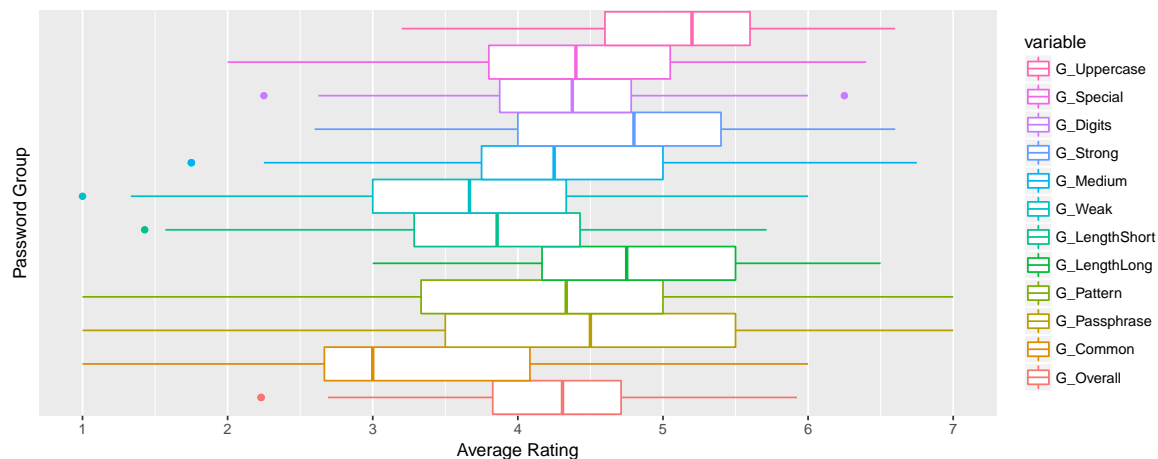
We collected 104 complete samples. We had to remove three samples from respondents who failed the attention check. Another response was removed because all responses on point scales were answered with the same value. This procedure is proposed by the IPIP project<sup>8</sup>. The resulting total  $N = 100$  was divided into 42 female and 58 male participants. Their average age was 28 years (*StandardDeviation* ( $SD$ ) = 9, *Minimum* = 16, *Maximum* = 61, *Median* ( $Md$ ) = 26). 44 responses came from students. The education level was high with 59 participants reportedly having a bachelor’s (44) or master’s (15) degree. 29 participants claimed to have a computer science or IT-related background. In summary, our sample stems from a young, educated and fairly technically savvy population. This convenience sample is not ideal, but we hope to deal with this skew by including demographics as predictors in the regression models.

#### Ratings Descriptives

On average, the respondents correctly identified weak, medium and strong passwords in the rating task, i.e. their perception matched reality. The average subjective scores were  $M = 3.60$  ( $SD = 1.07$ ) for weak,  $M = 4.25$  ( $SD = 0.95$ ) for medium and  $M = 4.71$  ( $SD = 0.90$ )

<sup>8</sup> <http://ipip.ori.org/newValidity.htm> accessed 02.09.2016

for strong passwords. A Friedman rank test showed that these ratings differed significantly ( $F(2) = 59.91$ ,  $p < 0.001$ ). Figure 7.6 shows all averages per group.



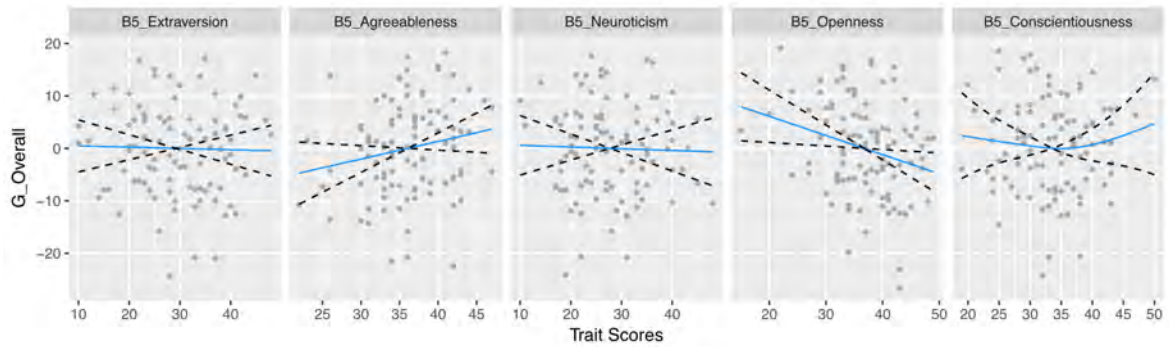
**Figure 7.6:** Participants’ average assessments of different password topologies, e.g. G\_Uppercase groups all passwords that contained uppercase letters.

## Standalone Strength Rating

Next, we analyze associations between personality and password perception. Internal consistency of the scale was fair (Cronbach’s  $\alpha = 0.72$ ).

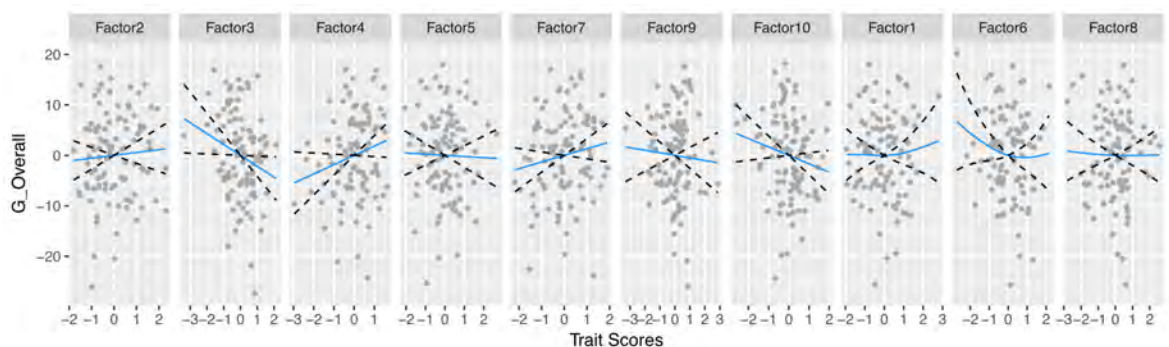
**Model 1: Big-Five Scores with minimal REML smoothing** For overall strength rating, most covariates revealed linear associations (Table 14.4 in the Appendix lists the coefficients). In Figure 7.7, we see that participants who scored higher on the *openness* trait generally judged passwords lower. This effect was flagged as significant in the model ( $B = -0.36$ ,  $\beta = -0.25$ ). For *agreeableness*, we can see a slightly more positive trend, i.e. participants tended to rate passwords higher, the higher they scored on the *agreeableness* scale. The other traits did not show any conclusive association. Having a computer-science background revealed slightly lower assessments ( $B = -2.92$ ,  $\beta = -0.14$ ), but the association was not significant. The model fit was rather low with  $R^2_{adj} = 0.06$  and an explained deviance of 14.7%. Associations and model fits were comparable for the “short password” and “weak password” categories. The highest model fit was achieved for the “Passphrase” category ( $R^2_{adj} = 0.17$ , explained deviance 26.3%), suggesting that passphrases were largely responsible for the overall strength rating model. In summary, personality did not explain much of the participants’ assessment. Penalizing smoothing parameters in stronger ways achieved higher model fits, however, the likelihood of overfitting the models increases. Therefore, we explored different factor constellations to get closer to explaining strength perceptions.

**Model 2: Extracted Factors as Predictors** Although the IPIP scale showed good internal consistency, we can try to break down the contributing factors. Usually, a *Principal Component Analysis* (PCA) reduces the number of factors, but does not have to. In our case, we



**Figure 7.7:** Visualization of generalized additive models for overall strength perceptions (i.e. tendency to judge a password stronger) with Big-Five traits as covariates. There is a significant negative association for openness: participants scoring higher on the openness trait judged password strength lower.

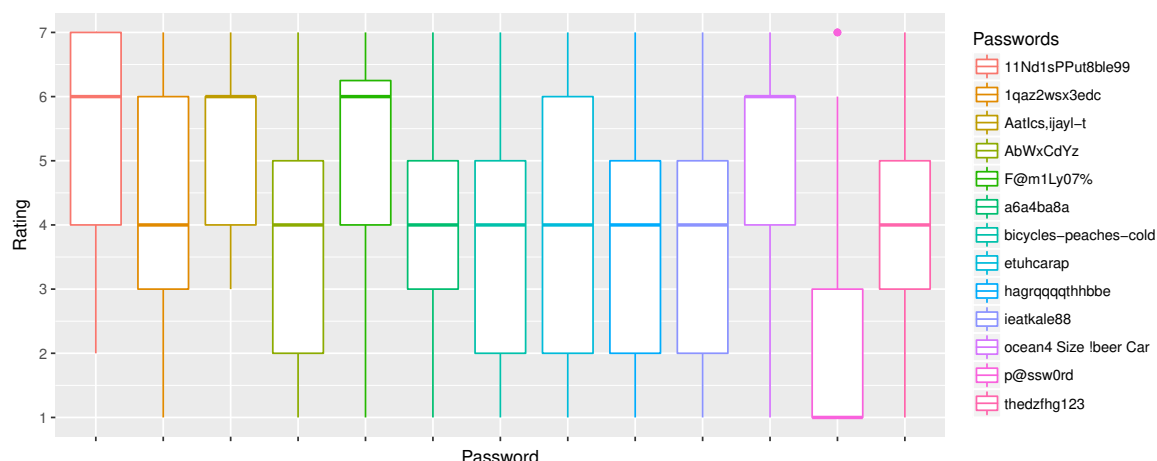
would expect five distinct factors from the 50 items, but a PCA revealed that there might be **ten** for our data set. We thus extracted those factors with a standard factor analysis using varimax rotation and used these factors as predictors instead of the big-five trait scores. The resulting model explained a larger portion of the deviance (20.1% vs. 14.7%) for overall assessment, but the R-square value remained constant. Coefficients for Factor 3 were biggest ( $B = -2.16, \beta = -0.21$ ), and all items of the openness sub-scale loaded onto it. To a much smaller degree, conscientiousness, agreeableness, and extraversion items also formed this factor. As a consequence, marginal effects, i.e. caused by only one personality trait, can be ruled out – it is a combination of trait scores that explains associations in our model. However, only a confirmatory study with a larger sample size can deliver final answers as to the specific trait combinations. As of now, we hypothesize that participants showing particular constellations of openness, conscientiousness and extraversion scores rated all passwords lower than other participants.



**Figure 7.8:** A principal component analysis suggested there might be 10 explanatory factors in our dataset for the personality construct, which we then extracted using varimax rotation. Factor 3, which was mostly loaded with *openness* items, Factor 9 (mostly *conscientiousness*), and Factor 10 (mostly *extraversion*) were associated with lower ratings.



**Model 3: Mixed Model – Password Characteristics and Big Five Traits** Similar to the evaluation of PASDJO strength ratings, we can take the different features of the passwords as covariates, e.g. the number of digits or the total length. Figure 7.9 visualizes the ratings for each password. We see that strength ratings take a broad score spectrum in many cases, with the exception of those passwords that contained symbols. Using password features as covariates allows us to identify their weight in the assessment.



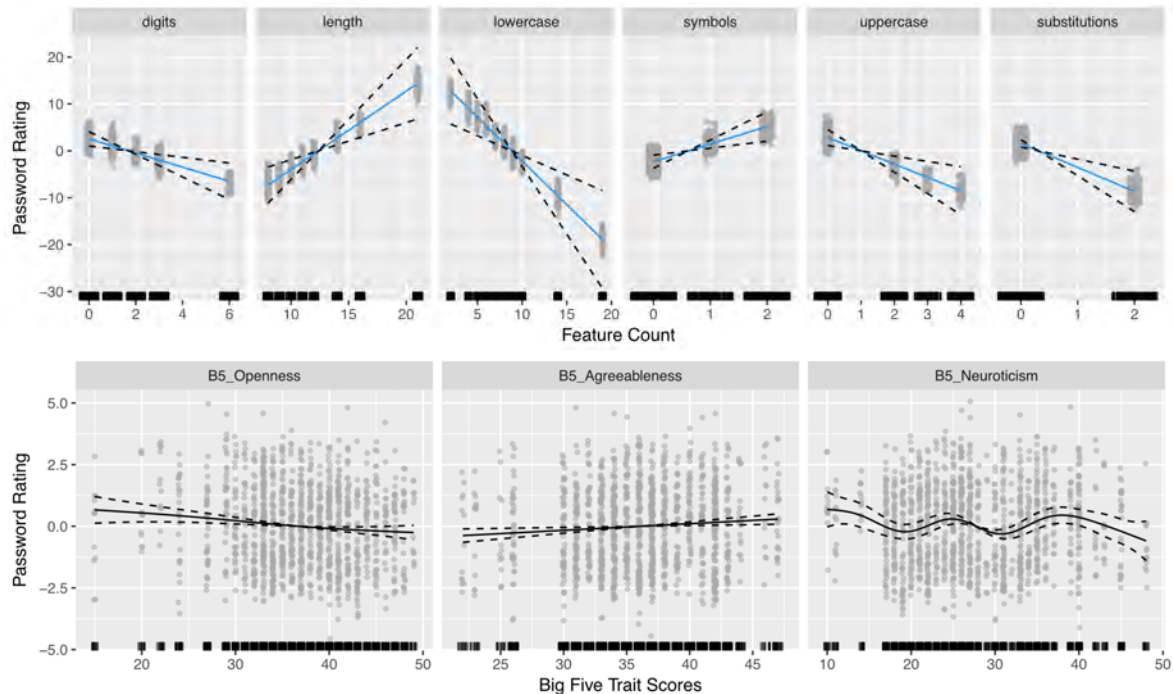
**Figure 7.9:** Participants' subjective strength assessments of the 13 passwords in the study. Broad interquartile ranges for indicate that participants largely disagreed on the strength.

First, we look at marginal associations between ratings and the number of lowercase, uppercase, digits, and symbols (LUDS metric), which is basically a linear model. All coefficients are positive ( $\beta_L = 0.15$ ,  $\beta_U = 0.24$ ,  $\beta_D = 0.26$ ,  $\beta_S = 0.13$ ) but weakly associated, and the model fit is rather low ( $R^2_{adj} = 0.12$ ). Mixing it with the Big-Five traits as predictors improves the fit slightly ( $R^2_{adj} = 0.15$ ). If we factor in *password length* as an interaction term with the LUDS metrics, the fit improves further ( $R^2_{adj} = 0.20$ ). The correlation between the number of digits and ratings becomes strong ( $\beta_L = 0.84$ ), i.e. more digits are perceived as stronger. This, however, was surprisingly contradicted when we account for the fact that two passwords had digits as substitutions for letters (*p@ssw0rd* and *F@m1Ly07%*). Thus, the final model includes an interaction term *digits \* substitutions \* symbols* as covariate and achieves the best model fit overall ( $R^2_{adj} = 0.25$ ). In Figure 7.10 we can see the resulting associations. Password length was the primary factor, while the influence of character substitutions reversed the positive impression of symbols, digits and uppercase characters. Personality traits on the other side, although statistically significant, were weak predictors. However, removing them from the model entirely would have led to a notable decrease in model fit ( $\Delta = 0.05$ ).

**Conclusion** The mixed model is strongly influenced by interaction terms with predictable character substitutions. In general, more digits, uppercase letters, and symbols led to higher strength ratings, but this association was strongly reversed if digits or symbols acted as



133t substitutions. Thus, we conclude that respondents were skeptical about this password creation strategy in the rating part. By and large, personality was a minor factor.



**Figure 7.10:** Top: Longer passwords were perceived as stronger. Correlations turned negative if, substitutions were factored into the model. Bottom: Personality traits showed very small associations.

## Standalone ratings modeled with Meta password

Trying to understand if participants' past behavior might have influenced their judgment, we created GAMs analogously with metrics of their *meta password*. The only notable association, which was conclusive across the board, was the reported number of uppercase characters. We found that with increasing usage of uppercase letters, participants tended to give lower ratings ( $\beta = -0.27$ ).

## Comparisons Between Two Passwords

We explored whether personality traits influence how participants decide between two given passwords. We modeled the comparison on the 7-point scale such that 1 represents a vote against a feature and 7 for a feature, e.g. more digits. Only conscientiousness was retained as predictor in all models. The most interesting association was found when one of the two passwords contained digits: Choosing the password with more digits positively correlated with higher conscientiousness scores ( $\beta = 0.43$ ). The opposite is true for participants scoring high on the openness scale. They are more likely to vote for the longer password (non linearly) than for the one containing digits ( $\beta = -0.17$ ). Totaling up all character classes

---

used in a password, we see that the more diverse it was, the more likely it was favored by participants with high conscientiousness scores ( $\beta = 0.36$ ). In this particular model, there were weak associations between gender and character diversity. Male participants were more likely to prefer the password consisting of more character classes ( $\beta = 0.23$ ). Having a computer-science background correlated with preference for the longer password ( $\beta = 0.21$ ).

In all the models, the predictive power was moderate and did not reach the levels from Model 3 in the rating task. However, the correlation coefficients were stronger for personality trait. We find that the decision between two given passwords is mostly associated with the conscientiousness and openness traits.

## Qualitative Findings

Entering an elaborate response to justify judgments was not mandatory. Yet, all but one participant ( $n=99$ ) gave a brief and in most cases comprehensible explanation for their ratings. The following numbers do not necessarily add up to  $n$ , because an answer could contain multiple codes.

We identified four overall themes in how the participants approached rating passwords: *Character diversity*, *creation strategy*, *predictability* and *other*. The character diversity code consists of participants mentioning the importance of symbols (69), digits (52), upper-/lowercase letters (45) and general variety of characters (16). Regarding the creation strategy, many participants penalized passwords when they contained actual words (40) or personal information (3). The predictability category was divided into answers referring to character substitutions (10), patterns (17), guessability (12), randomness (20), length (25) and the position of symbols/digits (6). The other themes were established from 8 participants using technical jargon (e.g. “attack” or “brute force”) and those who identified the obfuscated passwords (2). These themes echo the quantitative ratings very well.

### 7.3.8 Finding Summary

We found that participants evaluate password strength by looking for specific patterns. Regression models and qualitative analysis show that respondents mostly penalized lack of diversity and randomness which is consistent with related work [342]. Thus, the associations originating from different personality traits were small in many cases, but not negligible. Technical background and gender played a role in the comparison task, because male participants were more likely influenced by character diversity. On aggregate, the predictive power of the independent variables was higher in the comparison task than in the standalone rating. Nonetheless, a fine-grained mixed model revealed interesting side effects for standalone ratings. These include the reversal of the correlation if a digit or symbol was used to substitute a letter. The most important personality factors across both tasks were **openness**, **agreeableness** and **conscientiousness**. The sample size might have been too small

to yield narrow confidence intervals and impressive model fits. As exploratory stage, however, the data provides considerable evidence as to the feasibility of investigating “password personality”.

## 7.4 Study 3: Password Selection and Coping

As a final step in our “password personality” exploration, we ran another online study. Having investigated policies and the perception of passwords through the lens of personality traits, the main goal of the third study was to evaluate potential associations between personality and password *selection*. To overcome some of the limitations of the previous study, we hoped to increase the sample size and reduce the number of items during the study. Moreover, further answers about the participants’ explanations and motivations were considered to better understand the weight of personality factors. We determined the following research questions: 1) Are there correlations between password features (topology) and personality traits? 2) Do certain facets of personality shine through in password coping strategies, e.g. the tendency to write down passwords?

### 7.4.1 Procedure and Tools

The study was designed to take no more than ten minutes to lower the friction for participation. The briefing page informed participants about the purpose of the study and data disclosure policies. After acknowledging the conditions of participation, respondents were asked to create a password. To boost ecological validity, we provided a fictitious but realistic scenario [200]. The task was to come up with a new password for a new email account that they were going to use as their main address. Further, the instruction pointed out that the incentive would only be paid of if the participants chose a password they could recall later on. A *basic8* policy was enforced, as it is one of the most representative policies in the wild (see Chapter 6). This loose policy would also allow for both very complex and rather simple passwords, which could be associated with personality traits. Having successfully confirmed the password, respondents were taken to a questionnaire about demographics, just like in the first two studies.

Next, participants completed the BFI-K questionnaire consisting of 21 items that have to be rated on a 5-point scale. We opted not to use the 50-item inventory for the sake of saving time. We added an item that served as an attention check. It asked to respond to this item with “disagree”. Failure to follow this instruction allowed us to drop the response from the dataset. The resulting 22 items were shuffled to avoid sequence effects.

Afterwards, we surveyed respondents about their password management behaviors and preferences. We used multiple-choice and open responses to collect qualitative, self-reported data. For instance, we wanted to know how they cope with multiple accounts or how they

---

reuse passwords. Since the study was done as part of a bachelor thesis project, we were limited by the amount of time available. Therefore, we deemed the self-assessment sufficient to gauge security behavior rather than taking more time to administer a SeBIS questionnaire. The survey concluded with a recall task, where participants provided their initially chosen password. They could try as often as they liked, and the number of attempts were recorded. In case they were unable to recall their password, they could proceed anyhow and take part in the lottery. If they chose to provide an email address in the final step, this data was stored separately from the questionnaire data to avoid privacy issues.

## 7.4.2 Recruitment and Sample

Participants were invited via a university newsletter, and snowballing the link via personal connections and posts on social networks. The questionnaire was in German and participants were screened about their command of the German language. We instructed participants to take the survey on a desktop. 184 people completed the survey, but we had to drop the responses of 8 participants because their response timings were unrealistically fast (lower than  $2 \cdot \text{standard deviation}$ ). From the 176 remaining respondents, 89 were male, 86 female and 1 preferred not to answer. 116 were students, i.e. a rather high proportion (66%). Consequently, the average age was 25 years (range [16;55],  $SD = 6$ ,  $Med = 24$ ). 67 (38%) reportedly had an IT-background. 129 respondents chose to participate in the raffle for shopping vouchers.

## 7.4.3 Limitations

Like most password-selection studies, our study is limited by the purposefulness of the task: Participants knew they were only going to use the password in the study. To mitigate this, we tried to give participants enough context information to immerse themselves into the situation. Moreover, while the study was ongoing, we received 30 feedback emails asking for clarification about the attention check questions. We had introduced an additional item at a random position in the personality construct. This led to misunderstandings that we failed to identify in the design and piloting of the survey. Some respondents thought that this question was a measure of their personality, too, and indicated giving a wrong answer on purpose. We therefore had to omit this sanity check entirely, because it did not feasibly tell whether participants had read the questions carefully enough. Instead, we based the decisions to drop responses on the timings.

## 7.4.4 Results

The resulting 176 passwords had a median-length of 10 characters (range [8;22]) – 130 participants went beyond the minimum requirement of eight characters. Figure 7.11 visualizes

additional metrics that show that passwords were also stronger than expected with a mean guess number greater than  $10^8$ . Passwords with guess numbers greater than  $10^6$  are expected to withstand online attacks [115]. Moreover, overall internal consistency of the big-five construct was at  $\alpha = 0.65$ , thus slightly below the bar at 0.7. Subscales for each trait, however, were more consistent and above the threshold. In the following, we try to fit generalized additive models to the data using B5 trait scores as covariates.

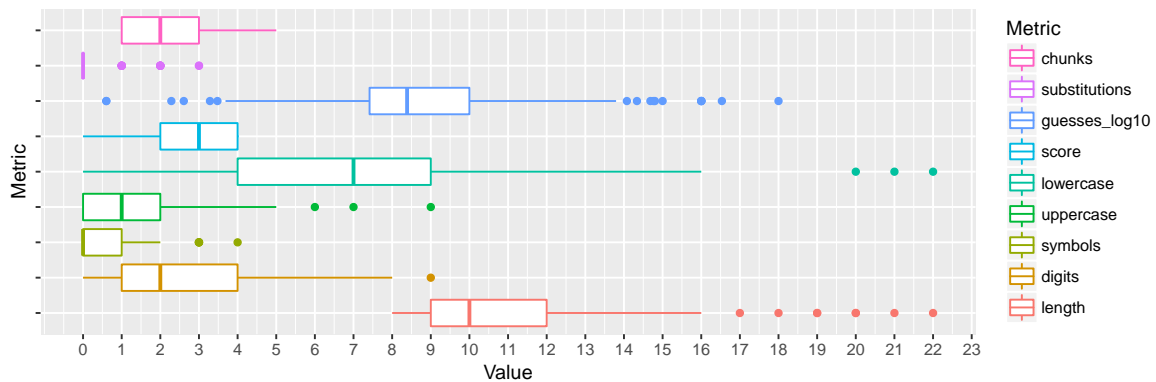


Figure 7.11: Zxcvbn metrics for user selected passwords.

## Password Composition

First, we use zxcvbn metrics as response variables (see Figure 7.11), and explore marginal associations with big five traits. As before, we include age, gender and IT-background as control covariates. We found that only password length was (significantly) associated with B5-traits: Participants with higher *neuroticism* scores tended to create longer passwords ( $\beta = 0.21$ ), using more lower-case letters as a side effect ( $\beta = 0.19$ ). A second corollary was that passwords from those participants also consisted of more word-chunks ( $\beta = 0.26$ ). However, it is hard to read this result, because even very long random passwords consist of only one matching sequence (“bruteforce”), so more chunks does not imply greater strength. Moreover, having an IT background was positively associated with password length ( $\beta = 0.21$ ). A one-tailed Mann-Whitney test confirms this ( $W = 4204, p < 0.05$ ). Consequently, password guess numbers were higher if participants had an IT background ( $\beta = 0.19$ ). None of the models however, achieved exceptionally good fit: the maximum R-squared (adjusted) value was  $R^2_{adj} = 0.13$  (for the number of symbols as target variable). We tried to achieve a better fit by performing a principal component and factor analyses. This was not effective, either. We conclude that password metrics are only very slightly predictable by personality profiles.

## Password Categories

We manually coded all passwords in two stages: first we tagged passwords with a creation strategy and afterwards we combined them to broader categories. The resulting codes

---

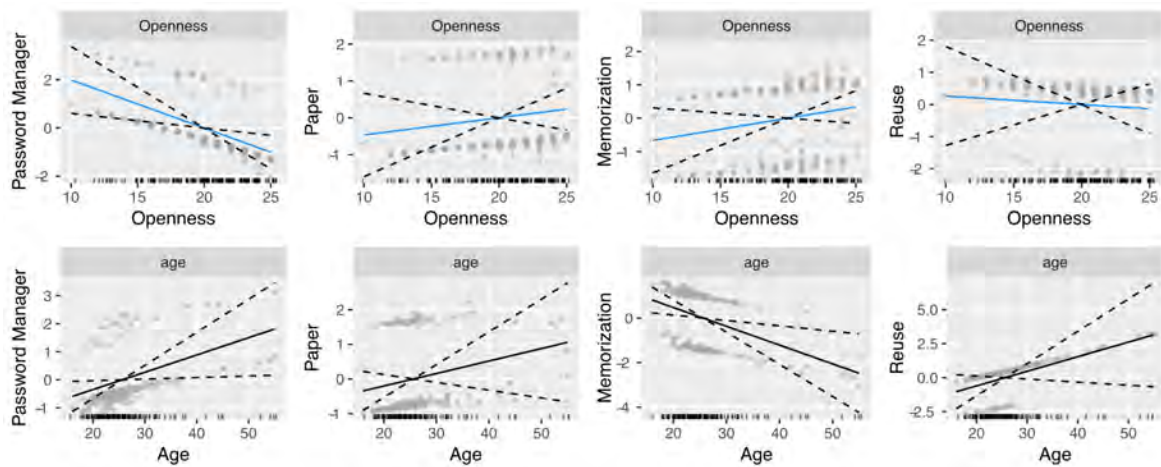
for passwords and their respective frequencies were: simple (17), systematic (7), random (16), mnemonic (8), passphrase (25), common-topology (56), and hardened-topology (47). *Simple* passwords can be found in dictionaries, consist of only one word, only lowercase letters or only digits. *Systematic* passwords are assembled with keyboard patterns. We did not see any coherent approach in *random* passwords. *Mnemonic* passwords have upper- and lower-case letters, digits and punctuation at predictable positions that indicate a sentence-like structure as origin. *Passphrases* are combinations of dictionary words with or without separators. The *common-topology* code describes passwords with more than one character class that show typical password structures like “hello123” or “Banana1!”. Finally, *hardened-topology* passwords use character substitutions and pseudo-random capitalization of letters, but clearly come from a common-topology password.

As before, we created logistic additive models to fit personality to a password category. The only notable associations we found was a stronger tendency to create a passphrase if participants had a background in IT ( $\beta = 1.97$ ), while female participants were less likely to create a random password ( $\beta = -2.8$ ). Although some (smoothed) factors from the personality dimensions were significant, they were not conclusively associated with the type of created password.

## Password Management Strategies

Respondents freely reported reusing passwords (88.4%), and 65.34% said to categorize their passwords. Memorizing passwords was the preferred strategy for 109 participants, followed by notes on paper or a file ( $n = 43$ ). Using a password manager was generally the least favored option ( $n = 24$ ).

We created generalized additive logit models for every management strategy, i.e. binary outcomes like “Using a password manager”: 1 (true) or 0 (false). Since the percentage of participants reusing passwords is so high, influence by personality traits was unlikely to be detected. Consequently, none of the associations was flagged as significant for reuse. However, older participants were more likely to reuse passwords as shown in Figure 7.12. Extroverted participants were significantly less likely to write passwords down on paper or into a digital file ( $\beta = -1.54$ ). Participants with an IT background showed fewer signs of password reuse ( $\beta = -1.97$ ). Instead, they more likely used a password manager ( $\beta = 2.1$ ). Since having an IT background was significantly correlated with being male ( $\chi^2 = 14.32, p < 0.001$ ), males were also more inclined to use a PWM. We can observe another interesting aspect in Figure 7.12 which depicts associations between all tested coping strategies and openness scores, respectively age. Age and openness show an *inverse* tendency towards using a password manager: while older participants seem to be more likely to use a PWM ( $\beta = 0.77$ ), the opposite is the case for participants with higher openness scores ( $\beta = -2.06$ ). Similarly, memorizing passwords was less likely for older, but more likely for “open” participants. Generally, memorizing passwords seems to be the **only** unfavorable coping strategy for older participants.



**Figure 7.12:** Age and Openness model plots for all coping tested coping strategies. Openness and Age show inverse associations regarding password managers and memorization.

### Memorability

Only 18 out of 176 participants (10%) failed to recall the newly created password at the end of the survey. Thus, associations with personality traits were unlikely. However, it was clear even with only 18 passwords being forgotten that the length of the password significantly contributed to failure rates ( $\beta = 2.25$ ,  $R^2_{adj} = 0.19$ , 25% of explained deviance). The R-squared value drops significantly if we remove the Big-Five traits as predictors ( $R^2_{adj} = 0.07$ , 8% of explained deviance,  $p < 0.05$ ). Thus, password memorability was associated with personality, but we are not able to tell exactly in which way.

### 7.4.5 Summary and Interpretation

We observed that password length was marginally associated with neuroticism, but none of the other personality traits. Higher neuroticism scores can be paraphrased as lower emotional stability. Losing access to an account might lead to stronger emotional reactions. So, possibly, participants showing this trait want to make sure that this scenario does not happen by choosing a longer and stronger password. Participants working in or studying an IT-related field, were also more likely to pick longer passwords, but achieved significantly higher guess numbers. This corroborates findings from Mazurek et al. conclusively [226]. Fitting generalized additive models to data of 176 participants did not show sufficiently strong model fits to warrant definite inferences about the influence of personality traits on password metrics. To a higher degree of certainty, however, we can state that personality must not be ruled out for password selection, which is an unprecedented result. This finding was particularly evident when we dropped the big five scores as predictors for memorizing the password.

Regarding the retrospective data, we also found salient associations. Although the overall sample shows that using a password manager is unattractive, a deeper analysis revealed that



**Table 7.4:** Overview of significant associations between Big-Five traits, control factors, and different metrics across three studies. Arrows indicate direction of the associations, colors also highlight significance levels (red/orcher:  $p < 0.05/0.1$  negative assoc., greens:  $p < 0.001/0.01/0.05/0.1$  positive assoc.) Openness and Conscientiousness are the most important factors from the Big-Five model, and having an IT background was an indispensable control factor.

Metric	Type	Study	O	C	E	A	N	Age	Gender	IT
Difficulty to create 2word12 PW	Usability	1							➡ (< .1)	
Difficulty to create 1emoji12 PW	Usability	1				➡ (< .1)	➡ (< .05)		➡ (< .1)	
Preference of 3class12 policy	Attitude	1				➡ (< .05)				
Preference of 2word12 policy	Attitude	1			➡ (< .05)					
Preference of 1emoji12 policy	Attitude	1				➡ (< .05)	➡ (< .1)			➡ (< .05)
Time to create PW	Usability	1		➡ (< .05)			➡ (< .1)		➡ (< .01)	
Overall tendency to judge PWs	Behavior	2	➡ (< .05)							
Comparing PWs based on complexity	Behavior	2		➡ (< .01)					➡ (< .05)	
Comparing PWs based on digits	Behavior	2	➡ (< .05)	➡ (< .001)						
Comparing PWs based on uppercase	Behavior	2		➡ (< .1)				➡ (< .1)		
Comparing PWs based on length	Behavior	2	➡ (< .05)	➡ (< .05)						➡ (< .05)
Length of created PW	Behavior	3					➡ (< .05)			➡ (< .05)
Guess number of created PW	Behavior	3								➡ (< .05)
zxcvbn score of created PW	Behavior	3					➡ (< .1)			➡ (< .05)
Created PW is passphrase	Behavior	3								➡ (< .05)
Created PW is random	Behavior	3							➡ (< .05)	➡ (< .1)
Cope by memorizing PW	Behavior	3		➡ (< .1)	➡ (< .05)	➡ (< .1)		➡ (< .1)	➡ (< .05)	➡ (< .05)
Cope by reusing PW	Behavior	3				➡ (< .1)			➡ (< .05)	➡ (< .05)
Cope by using PWM	Behavior	3	➡ (< .05)						➡ (< .1)	➡ (< .05)
Cope by using paper / files	Behavior	3			➡ (< .05)					
Number of significant associations ( $p < 0.05$ [0.1] )			4 [4]	4 [6]	3 [3]	2 [5]	2 [5]	0 [2]	3 [6]	9 [9]

the participants' age and openness scores are linked to using a PWM, or to memorization efforts. These predictors were inversely associated, which means that there could be a “sweet spot” of openness scores and age. Consequently, this gives us the opportunity to segment target groups for password managers more effectively: Older users might be more receptive for password support tools. Lower openness scores indicate that people are more conservative, “down to earth”, and appreciate conventions [229]. These values also appear plausible for older participants in the light of related research [310]. Thus, we conclude that the design of future password support tools should be centered on different age groups. We will also see in Chapters 8 and 12 that none of the current solutions are specifically tailored to different user groups. Current one-size-fits-all solutions might hence explain to some degree why PWMs are still underused.

## 7.5 Discussion and Implications

### 7.5.1 Overarching Themes and User Segments

Egelman and Peer highlighted the importance of the the question *which psychographic segments should be targeted [in security and privacy mitigations]*? While they focused on their newly developed SeBIS scale and other psychometric constructs, we are able to give new pointers for segmenting users based on their Big-Five traits in conjunction with demographic factors. To approach the segmentation, we can look at the overall influences

of personality and demographic factors on different metrics. Table 7.4 lists all significant marginal associations from all three studies.

**Neuroticism** In study 1, the primary observation was that neuroticism was associated with difficulty to create an emoji password. Neuroticism was not associated with any perception metric in study 2, but the third study revealed an interesting association between neuroticism and self-selected passwords. Metrics in the first and the third study revolve around password *creation*: It appears to be both **easier** to find a password expressing emotions for participants scoring high on neuroticism, and their passwords turned out significantly **longer** in the third study. Thus, targeting neuroticism with persuasive interventions during password creation might boost these positive associations. Nudges should thus focus on making emotional state more *salient* and point out benefits of password length to *positively reinforce* this behavior.

**Openness** The usability of different composition policies was not associated with openness, but the perception of password strength showed conclusive associations in that passwords were generally judged **weaker** with higher openness scores. Participants strongly showing the openness trait were also more likely to base their decision on **password length** rather than the number of digits. In study 3, we observed that coping strategies involving a password manager were **less likely** to be found with participants with high openness scores. The significant associations, and absence thereof, tell a conclusive story if we look at passphrases and mnemonic phrase-based passwords. Those types of passwords are strong and often not overly complex [189, 206, 297]. Passphrases, for instance, are a technique to facilitate memorization. They also easily exceed length requirements. If they consist of uppercase and lowercase letters, and are separated with regular punctuation symbols, there is no need to add digits to meet a three-class requirement: the passphrase is already complex enough. All this is visible in the study behavior of participants scoring high on openness, despite the absence of associations between openness and the type of created password in study 3.

**Conscientiousness** Like openness, conscientiousness was a major factor in study 2. There was evidence that participants, who strongly show this trait, tend to believe a more **complex** password is better than a long password. We could explain this finding by looking at the attributes that are usually found with conscientious people: diligence and neatness. Following password rules requires these facets to ensure a strong outcome. The results in study 1 (less time taken to create a password) and study 3 (slight tendency to refrain from memorizing passwords) are harder to interpret. We could expect that crafting a strong memorable password takes due diligence and more time, thus the shorter time spent by conscientious participants is counterintuitive. Perhaps it is due to the measurement approach in our study. We started taking the time as soon as the password field was focused and the timing ended when the field lost focus (“blur” event). It is possible that conscientious participants took their time to reflect on what they wanted to enter into the field, and only then started the task. However, we do not have the data to support this argument.

---

**Agreeableness and Extraversion** Table 7.4 shows that agreeableness only showed a conclusive result in study 1. A more demanding policy like 3class12 did not appeal to participants scoring high on agreeableness, while they did favor the emoji-policy much more. Cheerfulness, empathy and cooperation are often characteristics represented by this trait. Participants might have favored cheerful emojis to cooperate on finding a suitable solution together. Thus, password selection could have been more fun and thus more pleasing for those participants. Since there was no sign of focusing on password strength, we hypothesize that memorability might be more important for them. Regarding extraversion, we found that participants scoring high on this trait disliked a word-based policy and were more likely to memorize passwords than writing them down. This is probably the most difficult result to interpret, because extraversion is a more situationally dependent trait than the four others – a phenomenon coined as “ambiversion” [140]. This could have become visible if our regression models had shown more curvilinear relationships, but those were not significantly more likely for extraversion than for other traits. We therefore refrain from further discussion and note that additional data would be necessary to evaluate the stability of this predictor.

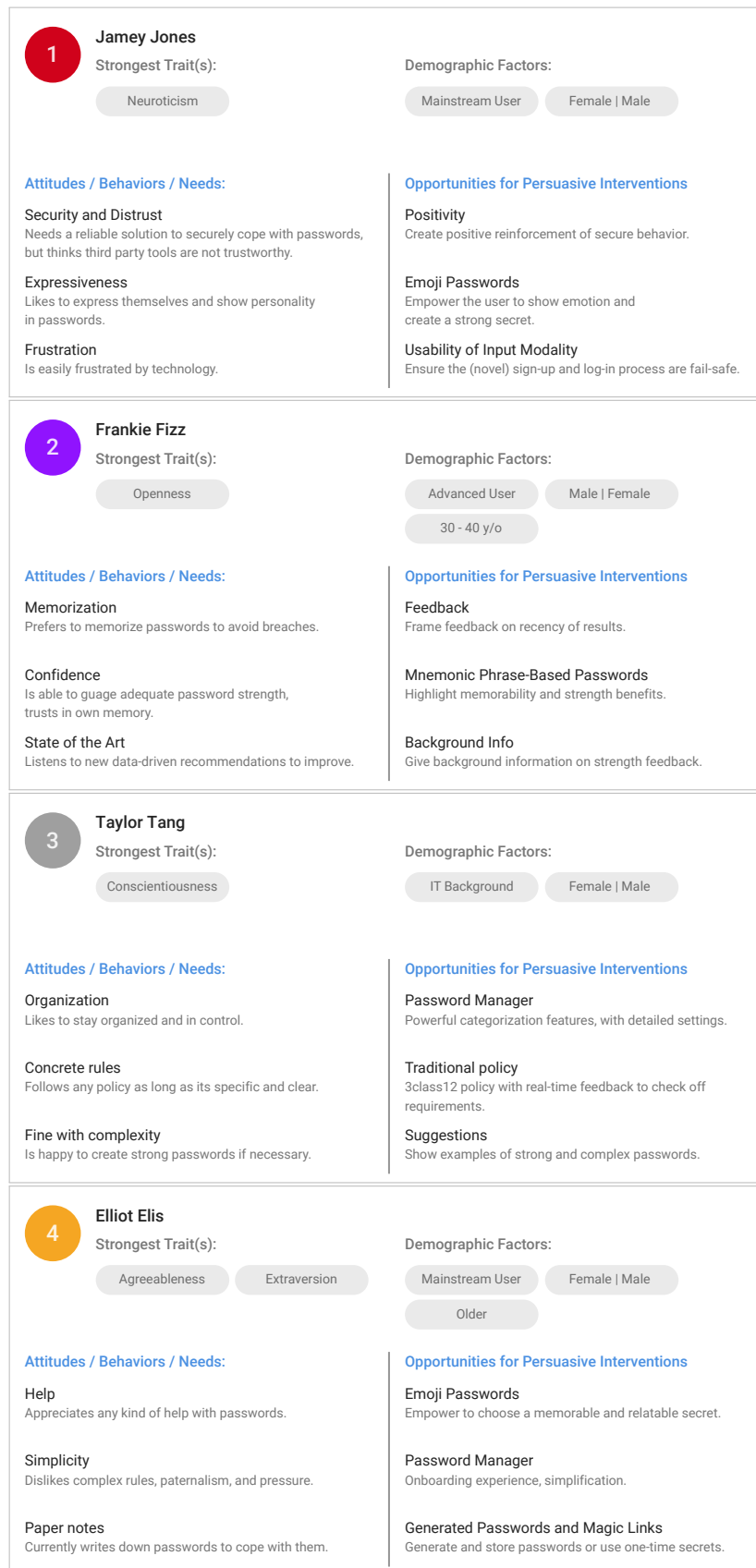
**Demographic Factors** From Table 7.4, it is evident that associations with the chosen metrics are more likely to be found with demographic variables as predictors. This was especially true for study 3, where almost all outcomes were associated with having experience or education in a computer-science related field. This is an unsurprising finding, because this user group can better judge the implications of their behavior. Nevertheless, the additional factors help us segment user groups in higher detail.

### Deriving Segments: Password Personas

With the above themes and stories, we are able to derive a number of user segments that can be targeted by security mitigations. Segmentation can inform upcoming research and design directions. For instance, segments serve the generation of hypotheses and play a role in the creation of new psychographic constructs. The “securing” dimension of the SeBIS might be enriched by user archetypes to *explain* attitudes and behaviors rather than just measuring them. Here, personas constitute a common design tool for segmentation. These fictional users can be targeted in the design of persuasive interventions. From the data of three studies on password personality, and backed up by related work, I created a set of four “password personas” that inform design choices in Part III of this thesis. They are shown in Figure 7.13. At this point, the personas still remain abstract to account for the early stage of research about password personality. Personas should be “living” templates that are updated based on new data [138].

## 7.5.2 Designing Feedback and Suggestions

Using our personas, it is possible to address personality facets in real-time feedback during password creation. In the wild, we encounter password meters that estimate the strength of



**Figure 7.13: Password Personas.** These fictional user profiles can inform design choices for password support strategies. Refining the details needs further research on password personality.

---

the password and in some cases even provide verbal feedback about what the user can do to improve the password (see [48, 369]). A simple approach tries to convince the user to pick a stronger password by **suggesting a new one** or a modified version of the entered password [126, 291, 300, 341].

Coming back to the finding that personality had a notable effect on how participants *compared* two passwords, we suppose that it plays a role for real-time suggestions, too. How a modified version or newly generated password is received likely depends on the user's personality. On mobile devices, the user's password might be visible in clear text while they enter it [232]. Displaying an alternative password then resembles the comparison task from our study, and they might wonder which one is stronger. Strength feedback facilitates answering this question. It seems to be easier for *conscientious* people to assess the strength of their password, if there is a clear list of requirements than can be "checked off", to make it superior to the suggested password (see the persona 3 "Taylor Tang" in Figure 7.13). At the same time, users who strongly show the *openness* trait seem to benefit from data-driven background information about the strength estimation algorithm. We conclude that Ur et al.'s meter might be especially helpful for the persona 2 (Frankie Fizz). To make it work for persona 1 (Jamey Jones), it needs to be more reassuring than it currently is: The version presented at CHI 2017 constructively criticizes the user's password and suggests alterations. Adding a positive message might make persona 1 more receptive for this kind of suggestions.

Moreover, we found that demographic factors are useful to segment audiences. We found that older participants in our third study were more likely to appreciate if they do not need to memorize passwords, which was considered in persona 4 (Elliot Elis). They were more likely to use a password manager, paper notes or reusing passwords to reduce cognitive efforts. Thus, there is a great opportunity to suggest a coping strategy instead of a stronger password. For instance, websites or browsers could extend sign-up forms with a prompt to use a password manager. Pointing out the value of not having to memorize the newly created password could speak to older users and drive adoption rates. It is very important that the user journey from sign-up to password-manager set-up is simple enough to meet persona 4's needs. During account creation, interventions can leverage the "opportune moment": *Kairos*, one of the persuasion principles presented in Section 3.4.2, likely possesses a higher nudging power than, e.g. a news article recommending a password manager, because users are experiencing the problem first-hand during account creation.

### 7.5.3 Assessing Personality Traits in Password Studies

In our studies, we *explicitly* measured personality through psychometric constructs. This was straightforward, because online studies have become a reliable go-to method to study passwords. However, in many cases, omnibus tests like ANOVAs fail to reveal significant effects, and confounding variables could blur causality. Only few such covariates have been considered beyond demographic information. Our studies highlight that personality is a promising candidate to consider: for instance, average password strength perceptions were

distributed normally. However, using general models, the underlying associations became visible and were statistically significant. Thus, psychometrics should not be neglected, so we can boost efficacy of security mitigations for different user segments [98].

Still, extending surveys with long psychometric constructs is probably unrealistic. In our studies we used 50 and 21-item constructs. The latter already showed slightly reduced internal consistency. Thus, to keep studies short, to prevent participant fatigue, and to obtain reliable data, we propose focusing on the **openness** and **neuroticism** personality traits. Their associations were most stable in our studies and provided a coherent picture. They can be measured with only a couple of items (between 9 and 20), so the corresponding study part is finished within a few minutes. Nevertheless, the specific choice of the traits should be backed up by confirmatory studies in the future, and may also depend on the research question.

Adding even a few more items to a questionnaire might be impossible due to study constraints like budget or timeliness of data elicitation. There are, however, promising solutions to quickly and inexpensively obtain personality data on all dimensions from a user's past behavior. It is possible to infer personality facets from **social media data**, e.g. public interactions on Facebook [386]. Youyou et al. found that such metrics can even outperform psychometric questionnaires. Stachl et al., as well as De Montjoye et al. found conclusive associations between **smartphone usage** and personality traits [80, 311]. We thus suggest requesting permission to read either smartphone or social network data as part of the study. Crowd-workers in a large-scale study by Bentley and Chen, for instance, showed little concern to install software on smartphones for research purposes [20]. While we would not go so far as to read all contacts, call and message history, we believe a differentially private<sup>9</sup> approach can replace personality constructs.

#### 7.5.4 Statistical Models

After consultation with statisticians, we opted for the use of generalized additive models (GAMs). Their usefulness hinges on the interpretability of their corresponding regression plots. Traditional indicators like (un-)standardized coefficients are limited in their contribution to gauge marginal associations and must be interpreted with special caution. In the case of logistic regression, standardizing binary variables is not useful, but we reported beta values to describe the slopes in the plots. Especially in study 2, we tried to select the model with the highest goodness of fit as indicated by the adjusted R-square value. We also considered the models' Akaike information criterion (AIC) to move forward. Here, increases and decreases of the AIC matched those of the R-square value. Interestingly, the models achieved better fit than related work on personality in privacy behavior [98]. Thus, we advocate the use of GAMs in future analyses.

<sup>9</sup> [https://en.wikipedia.org/wiki/Differentially\\_privacy](https://en.wikipedia.org/wiki/Differentially_privacy) (last accessed 04.03.2018)

---

## 7.6 Conclusion and Future Work

In three studies with a total of 440 participants, we broadly explored associations between personality traits and password behavior. This effort was one of the first of its kind. We focused on the relationship between the Big-Five model and password policies, strength perceptions, password selection, and coping strategies, while controlling for demographic factors. We found evidence for associations between (a) the usability of password policies and the neuroticism trait; (b) password strength perceptions and openness, respectively conscientiousness; (c) password length and neuroticism; (d) coping strategies and extraversion; and (e) IT-background and more secure behavior. Although the associations were not always particularly strong, they are still useful to inform the design of persuasive interventions and password policies. To that end, a set of four password personas was created to segment user groups by behavioral, attitudinal, and demographic archetypes.

### Future Work

Our work was of exploratory nature. Thus, the next step should be a focused study that increases statistical power to confirm or refute the associations. We provided a user segmentation in our personas to derive testable hypotheses. The refined knowledge about personality traits can be used to personalize password feedback and make it more effective. As pointed out by related work, some users overestimate the strength improvement of using digits in passwords [342], which we also saw in studies 2 and 3. However, not all participants were equally influenced by digits or different character classes inside passwords. Respecting the psychographics in real-time feedback while carefully enforcing sensible policies could make these messages more effective. Besides, personality traits can inform choice architectures beyond passwords. For example, the Choose-Your-Own-Authentication approach [125] can benefit from improved default settings depending on personality trait characteristics. The collection and aggregation of the necessary information could be done implicitly on mobile phones or social networks [80, 311, 386]. We see this a promising opportunity to improve the user experience of authentication mechanisms, because off-the-shelf technology can already achieve this goal. Finally, in light of some participants being more positive towards including emoji in their passwords opens new possibilities for password authentication. Emoji usage itself has been shown to be associated with personality [223], which is supported by our research. Thus, it is important to investigate how personality traits might be exploited to model password strength of emoji-passwords.



## Take Aways

- Personality was weakly associated with all the measured dimensions: strength perception, policy preference / usability, password selection and coping strategies. Most notably openness, conscientiousness and neuroticism showed the most conclusive associations.
- The models for the perception of passphrases achieved the highest fit, suggesting a predictable association between personality and strength perception for this type of password. Comparing two passwords was associated with the conscientiousness traits. Mixed models that use both password features and personality trait scores as covariates were the most feasible approach to improve model fit.
- We can use “Password personas” to inform design decisions of persuasive interventions in the future. For instance, older users might be the best target group for password support tools, because age was a good predictor of their usage. Suggesting good tools during account creation might lead to higher adoption. Nudges designed for neuroticism should make emotional state more salient and positively reinforce the benefits of long passwords.

---

## Mental Models of Password Managers

An important spillover of our previous exploration is that password managers are more likely adopted the longer people had struggled with passwords: Older participants in the third personality study (Section 7.4) were more likely to use a PWM. We have already corroborated market surveys that indicate a generally low adoption rate of password management software. As discussed in Sections 3.2 and 3.3.4, it has been hypothesized that users do not fully trust third parties with their credentials, so there seems to be an urge to stay in charge. Consequently, most users still try to memorize their passwords. On the other hand, password managers do provide many usability and security benefits, but why do users fail to see them? To this end, we see a lack of understanding about how users make sense of password managers. Our goal was to understand users' mental models of password managers first and then identify opportunities to improve them, which could increase adoption rates. We thus aimed to answer the following research questions: **(1)** How do users think a password manager works? **(2)** How does adopting a password manager change user attitudes and behaviors?

To answer these questions, Martin Prinz and I explored attitudes and understandings in semi-structured qualitative user interviews. To get a more complete picture, we interviewed both people who already use a PWM and also people who prefer other coping strategies.

### Publication Statement

Parts of the outcome of this investigation have been published as an extended abstract at SOUPS 2017 [257]. For the most part, the chapter picks up the content of that paper and puts it into the larger context of this thesis. There are two significant additions:

- The chapter includes the detailed structure of the mental model
- The chapter elaborates on the design of a value proposition for password managers that takes users' mental models into account.

---

## 8.1 Background and Context

Password managers can be either built into web browsers or act as a standalone solution that is independent of the password's purpose. Dedicated password managers have existed since the mid to late 1990s. Web Confidential<sup>1</sup> was probably one of the first programs to facilitate password management, when it first surfaced in 1998. Which of the browsers was first to add password storage capabilities cannot be easily traced back, but all major browsers added this feature in the early 2000s. Given the long history of supporting authentication with software tools, adoption of password managers is still at only 12% [247]. Even security experts disagree on the specific security benefits of different implementations<sup>2</sup>. If the auto-fill feature is enabled, this can be used to create digital footprints for individuals<sup>3</sup>. Nevertheless, similar attack vectors could easily target regular password entry, and are not limited to auto-fill.

There are different service architectures for handling passwords: *offline* password managers keep a database of encrypted passwords locally on the user's machine, while *online* managers provide more portability because passwords are held on a server or a distributed storage solution [228]. *KeePass* and *Password Safe* are notable representatives for the offline storage paradigm, while the cloud-based approach is dominated by third-party solutions like *LastPass*, *1Password*, and *Dashlane*. Browser vendors have also transitioned to store passwords in the cloud, e.g. Apple Keychain for Safari, or Google Smartlock for Chrome. On the one hand, this provides consistent user experiences across multiple devices. On the other hand, such architectures create lock-in effects and dependencies on the browser. To remove those drawbacks, third party tools typically provide browser-extensions to automatically fill username and password fields. This way, they can create similar experiences as built-in PWMs, but the user remains locked into one solution most of the time.

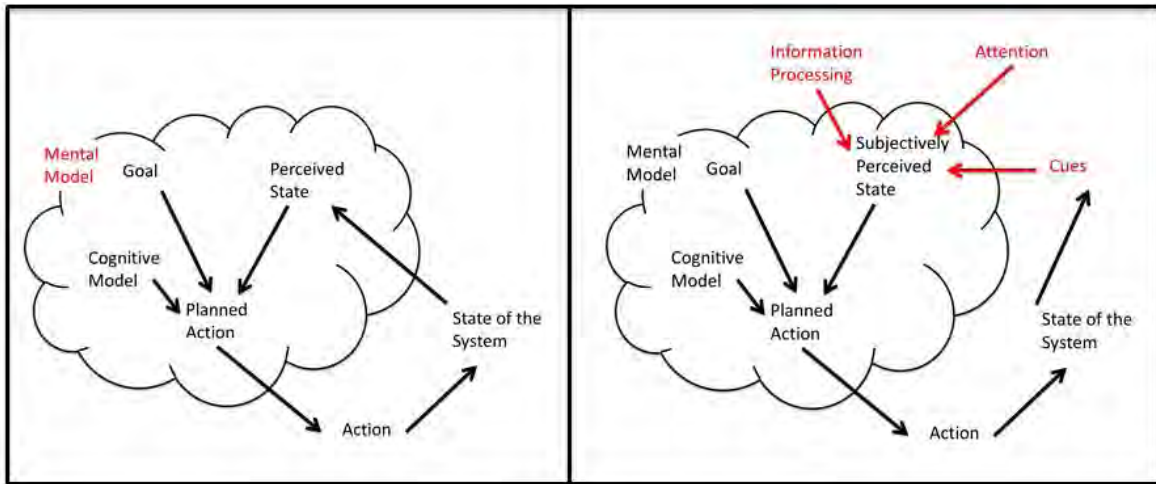
**Mental Models** Norman suggests that exploring mental models provides predictive and explanatory power for understanding an interaction [244]. Volkamer and Renaud meticulously described different aspects and definitions of mental models, which are too elaborate to discuss in this work [352]. As a takeaway, a mental model describes a user's sense-making of any system they interact with. Volkamer and Renaud highlight that mental models are not necessarily static, but can be shaped with different cues to internal feedback loops (see Figure 8.1). For our purposes, we use a simpler definition provided by Young: Mental models are "collections of the root reasons why a person is doing something" and "represent what a person is trying to accomplish in larger context, no matter which tools are used" [385, p.11]. In our case, they describe the reasons for (not) using a password manager, as explained by current actions to cope with authentication tasks.

---

<sup>1</sup> <http://www.web-confidential.com/> (last accessed 16.02.2018)

<sup>2</sup> <https://www.wired.com/2015/07/websites-please-stop-blocking-password-managers-2015/> (last accessed 16.02.2018)

<sup>3</sup> <https://freedom-to-tinker.com/2017/12/27/no-boundaries-for-user-identities-web-trackers-exploit-browser-login-managers/> (last accessed 16.02.2018)



**Figure 8.1:** Volkamer and Renaud see the formation of mental models as loop involving different plans, perceptions, system structures, and actions. Image from [352]

In Chapter 5, we took a quantitative approach to elicit data on the mental models. This was useful because we tried to understand *what* contributed to password strength perceptions and by *how much*. For password managers, related work on usage motivation is scarce, thus we strove to answer *why* users perform certain actions and *how* these could be supported by a password manager. Therefore, qualitative methods were more useful. Eliciting such data to understand mental models, Bravo-Lillo et al. relied on open-ended one-on-one interviews both with advanced and novice user groups [39]. They highlight the usefulness of this approach to understand thought processes. Kang et al. relied on *drawing tasks* to elicit additional data [185]. Participants were asked to sketch what they thought happens to their personal data when they interact with different online services. During sketching, a *concurrent think-aloud protocol* was used to avoid misunderstandings. Once the data has been collected, Young suggests the *affinity diagramming* method to derive themes and identify opportunities for supporting tasks [385]. In this research project, we combined interviewing, drawing tasks, and affinity diagramming.

## 8.2 User Interviews

Our main objective was to understand how users make sense of password support tools, in particular password managers. This would allow us to explore solutions that fit user expectations better.

---

## 8.2.1 Method and Protocol

We chose to conduct semi-structured open-ended interviews for the reasons mentioned above. The sessions started with a thorough briefing about the study purpose and asking for permission to audio-record the conversation. The main questions were (1) *Why do you need passwords?* and (2) *What is your strategy to manage multiple accounts?*. From there, the study followed up with more fine-grained questions to investigate the specifics of these two aspects. Moreover, Bravo-Lillo et al. showed the benefits of drawing tasks to find structural patterns in beliefs [39], so we also asked participants to (3) *sketch how a password manager works*. This required that participants were aware of PWMs. If they were not, they were told that it is a “piece of software that stores a user’s password”, which was expected to be vague enough to explore participants’ unbiased expectations of this kind of software. The interviews took between five and 16 minutes.

### Recruitment and Sample

We first approached random passers-by on a popular street in Munich to obtain a diverse sample of participants. Six interviewees were recruited this way. However, these first six interviews showed that participants were unable to provide sufficient detail in answers to allow thorough analysis later. Thus, we changed the recruiting method and approached employees of a design agency with which we had collaborated in the past. We also knew that the agency’s policy required employees to utilize password managers. Moreover, the user group was more likely capable of visually expressing mental constructs, allowing for the envisioned analyses. Eight additional participants were interviewed in this sample, giving a total of N=14 (6 male, 8 female). They worked as experience designers and concept developers, and did not have formal training in computer science or engineering. The age of all interviewees ranged from 20 to 41.

None of the interviewees in the first group had used a password manager before, so we call this participant group the *novices*. Since the PWM was part of the company policy, all interviewees in the second group had used one before, so we call them the *actives*. The separation allowed us to detect a shift of expectations before and after adopting a password manager.

### Method Limitations

The recruiting and sampling methods are inherently limited. The novice user group was asked at a public spot, so it was difficult to provide enough contextual information and minimize distraction. On the plus side, we achieved diversity and the face-to-face set-up reassures trust, because it was clear that none of the information they gave us was going to be used to access their online accounts. 5 of 6 *novices* were unable to describe what a password manager was, before we gave them the above definition. Thus, their decision to refrain from using a PWM was not made actively, but rather results from the lack of awareness. This limits the analysis of attitudes and self-reported behavior. Finally, the second user group has

a homogeneous background: design and communication. Moreover, they were all using the same password manager due to the policy at their company. Finally, the size of the sample might have been too small to achieve theoretical saturation, i.e. additional participants might have brought up aspects that were not covered at all by the existing sample. All these limitations demand that the results not be generalized to a larger population. Instead, they should be seen as rough trends that help understand a first set of underlying mental models, rather than the entire spectrum thereof.

## 8.2.2 Data Analysis and Results

Young proposed a design strategy to translate qualitative data into mental models [385]. The method is based on *affinity diagramming*. The resulting clusters and themes from the diagram are then mapped to a hierarchical structure that consists of Mental Spaces, Task Towers, Tasks, and Particular Tasks. Here, we focused on those Mental Spaces and Task Towers that involve password managers. Table 8.1 shows the resulting model in this format. In the following, we report the results that notably contributed to the formation of the model, which is described in Section 8.3.

### Selection and Coping Strategies

All interviewees reuse passwords to cope with the multitude of accounts. Participants often developed coping strategies without deliberation: they were unaware of how they cope with passwords until we specifically requested more details. Only then did they reflect and realize how they behave. One such revelation was that they categorize passwords in different ways. For instance, the context in which they created an account, e.g. the URL or policy of the corresponding website, was factored into passwords and facilitated the decision-making process which password to pick from their portfolio (see Figure 8.2 A). Similarly, the perceived importance led to distinct categories, although the interviewees had not realized this. Participants generally tried to justify their “insecure” behavior. Beside memory burden of new secrets, time pressure was mentioned as reason for password reuse.

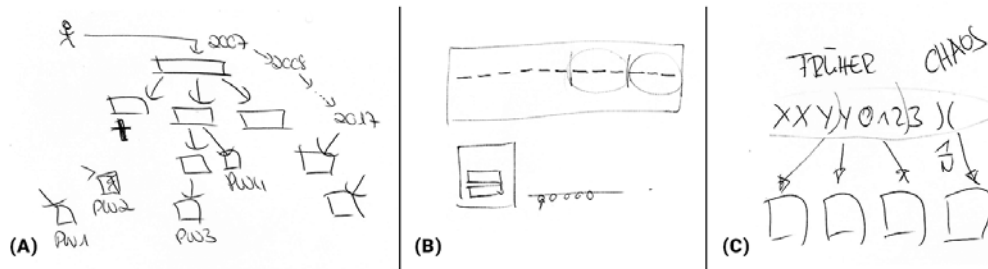
Furthermore, we also heard an interesting, deliberate strategy that seldom appears in the literature: two participants mentioned memorizing a list, or a list of letters that are then transformed into a new, quasi-unique password. This method is comparable to the Diceware technique (details in Section 3.3.3). Instead of rolling a die to generate a random number by which to look up a word from a list, those participants algorithmically select the order of words/letters based on contextual cues (see Figure 8.2 B and C). Another participant said to have memorized a randomly generated password and reusing it many times.

We also inquired situations when their strategies failed. The primary problem of having a multitude of accounts was the correct *combination* of user-name and password, which is a common pitfall of knowledge-based authentication [316]. Not only did they forget passwords, but also user names, which is just as severe because web sites generally do not inform



Mental Space	Task Tower	Tasks
Creation & Selection	Influential Factors	Personal & Historical Policies & Rules Algorithmic Strategies Account Context Memorization
	Support Tools	Generate & Memorize Use given Password Generate & Digitally Store
	Personal Algorithms	Passphrases Reuse Word Blocks Base-Password with Modifications Website Influence Use Reduced Alphabet
	Handle Failures	Trial and Error Lookup Password Show Entered Password in Plain Text Reset Password
Log-In	Manual Tasks	Copy & Paste password Lookup hints and cues Recall from memory
	Simplify	Stay logged-in Cross-device support Autofill forms Rely on manager
Organize and Commit	Share Passwords	Secure sharing with colleagues or friends Write down password Reset after sharing
	Use Aid	Password Manager Word/Digital Document Handwritten Notes
	Memorize Passwords	Algorithmic Website Cues Mental Drawer Base Password and Modifications
	Protect Passwords	Modify Passwords Unlock access with Master Password Hide or Encrypt File/Notes
	Automation	Reset Multiple Passwords Autosave Credentials Autofill Username and Password Warnings when websites are compromised
	Build Password Categories	Security Importance Frequency Policies & Rules Mental Drawer

**Table 8.1:** Mental Model of Authentication and Password Management, adapted from Martin Prinz [256]

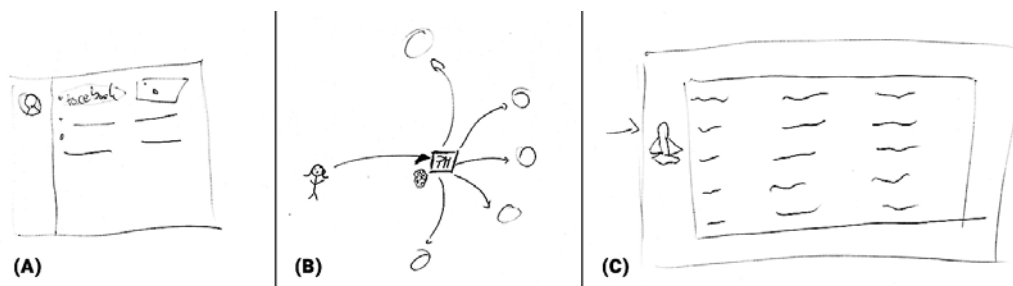


**Figure 8.2:** Participants were asked to visualize their methods to create passwords, if they had a specific strategy. (A) participant who categorizes and remembers passwords by the time they were created (context factors). (B) participant uses words and cued positions to recall the constellation. (C) participant who uses a fixed set of characters and algorithm to “calculate” the correct constellation.

users which was the error source to thwart attacks. Participants reportedly use a trial-and-error approach to go through their portfolio and ultimately use self-serviced password resets as a convenient solution. Interviewees expected that websites offer this kind of fallback scheme, because two of them do this on a regular basis.

### Password Manager Impact

Overall, the *novices* and *actives* did not behave differently in their selection and coping strategies at first sight. However, we found that *actives* did in fact change some habits when they had started using a password manager. First, although they were initially exposed to PWMs at work, *actives* started using them in private shortly afterwards. This interaction and experience with the tool led to their migrating passwords into the manager step by step. One participant mentioned that it helps him stay organized where there was “chaos” before (see Figure 8.2 C). Others were somewhat ashamed of their weak practices before using this kind of software. Password sharing with other users was the central advantage for four interviewees, especially at work. It facilitates secure collaboration with colleagues and clients. Participants do not memorize these passwords, because they often use built-in password generators. They realized that shared passwords are short-lived because colleagues leave or contracts with clients end, upon which passwords are invalidated. One interviewee fully embraced generated passwords even for private purposes and only memorizes his master password. Interestingly, however, for their most important accounts, most *actives* kept on manually crafting passwords and refrained from putting them into the PWM. Having used the tool and become aware of their own weak behavior in the past, they had gained confidence in selecting stronger passwords. This gain of mastery left them with a positive experience of password managers. As a contrast, *novices* were all comfortable with how they managed their passwords and did not show that sense of insecurity.



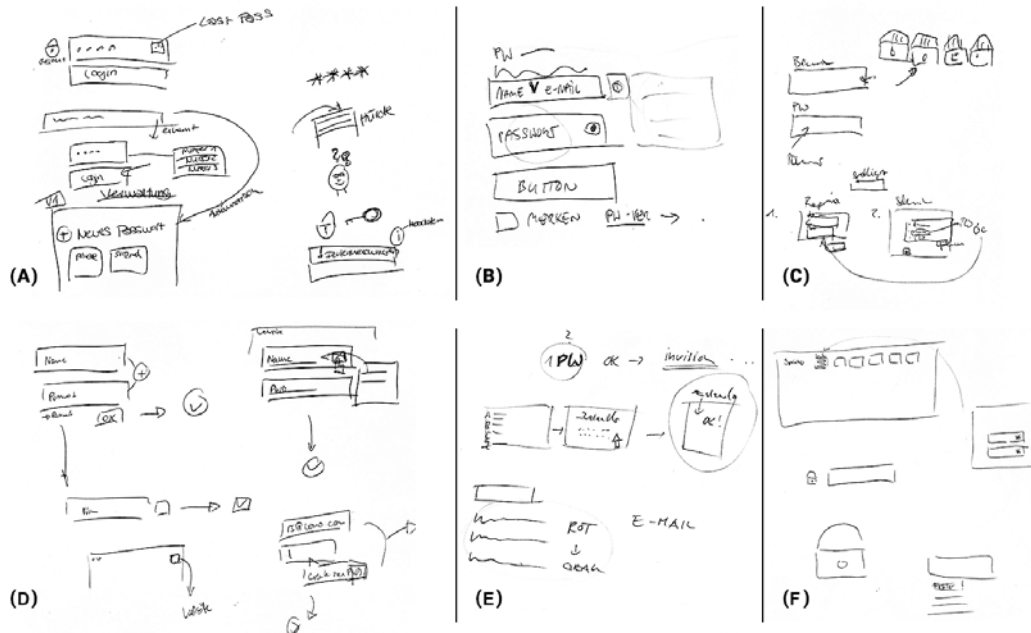
**Figure 8.3:** Drawings to the question “What does a password manager do, and how?” from three *novice* users. (A) only shows a profile on facebook. (B) emphasizes that the PWM is a central hub and acts as the user’s “brain” for different entities. (C) shows a table-like structure that holds all username-password tuples.

## Drawing Tasks

Participants were asked to sketch their thoughts whenever this was appropriate. We encouraged them to sketch as much as possible. All participants mentioned that it was challenging to communicate their understanding this way. Already for basic functionality and purpose of passwords, we asked to create sketches. This task was still relatively easy for all interviewees. Padlocks and keys were two of the most commonly drawn elements. It also helped to communicate individual elaborate selection strategies (see Figure 8.2). However, the difficulty rapidly changed for the workings of a password manager. Here, especially *novices* struggled with the task and could not proceed without further explanation, telling us that it is hard to frame an unfamiliar concept with a sketch. Their drawings were more vague than those of *actives*, which might also be due to the different professional background and expertise in creating concepts. The drawn elements and metaphors differed among the two groups:

**Novices** had a vague model of how such a password manager might work and found it especially difficult to sketch this. The benefits and system architecture of the software were unclear to them. One interesting drawing depicted a password manager as a virtual “brain” that acts as a central hub to make the user’s life easier (Figure 8.3 B). Another participant, who reportedly used a Word document to keep track of his accounts, imagined a password manager to behave the same way (Figure 8.3 C). The only expected difference would be that accessing the list of passwords would be protected by a password, which is indicated by the keyhole and an arrow that points to it. This represents *novices’* understanding on a more general level, as they explained a password manager as a special way to manage a *secure list of passwords* that helps find the right ones. Five said that it facilitates logins by allowing them to **copy-and-paste** passwords from the manager into the webpage.

**Actives** Since all active users were also visual communicators, it was somewhat easier for them to sketch the workings of a password manager. They clearly focused on the interaction between users and the system and highlighted the benefits in their drawings. Having expe-



**Figure 8.4:** Drawings to the question “What does a password manager do, and how?” from six *active* users. Common components are UI elements that link to other entities, therefore the interplay between user, PWM and website is clearer.

rienced the advantages, they strove to convey these visually and came up with more details (cf. Figure 8.4). Instead of a password-table, we can see workflows that show the interplay between user, database and website. Common components are UI elements like input fields or buttons that link to other screens or entities. Only one interviewee from the *active* group refrained from UI elements and instead sketched a flow-chart.

## 8.3 Mental Model

From the behavioral, attitudinal, and experiential data, we created a mental model schema in the style of Young [385] (see Table 8.1). We tried to stay close to the data as possible, but a few points are enhanced by knowledge from related work. We briefly elaborate on the mental spaces to allow the reader to delve into task-towers and tasks.

**Password Creation and Selection** First, users have a variety of particular needs when they are challenged to create a password. This task tower describes both the constraints, prior experiences and strategies to accomplish the task. Our participants often mentioned highly individual selection strategies that allow for both secure and memorable secrets. From a support tool, they expected guidance and feedback. Password generators that simplify this task can aid here.

---

**Log In** Most prominently, authentication still involves *both* manual and automatic tasks. Interviewees expected to copy and paste passwords from the manager to their browser, or at least provide a way to retrieve password hints. On the other hand, participants expected that support tools are deeply integrated into the browser by automatically filling password fields in a highly reliable manner. If possible, the solution should work across multiple devices.

**Organize and Commit** The third mental space resembles the “Commit to password” and “Live with password” stages of the Password Life Cycle [316]. Each participant mentioned some way of organization strategy that allowed them to live without password managers to some degree. However, these were not always deliberate choices, but rather have formed over the years. Especially *novice* users made sense of password managers by their capability of protecting passwords, i.e. encrypting a list of passwords rather than just storing them in a Word document. *Active* users were already aware of the sharing capabilities and appreciate a simple process to reset passwords when they need to be invalidated for security reasons.

## 8.4 Opportunities and Challenges

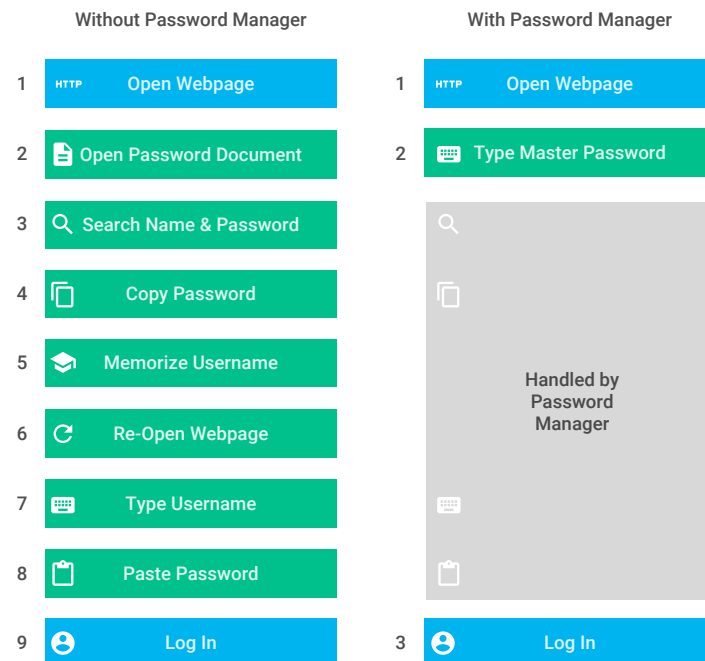
Having fine-grained insights into the mental models of password authentication sub-tasks lets us explore novel ways to support users in many challenges through simplification. In the following, we highlight key opportunities for future work on password managers and persuasive password support in general.

### 8.4.1 Leveraging and improving novices’ mental models

There were two important preconceptions about password managers on the *novices* side: simple password lists and copy-paste interactions. Future password managers can leverage these models to persuasively communicate functionality. The value proposition should thus ensure that potential users understand that PWMs are *not only* a secure list of user-names and passwords, but also help them *select* passwords – a benefit that *actives* had realized in retrospect. The *password list* metaphor is also useful to communicate automation features: explicitly showing new users that they do not have to search through the list, nor copy and paste passwords from the list to the website might help them understand the simplicity of the interaction paradigm. This can happen during an onboarding user journey, e.g. with an image showing the steps saved by the PWM (cf. Figure 8.5).

### 8.4.2 Increasing Sense of Agency

While automation simplifies processes and thus improves usability, staying in control of security-related interactions is important to users. *Novices* were confident in their current



**Figure 8.5:** Visualizing users' previous behavior might help communicate how the password manager can simplify authentication tasks. This can create a better mental model of their workings.

behavior. *Actives* had realized that their past behavior was sub-optimal, but they had gained confidence to create passwords and handle authentication for their most important accounts on their own. As a consequence, a password manager needs to stay flexible enough to respect user preferences for different account *categories*, e.g. unimportant vs. important accounts. A PWM could learn which types of websites the user prefers to handle on their own and refrain from asking to store the credentials at similar sites in the future. Moreover, reassuring users that handling such situations on their own is reasonable can inspire trust in the system. A PWM could automatically detect when it is appropriate to offer help. Current managers only provide the opportunity to decide whether the password should be saved, maybe saved later, or never be saved for a particular site. Such decisions can be automated once enough training data has been provided by the user.

### 8.4.3 Leveraging Context

Context factors can be leveraged by PWMs to adapt to different situations. For example, usage-context informs future interactions. If the user sets up the system at work, this is an indicator about how passwords are going to be categorized, how often they will be reset, and how likely they are going to be shared with others. Adapting the interface to such scenarios can simplify interactions and the formation of mental models of the benefits. Moreover, automated generation of passwords is also context-dependent. As we have shown in Chapter

---

6, password policies in the wild impose varying restrictions on the use of characters. To avoid user frustration arising from rejected passwords, e.g. because they contain forbidden characters, the generator can ensure the random password meets the website's composition policy.

#### **8.4.4 Customization and Personalization**

It is evident that current password managers work fundamentally differently compared to how users normally cope with passwords. Our participants reused passwords with different strategies and relied on digital documents, paper notes, and highly individual password creation or memorization techniques. Beside the document-metaphor discussed above, the other strategies are not reproduced by password managers. However, a general philosophy in user-centered design is the aim to “fix the system” rather than to “fix the user” and acknowledge that people prefer maintaining existing strategies, e.g., to make ensure being able to log in even without assistance. A PWM can hence make a user's creation experience more convenient by leveraging existing strategies. For instance, it could let the user specify a creation technique: if they usually modify a base-password depending on the context, the PWM could offer to generate passwords like this in future scenarios, i.e., automate the modification strategy. While this approach would not necessarily improve security, it could save the user time and remove critical pain points. Besides, the user stays independent of the tool, because they can reproduce their personal system and authenticate on other devices even without the support.

### **8.5 Conclusion**

To understand additional psychological context factors of persuasive password support, we explored the mental models of authentication tasks and password managers with a qualitative approach. Both participants experienced with password managers and inexperienced novices shared their insights, attitudes and behaviors during short interviews. Fourteen interviewees provided us detailed descriptions of their password selection and coping strategies, and how they make sense of supporting tools. We contribute evidence that (a) individual coping strategies persist even after adopting a password manager for important accounts, (b) work environments serve as onboarding triggers even for private use (RQ2), and (c) novices were mostly unaware of functionality and the ramifications of adopting password managers (RQ1). These findings show that the value proposition should be communicated concisely. One potential solution could compare the benefits of using a PWM to inefficient and insecure password practices. At the same time, future solutions should respect highly individual



coping strategies to match user behavior better. Ultimately, this could increase adoption and, more importantly, retention rates.

### Take Aways

- Password managers appear to be a “black box” for people who have never used one. They suspect that such tools are a slightly more secure version of text files to write down lists of username-password-combinations.
- The mental model of password authentication and managers is mostly divided into the mental spaces “Select password”, “Log in”, and “Organize”. There seem to be discrepancies between current user behavior and current password managers.
- Making password managers adapt to context and individuals seems a promising direction for future systems.

---

# III

## PERSUASIVE DESIGN STRATEGIES



## Exploring Needs in Persuasive Feedback

The overall goal of this thesis is to support users in how they interact with passwords. As a sub-goal, we wanted to help those users who want to handle password management on their own, i.e. without a password manager. We set out to achieve this goal by studying the design space of password feedback mechanisms like password meters, i.e., the various dimensions, choices and outcomes of a design challenge. There have been many proposals and numerous evaluations already for such feedback systems. However, the design space appears rather narrow if we only look at existing solutions, and many have not supported users well enough. Therefore, we posit that the design space needs to be opened up in more breadth. To get there, studying the requirements of password feedback is a necessary first step. Previous solutions have seldom reported a structured requirement elicitation based on user research. Thus, before designing and implementing support strategies outside of the usual spectrum, we first aimed to understand the needs and expectations that users may have about password feedback. We tried to learn from feature requests and co-designed proposals as a starting point for generating novel ideas about actionable password feedback. In particular, we posed the following broad research questions:

- RQ1 What do users *expect* from password support tools during account creation (explicit) and what do they really *need* (implicit)?
- RQ2 How can we design password feedback in novel ways outside the usual spectrum of visual password meters and verbal real-time feedback?
- RQ3 How can we leverage the interplay between feed-*forward* and feedback?

This chapter reports on a requirements- and design-exploration for password feedback, where we wanted to involve users early in the process. The project was a collaboration between myself and Caroline Olsienkiewicz, respectively Katharina Schwarz, who wrote their bachelor theses on specific aspects of the exploration [248, 284].

---

## 9.1 Background and Context

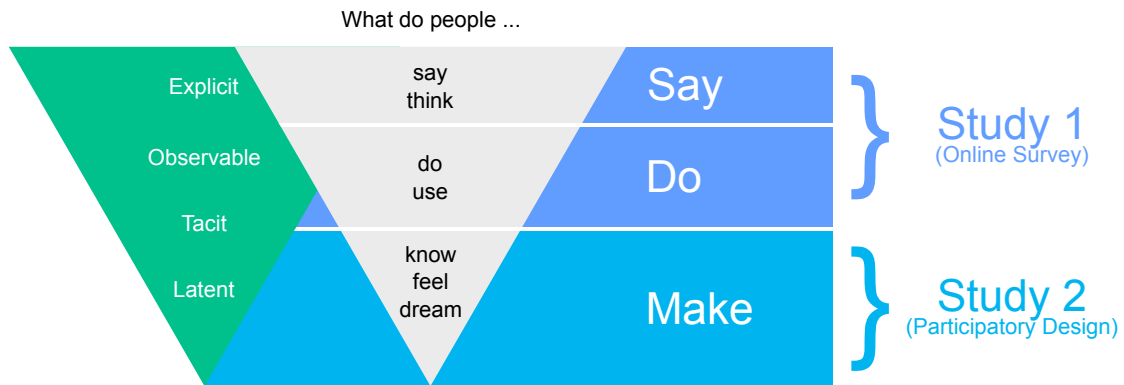
The design space of persuasive interventions in authentication was put into a framework by Forget et al. [124]. They established high-level strategies to steer users away from risky actions. However, while their Persuasive Authentication Framework described how hypothetical design solutions could be categorized, it did not tell us much about the specific user needs in feedback systems. The same goes for more generic persuasive models like Cialdini’s “six weapons” [57], or Fogg’s Behavior Model [119]. The latter provides hints to user requirements in system design, but those are not specific enough for password authentication.

On the other hand, there exists a wide range of designs for persuasive password support that have been empirically evaluated, too. For instance, Ur et al. studied feedback with numerous solutions [341, 343, 344]. Egelman et al. also explored novel designs for password meters [99]. Moreover, Shay et al. continued Forget’s exploration of real-time feedback and suggestions [126, 300]. Beyond real time feedback and meters, Komanduri et al. [199] and Yee [384] showed an approach to make predictability of passwords graspable. All these publications report on *solutions* and corresponding usability evaluations, but they often do not talk about the *needs* that users have regarding the support.

Thus, our project focused on eliciting these needs and provide a holistic view on password feedback and support systems. Ultimately, understanding the needs might help us broaden the design space further and find more effective and actionable password support. To get there, we chose a “put the user first” lens and conducted two studies that focused on different levels of user needs (see Figure 9.1). In the first study (Section 9.2) we wanted to learn explicit and observable needs, i.e. the things that users *say* they need or that can be understood by *observing* how they interact with a solution. The second study (Section 9.3) was more involved with tacit and latent needs. These are hard to observe and require more engagement with users, to understand what they actually know and feel when they interact with a password support system.

## 9.2 Explicit and Observable Needs

After the literature review (see Part I), there were still open questions about user **expectations** regarding password strength feedback. Most studies went ahead to test new solutions quantitatively and rarely focused on the intrinsic needs to derive requirements for feedback. In other words, users were often involved late in the design process, although an earlier involvement could have led to different design decisions. Our goal in the first step of the co-designing process was gathering early insights to learn about the requirements of persuasive interventions in the realm of password authentication. To narrow down the focus of the requirements, it was important to consider a realistic use-case where the burden of password authentication is relatively high. From related work, we know that password



**Figure 9.1:** Characteristics of user needs and how they show (adapted from [16]). We explored explicit and observable needs with an online survey, and used a participatory design approach to learn from what users *make*.

changes are more frequent in work environments, because expiration-policies are still common there [172]. Frequent password changes lead to weaker practices, which might be mitigated by effective password feedback. To find out how we might improve such feedback, we used a lightweight, rapid survey with a large portion of open questions. In the following, we briefly describe the method and findings.

### 9.2.1 Method

We opted for an online survey to inform our requirement elicitation for the above described reasons and straightforward data collection. Our questionnaire was centered around the following core research questions:

- What do users **expect** from password feedback?
- What do they **miss** in current solutions / interventions?
- How would they **improve** existing feedback?

#### Prototype

To ensure that participants have a shared understanding of password feedback mechanisms, we created a small website featuring only a password field and textual feedback beneath. The underlying strength estimation and feedback phrases were based on the zxcvbn library. Warnings and suggestions that come with zxcvbn were translated to German and shown as the users typed inside the password field. The 26 warnings included statements like “Names and surnames by themselves are easy to guess” or “Avoid repeated words and characters”. We tried to simplify the page as much as possible to minimize priming effects, although this was hard to achieve. In particular, we did not show any form of visual feedback to



---

explore whether this is a need that participants mentioned often. Instead, we showed a password score in the form “strength: 2/5”. The page introduced the scenario (old password at work has expired) and asked participants to create a new password that would fulfill the composition policy of the participant’s current employer. The composition policy, however, was not enforced, but we probed it in the questionnaire. We did not log the passwords whatsoever, since this was not the focus of this study. This was disclosed to participants, and they could decide to drop out of the study if they were still uncomfortable with providing a new password.

## Questionnaire

The survey questionnaire included 37 questions, many of which are quick single- and multiple-selection items to collect context information (e.g. demographics, semantic differentials, social desirability scale). This information would help us assess the value and relative importance of qualitative statements that followed. The core of the questionnaire is formed by eleven items about password feedback, six of which were open questions, e.g. what would need to be changed to make the feedback more comprehensible and how password creation could be facilitated. Moreover, we inquired coping strategies at work and personal password behaviors to derive further requirements. During this part of the questionnaire, we added an *instructional manipulation check*, i.e. an attention check, to filter out participants that were only trying to complete the survey as fast as possible to receive the incentive [249]. This is a well-known issue with crowd-sourced survey data, and the attention checks can effectively reduce the risk of low-quality data. The items were randomized where necessary, but the overall structure was the same for all participants, i.e. there were no independent variables.

## Sample

We recruited participants via Prolific<sup>1</sup> which provided similar crowd-sourcing features as Amazon’s Mechanical Turk but has a stronger focus on research surveys. Since mTurk does not allow German users to sign up, Prolific is one of the best alternatives because of its large user panel. We screened for German language proficiency via Prolific’s internal screening tool. The survey also required participants to be employed and using alphanumeric passwords in their work environment on a regular basis. From the 87 users who started the survey, we had to eliminate 47 responses based on previously defined exclusion criteria: incomplete or mechanically translated; incomprehensible answers; failure to complete the instructional manipulation check [249]; and fourth-quartile scores on the social desirability scale. The remaining 40 respondents had a diverse educational and professional background, but the largest part (n=15, 37.5%) held positions in IT or online media. Sixteen were female (40%) and 24 male (60%). They were aged between 20 and 53 ( $M = 32, SD = 7$ ). As an incentive, participants received 1.50€ for approximately 15 minutes worth their time, which meets Prolific’s work ethics guidelines.

---

<sup>1</sup> <https://prolific.ac> (last accessed 06.03.2018)

## Analysis Approach

We performed structured, iterative thematic analysis of the qualitative data. This approach is inspired by Grounded Theory and useful for exploring sentiment and mental models early in the design process [321]. It consisted of three distinct parts: open coding, axial coding and a final selective coding stage. In the first stage, the data is labeled with an unlimited number of fitting codes. Next, the codes are grouped and abstracted. A second coder independently put the codes into the groups. Differences were discussed until a common solution was found. Finally, the number of code-groups are reduced to the essential themes that best describe the thinking processes, or as in our case, the requirements.

### 9.2.2 Overall Results and Central Themes

Overall participants were positive about receiving support during password selection, often because “any kind of help is good help”. Some stated they were convinced that feedback is helpful to create stronger passwords. The primary benefits they mentioned were reduced frustration and simplification, because it gives reassurance. The latter is a direct lead to the Persuasive Authentication Framework [124]. If participants were negative about password feedback (n=4), their primary qualm was that they were convinced to achieve strong passwords without external help. In the following, we present the central themes in participants’ statements. Multiple coding stages, discussions, and the selection tasks helped us identify overarching needs: **Show, Explain, Help, Empower**. To keep the narrative coherent, we omit all corollary results that did not guide further research steps. These are reported in higher detail in C. Olsienkiewicz’ thesis [248].

#### Show

“Show” appears to be the most important category because at least one answer from each participant could be put into this code. Unsurprisingly, participants preferred **visual feedback** over verbal feedback. Our prototype refrained from graphical elements entirely, but participants wanted a visual representation of strength. Most notably, a “horizontal bar” was mentioned, i.e. a simple password meter. Some participants felt that verbal strength categories are patronizing. “Colors” and “steps” were commonly mentioned. This theme echoes Ur et al.’s quantitative findings [341].

At the same time, the “Show” theme encompasses that participants would like the password reset form to *show* what it expects from the user. If digits or symbols are expected, the feedback should *show* their impact on the users’ security or show the risks that are taken by not including symbols and digits. P4 mentioned that if old passwords are disallowed, then these passwords should be shown when creating a new one to know what is going to be forbidden. Hence, “show” does not only include *feedback*, but also *feedforward* to align expectations. Therefore, compliance is a mutual contract, and the theme again highlights Sasse and Flechais’ argumentation that authentication schemes are *socio-technical systems* [277].

---

## Explain

When asked what advice they would have expected from the feedback, answers clearly indicated that participants had a fixed notion of what makes a strong password. Verbal feedback, in their view, should point out that digits, symbols, uppercase letters, randomness, and password length are beneficial for password strength. Most of this is in line with findings from previous chapters and related work. It is interesting, though, that this consensus also shows us that the feedback would not be necessary at all. Responses from participants who unexpectedly received a low strength rating and feedback highlighted an important aspect: If feedback stands in **stark contrast** to how the user perceives their password's strength, explanations are both welcome and necessary: "Why is the password rated only with two out of five stars?" (P7) – "How precisely is the strength and feedback determined?" (P34,P28). The desire to understand the rating is hence the key to correct mental models, but needs to be cued by lower-than-anticipated strength ratings. Potentially, past study results pointed towards the inefficacy of password meters, because they were not accompanied by explanations in case the rating contradicted user beliefs.

The theme sometimes overlaps with "show", because once a feedback system *shows* the risks, participants suggested *explaining* them in detail, i.e. explain the consequences of weak passwords. At the same time, a good comprehensible explanation was mentioned to convey the notion that service providers "know what they are talking about" (P1). Thus, nuanced "explanation-design" is essential for those users eager to learn more. Realistically, though, experience tells us that not many people actively seek explanations, but our analysis hints at opportunities to explain the details when strength feedback breaks mental models.

## Help

The "Help" theme informs design decisions around **suggestions**. Eleven participants noted that specific guidance towards a stronger or memorable password might help them. The well-known repertoire of tips, examples, "formulas" (P4), generators, mnemonics, etc. was the center of attention. Showing users *examples* was mentioned by two participants to help them understand what makes a "perfect" password and is therefore an overlap with the "Show" theme. Another two participants had the idea to show "best practices" from other users of the service. At the same time, one participant was skeptical about the use of help, because the outcome might be too predictable: "*The feedback is helpful, but [if everybody takes the advice,] won't that mean that all passwords become too similar?*" (P12). In some way, she was right, because example-passwords can persuade users to mimick the given example and become more vulnerable than before, so the design should respect this concern. One participant hinted at solving this problem by suggesting modifications based on the currently entered password, or give *personalized* suggestions. This is basically Forget et al.'s earliest approach to persuade users towards stronger passwords, which was later studied intensively by Shay et al. [300] and Ur et al. [341]. Thus, the participants' expectations are reflected in this line of research.

## Empower

The three aforementioned themes flow into the final one: **empowering** users to be creative, put suggestions into practice, and to be confident in their choice. Most notably, the concept of restricting characters and limiting password strength was a primary concern. Although we initially would not have considered composition policies as “feedback”, the participants made a fair point: rejecting a password due to a composition policy is a feedback mechanism. However, rejection can easily be perceived as destructive feedback, which manifested in the mentioned negative experiences. Instead, combining “Show”, “Explain”, and “Help” can become a full-fledged creativity support tool: Show what can be improved, explain why, and help by demonstrating alternative ways.

## Limitations

The insights should be implemented with the study’s limitations in mind. Overall, the themes we found are a small snapshot of user needs and need further consolidation for production-level solutions.

First, we were surprised that so many English-speaking users from the Prolific panel were able to make it through the platform’s screening process. Therefore, we had to discard many responses where we were unsure about the participant’s language skills. Ideally, we should have introduced a small passage of prose to which the participants answers 2-3 comprehension questions. However, we failed to anticipate this until the data was in and also did not plan the budget for the study accordingly. For the remaining participants, however, we ensured the qualitative responses were solid. Also, it would have been feasible to be able to ask follow up questions, which was a caveat of the online study method. The sample we would have been able to recruit for in-person interviews, however, would not have been diverse enough, so we were limited in the choice of methods. The number of open questions and the resulting answers were still sufficient to perform in-depth analyses and led to interesting findings.

## Intermediate Summary

With an online survey, we explored what users explicitly expect from password support. The most important take-away was that support is especially important and potentially most effective when feedback contradicts users’ mental models about their own password practices. We were able to derive a new perspective on password support. We will call it the *show-explain-help-empower* paradigm in the remainder of this thesis.

## 9.3 Tacit and Latent Needs

As second step of the co-creation project, we aimed to learn from specific user-generated designs and solutions. Therefore, we took to the participatory design methodology. In its

---

essence, it is a technique that constitutes a “shift in attitude from designing **for** users to one of designing **with** users” [16]. Therefore, having identified a research question, users from the targeted segment are involved from the beginning in design explorations and create their own solutions (“Make” in Figure 9.1). Moreover, a tight feedback loop makes sure that all designs are created under a *shared ownership*. In our case, we were eager to find out what a novel password support system could look like. In particular, unusual and exciting ideas beyond simple password meters were the center of attention. The overarching goal was to identify **what users require** from password feedback systems, which would help us design them in more persuasive ways.

### 9.3.1 Method

We used the participatory design approach to learn new requirements for persuasive feedback. The methodology included a co-design session with various exercises, an exploration and refinement of the initial ideas, a feedback loop on the progress of the concepts, and the creation of a final interactive digital prototype.

#### Procedure of the Co-Design Session

The 60-minute co-design session started with a briefing about the goals and informed consent to record the session on video. Afterwards, we sensitized participants for the topic with two *ball-bearing exercises*. For this, participants were divided into two groups and those faced each other in a circle. Each person interviewed another on a specific question of password behavior for one minute. The outer circle moved to the next interviewer until they had talked to all people from the inner circle. In the first round, interviewers were given questions on personal password selection and coping strategies. The second round focused on recommendations to different user groups. The interviewers then summarized their insights and shared them with the group. The exercise was helpful to learn other people’s password strategies and broaden the participants’ horizon. They said it was interesting to reflect about obstacles in acting more securely. In summary, participants showed typical selection strategies based on word-digit-symbol patterns and would recommend personal, memorable events as starting points for password selection.

What followed were straightforward brainstorming exercises in two groups. The question they tried to answer was “How would a registration form of an email provider need to be designed to help you create a secure password?” The facilitator encouraged participants to think out of the box and contribute unusual ideas. After five minutes, half of each group moved to the other group. In total, participants produced 17 distinct ideas and presented them on a whiteboard. Everyone received two votes for their favorite ideas. Participants then formed groups of two or three to create paper prototypes for the three ideas that had the most votes. The facilitator made sure to explain the process and the expected fidelity of the outcome. Once the prototypes were ready, the groups presented them to the entire crowd.

**Participants** We recruited seven participants through postings in public groups on social networks. Six of them were female. Three were studying philology, while one each were studying pedagogy, media informatics and business studies. The only male participant was an employed product designer. The average age was 23, ranging from 19 to 29. As an incentive, we offered a 10€ shopping voucher that was handed out after the brainstorming session. We had tried to recruit a more diverse sample, but we did not receive additional responses meeting these requirements.

### 10 plus 10 Method and Iterative Improvement

Based on the participants' paper prototypes, we designed ten variations of each idea, i.e. 30 concept candidates. This procedure was inspired by the *10 plus 10 method* to explore the design space<sup>2</sup>. Afterwards, we discussed the feasibility of each solution and identified six candidates that were presented to the study participants. Their feedback informed the decision on the final prototype. After it was implemented with standard web-technologies, participants provided a final round of qualitative feedback and a reflection on the process. Participation thus occurred in each step of the design process.

## 9.3.2 Concepts and Prototype

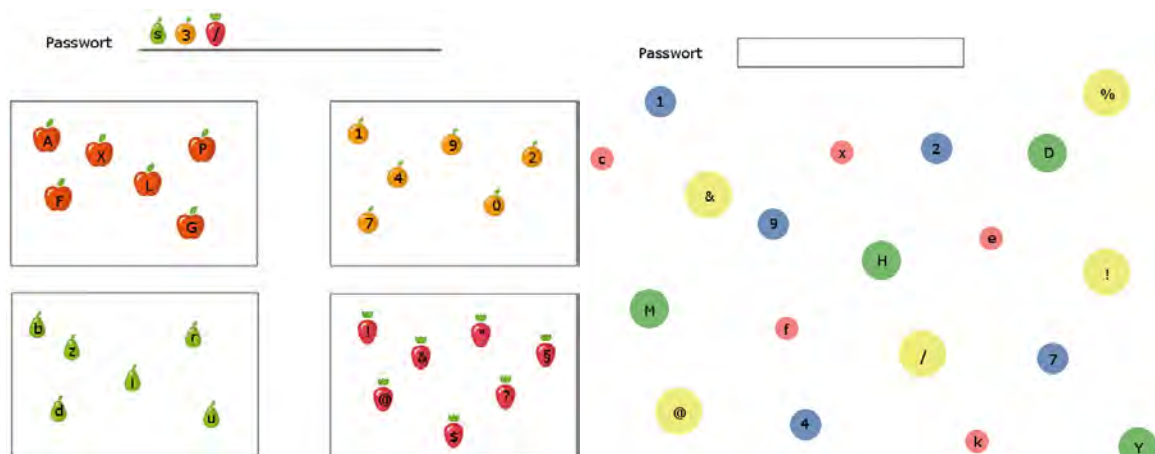
The paper prototypes of the brainstorming session showed three central components: **rewards**, **analogies and compliance**, and **playfulness**. After the 10 plus 10 method, we had six candidates that made it to the final feedback round, two of each category. Please note that we tried to stay as close to the participants' original ideas as possible and maintained concepts that were unrealistic.

**Rewards** The “beautify” concept removed all color from the website and faded it back in as the entered password increases in strength. The “friends” concept would provide a password meter that uses pictures of the user's friends instead of color to reward stronger passwords and add a personal touch.

**Analogies and Compliance** Concepts in this category tried to nudge users by making strength more graspable and salient. The “songs” concept shows the number of songs that one can listen to until an attacker would have cracked the password. The “fruits” concept (see Figure 9.2a) draws from a healthy-eating analogy and represents each character type with a fruit. While the user types, fruits are “plucked” from a character patch and moved to the password field. If a password only contains lowercase letters, the lack of complexity is immediately visible because only one type of fruit would have been plucked.

---

<sup>2</sup> <http://sketchbook.cpsc.ucalgary.ca/wp-content/uploads/Chapter-1.4-10Plus10Method.ppt> (last accessed 24.04.2018)



(a) Fruitsalad: Each character class is represented by a fruit and the user sees a lack of diversity, e.g. if their password only contains lowercase letters. (b) Bubbles: Floating bubbles on the screen that the user pops as they type the password. Symbols are placed in bigger bubbles to be more salient.

**Figure 9.2:** Two out of the six concepts that made it to the final feedback iteration.

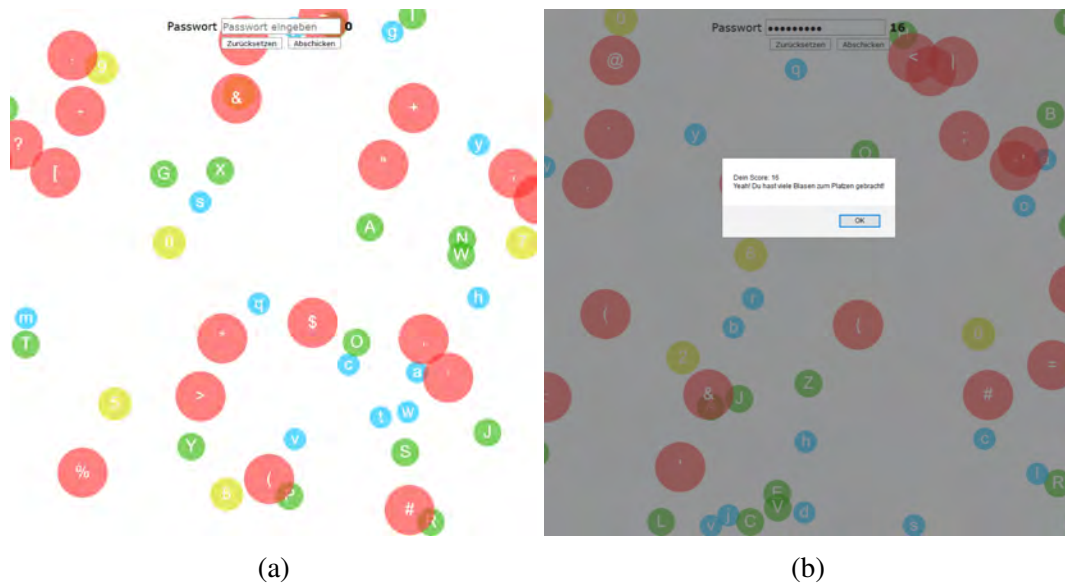
**Playfulness** The “slotmachine” idea lets the user first enter a password. Then they pull the lever of a virtual slotmachine. When it stops, it displays four random characters that would make the password stronger. The user can then claim the win by adding the random characters to their password. Finally, the “bubbles” concept lets the user burst bubbles floating on the screen (see Figure 9.2b). Inside each bubble, there is either a letter, digit, or symbol. The user can click on them directly to transfer the corresponding character to the password field, or burst bubbles by typing on the keyboard. The longer the password, the fewer bubbles remain. Also, less common character classes are shown in bigger bubbles to persuade users to burst those and hereby increase complexity.

### “Bubbles” Prototype

Participants provided feedback on the six concepts and generally considered the “bubbles” concept the best solution to motivate themselves to add more symbols or digits and also make the password longer to watch more bubbles burst. Therefore, we implemented it as an HTML5-based prototype. To add a more game-like feel, we put a score on each character class, and show the currently achieved total score in the GUI. The bubbles move slowly enough to trace them and their bursts.

Five of the participants provided a final round of feedback. On the positive side, they felt it intriguing to burst the bubbles. They understood the purpose of the bubbles and the different sizes and colors right away or after typing the first character. Two participants noted that it fosters creativity. All said that a system like this would catch their eye and might impact their selection behavior. They also mentioned a number of improvements. For instance, two





**Figure 9.3:** We made the “bubbles” concept into an interactive prototype and gathered a final round of feedback on the outcome and the process.

participants found the page chaotic and wanted it simplified. The scoring was not transparent either, and three said the bursting animation should be more obtrusive.

### 9.3.3 Process Reflection

Participants were also asked to provide feedback about their experiences along the way. Generally, they were positive and felt involved generating a new design. They reported that the final prototype reflected the entire process well and that elements from the idea generation stages were visible. In particular, they liked to think about finding solutions to a problem that they had not thought about in depth. Three of them said that they would have still liked more involvement so that they can shape the outcome even more. One participant, on the other hand, said that s/he disliked that most ideas were far-fetched and too unrealistic. S/he would have liked a more product-centric development process.

### 9.3.4 Lessons about Persuasive Password Interventions

Discussions among participants and their continuous feedback throughout the design process helped us understand various user needs. We heard that the group had not given the topic much thought before joining the brainstorming session. Many of their ideas went beyond typical password meters, which we highly encouraged. In particular, the concepts show a tendency to **visual elements** (beautify, pictures, bubbles, fruits) to catch the user’s attention. This corresponds to the *show* theme of the requirement-elicitation. The *help* theme was



visible especially in the compliance-centered concepts: The “fruit-salad” supports people in recognizing a potential lack of character diversity. While the “songs” analogy is a kind of *explanation* of password strength because it gives background information, the ideas did not contain many attempts to explain additional aspects. Perhaps, participants equated strength with complexity – a mental model that we found in Chapter 5 – and their ideas confirmed this thinking model. Finally, the “bubbles” concept aimed to boost creativity and thus showed elements of the *help* and *empower* themes.

## 9.4 Discussion

The insights from two rapid methods allow us to derive requirements for persuasive password support, which we put into the context of existing frameworks.

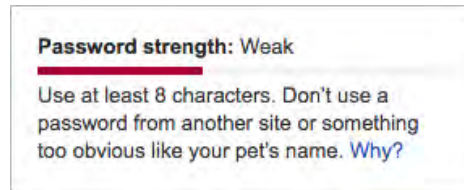
### 9.4.1 Requirements

Although the first study was focused on explicit and observable needs, the qualitative analysis even brought about more profound needs. Similarly, the second study also helped us understand explicit user needs. Therefore, it was hard to classify the needs, because they were very nuanced. Nonetheless, we show the central characteristics of the requirements in Table 9.1.

#	Requirement	Type	Study
<b>The persuasive tactic needs to ...</b>			
(1)	Visually indicate current password issues	Explicit	S1 & S2
(2)	Offer specific help to diversify passwords	Explicit	S1 & S2
(3)	Be personal	Explicit	S1
(4)	Provide background information, especially if feedback discounts mental model	Explicit/Observable	S1
(5)	Stay trustworthy	Observable	S1 & S2
(6)	Empower users to act differently	Tacit	S1 & S2
(7)	Be straight-forward, easy, and streamlined	Tacit	S1
(8)	Catch attention, be engaging, foster curiosity	Latent	S2
(9)	Open eyes and broaden horizons, e.g. through making unknown coping strategies salient	Latent	S2
(10)	Avoid habituation effects through surprise and delight	Latent	S2

**Table 9.1:** High-level requirements to satisfy different levels of user needs in persuasive password feedback.

**Opportunities** Password meters and real-time feedback are the de facto paragon of persuasion in the wild. Most commonly, they fulfill requirements (1), (2), (4), and (5). For



**Figure 9.4:** Google’s Password Selection Support UI (March 2018)

instance, Google’s password meter visually encodes password strength with color and bar-size, suggests not using a pet’s name, offers extensive background information with a clear call-to-action, and does not jeopardize trustworthiness with extravagant design. On the other hand, it does not necessarily empower users to break habits, is not personalized (what if the user does not have a pet?), and does not account for other “obvious” strategies. Academia has produced alternative designs that fulfill requirements (7), (8), and (9), although unusual concepts (like in [343]) were rarely adopted in practice. Empowerment, personalization, and breaking habituation effects have been underexplored, and thus might be worthwhile opportunities for future studies.

## 9.4.2 Relation to Current Frameworks

Most of the themes and requirements show strong links to various persuasion frameworks. In the first study, the qualitative analysis resulted in four essential themes: show, explain, help, empower. Interestingly, there are parallels to **teaching**. Teachers *show* something new, and then *explain* it in more detail. They *help* students understand it. Often students can then exercise and explore ways to apply their knowledge, i.e. the process *empowers* students to achieve something they could not do before. Forget et al., who contributed one of the early works on persuasion in cybersecurity, promoted their Persuasive Authentication Framework as a tool to **educate** users. All the elements of our qualitative analysis are thus visible in their framework. The requirements in Table 9.1 represent the themes on a more fine-grained level. Only requirements (8) and (10) are hard to put into the PAF, hence they extend the framework. Requirement (8) states that a password support tool should be noticeable and foster curiosity. Generating curiosity can be done in numerous ways, but in this context it relates to Cialdini’s *scarcity* principle. The number of “good” passwords is much smaller than the number of “bad” ones, hence they are a scarce resource. The requirement states that users should realize this and become motivated to access the limited resource.

## 9.5 Conclusion

Our goal was to understand the requirements of persuasive password support, especially user needs and expectations. We triangulated methods to rapidly derive feasible requirements that

---

guide future designs of interventions. The survey yielded explicit and idealistic needs, like having one's own mental model confirmed by the feedback system. On the other hand, the participatory approach of the second study helped us identify latent needs that would not be detectable in a survey. The results, however, fit together well and solidify the overarching themes “show”, “explain”, “help”, and “empower”. Here, we found that participants' expectations of password feedback were nuanced and can thus only be met by combining both verbal and visual support. Feedback was not the only component, because participants showed a need for more guidance in the form of *feedforward*. The requirements might have been partially predictable even without conducting user research, but tacit and latent needs are uncommonly discussed in related research. Especially the thinking processes and participants' design approaches in the participatory sessions revealed much of what they envisioned in a successful intervention. Applying the requirements to gauge existing solutions, we identified opportunities for future work in personalization, empowerment, and preventing habituation effects from repeated exposure to the same intervention. These are addressed in the following chapters.

## Take Aways

- Triangulation of study methods helped us identify different levels of user needs. Especially the tacit and latent needs are harder to elicit, but participatory design methods were feasible to enhance survey data.
- We observed that participants wanted to have their mental models confirmed with password feedback. They become skeptical if the feedback shows unexpected strength results.
- Password support systems should meet four essential user needs: **Show** the current behavior and its consequences visually, **explain** things that contradict usual mental models, **help** resolve such dissonances and **empower** users to act differently.

# 10

## Password Selection and the Decoy Effect

As laid out in Part II, one dimension of the problem space is that users pick weak passwords. In many situations, e.g. for unimportant accounts, this is acceptable. However, with rising importance, the need for a stronger password becomes salient. In these situations, mental models of password strength (see Chapter 5) play a vital role, because they influence the choice of the password. There is increasing evidence that mental models about strength are often sub-par: many users overestimate the strength increase by adding digits and symbols or substituting letters with them [289, 342, 345]. Thus, our goal is to avoid risky practices originating from flawed mental models. In this research project, we try to steer users away from weak passwords by providing more secure alternatives. Since it is hard to compete against a user's preferred password choice, we try to leverage persuasive strategies from marketing to "sell" the stronger alternative. This strategy is the so-called decoy effect. Given a range of products, vendors use this technique to make the one item stand out that is best for them (and often for the client). A decoy choice architecture includes one item that is "unattractive by design" to make a superior option salient. In numerous studies, it successfully shifted prospective buyers' reference point and achieved its intentions. Thus, we aimed to translate it to the decision-making problem of selecting passwords. In essence, we can craft password suggestions in such a way that one of the suggestions is more attractive in terms of its strength and usability benefits.

We thus aimed to answer the following research questions:

**RQ1** How can we translate the decoy choice architecture to password selection?

**RQ2** Does the decoy effect make users select stronger passwords?

**RQ3** What are the opportunities and pitfalls in this approach?

To that end, we conducted an online experiment where different groups of participants received variations of password suggestions. This chapter reports on the design and evaluation of a novel persuasive strategy and discusses the implications for its future application areas.

---

## Publication Statement

The results of this project have been published at EuroUSEC 2016 together with Stefanie Meitner, Emanuel von Zezschwitz, and Heinrich Hussmann as co-authors [291]. The data reported here is identical to the paper. The chapter puts the insights into the context of the thesis and adds two significant aspects that go beyond the original publication:

- We explain the choice architecture and design rationales in much higher detail.
- We present additional fine-grained analyses about the effectiveness of the nudge and discuss those in context.

## 10.1 Background and Context

Our project is positioned in a line of work that tries to nudge users towards strong passwords (see Section 3.4). We approach this problem through two main components: suggestions and choice architecture. We briefly provide context on these two elements.

### 10.1.1 Suggestions

Suggesting passwords was at the heart of the first work on persuasion for stronger passwords [126]. We can identify three common patterns here: (I) trying to get users to accept a suggested password verbatim [347]; (II) suggesting alterations or insertions based on their own password [126, 285, 300]; or (III) suggesting a random password that merely serves as a basis for the user's password [170]. All password generators fall into the first category because they encourage users to make sure the string is completely random. The second approach is the most common and most evaluated of the three and has seen mixed results overall. Approach (III) seems promising in environments with critical security levels requiring more stringent policies. Moreover, suggestions are sometimes included in real-time feedback where they can act as *feed-forward* (opposed to *feedback*) [341]. In summary, suggestions are an essential part of the persuasive toolkit and offer a wide range of new design opportunities.

### 10.1.2 The Decoy Effect

The decoy effect was discovered through research in consumer psychology. Buying a product usually involves deciding between different alternatives. On a high level, products are easily comparable by their quality and price, so customers heavily rely on these two metrics.

These two dimensions also build the foundation of the decoy effect. In one of the first experiments on “asymmetric dominance”, Huber et al. illustrate decision-making processes with a simple example to decide between six-packs of beer that differ in price and quality [167, slightly adapted for simplicity]:

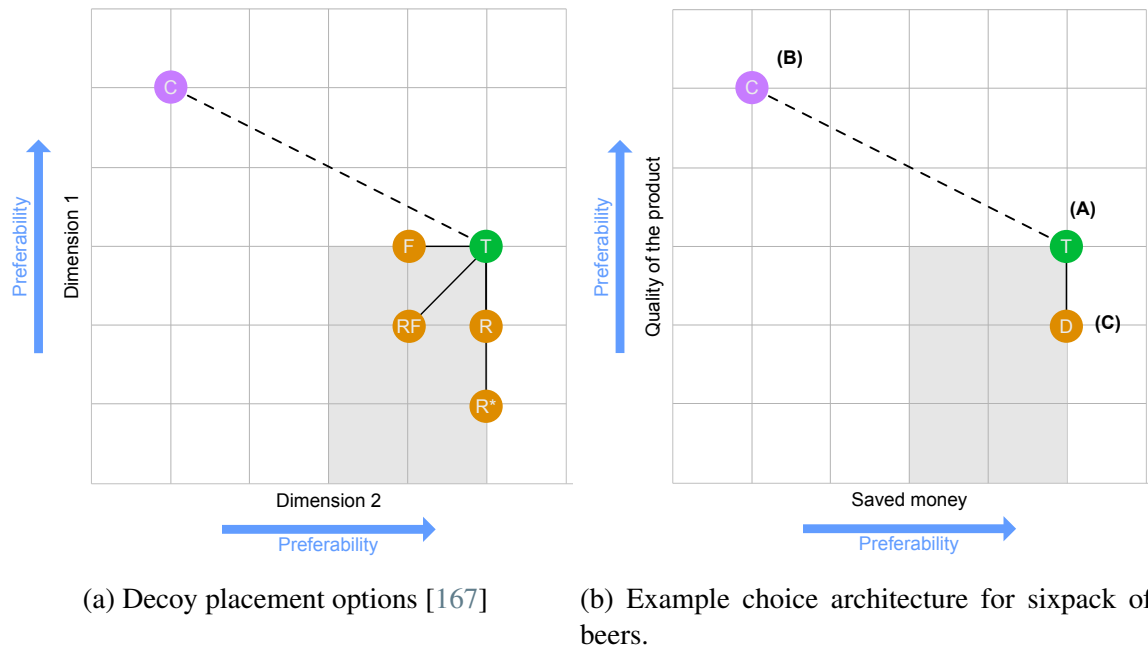
Option	Price	Quality rating
(A)	\$4	50 /100
(B)	\$6	70 /100

While option (A) is cheaper than (B), it is also lower in quality. Spending \$2 more, the buyer will get a higher-quality beer (B). With the information available at this point, it is hard to decide. Buyers may have a general preference for either lower price, or higher quality if all other factors are excluded. Imagine the vendor wants to sell more of beer (A) because margins are higher for that product. To achieve this, Huber et al. explored different ways of adding a third option:

Option	Price	Quality rating
(A)	\$4	50 /100
(B)	\$6	70 /100
(C)	\$4	40 /100

Product (C) is as expensive as (A) but falls behind in quality. Thus, buyers will make a “better deal” if they choose option (A) by comparison with (C). Options (B) and (C) are more difficult to compare because both dimensions (price and quality) are higher in (B) and make it appear like an outlier. Option (C) is thus called the **decoy** that is *dominated* by option (A) (**target**). Option (B) (the **competitor**) also beats the decoy, but the comparability between the target and the decoy boosts the favorability of the target. This is exactly what Huber et al. found, and which was later confirmed numerous times in different decision-making scenarios [14]. Adding the decoy has the potential to reverse existing preferences, making it a powerful tool for marketing and sales. The decoy itself often serves the sole purpose of making another item stand out, i.e. the vendor does not intend to sell the decoy item. Thus, the underlying persuasion principles are **salience** and **framing**. Constructing the dimensions in a certain way is referred to as *choice architecture* [333], and is an important persuasive design strategy.

Huber et al. stated that there is a specific combination of attributes to position the decoy (see Figure 10.1a). Depending on the target, the decoy acts as range increasing (R, R\*), frequency increasing (F), or both (RF). By extending the range on the dimension on which the *competitor* is superior, the fixed difference between all items on that dimension is weighed less. Figure 10.1b illustrates this type of decoy placement for the sixpack example. The range of the quality is clearer after the decoy is added: before, the span was 50-70, and afterward 40-70. The superiority of the competitor appears less significant. Hence, the higher price can appear unjustified.

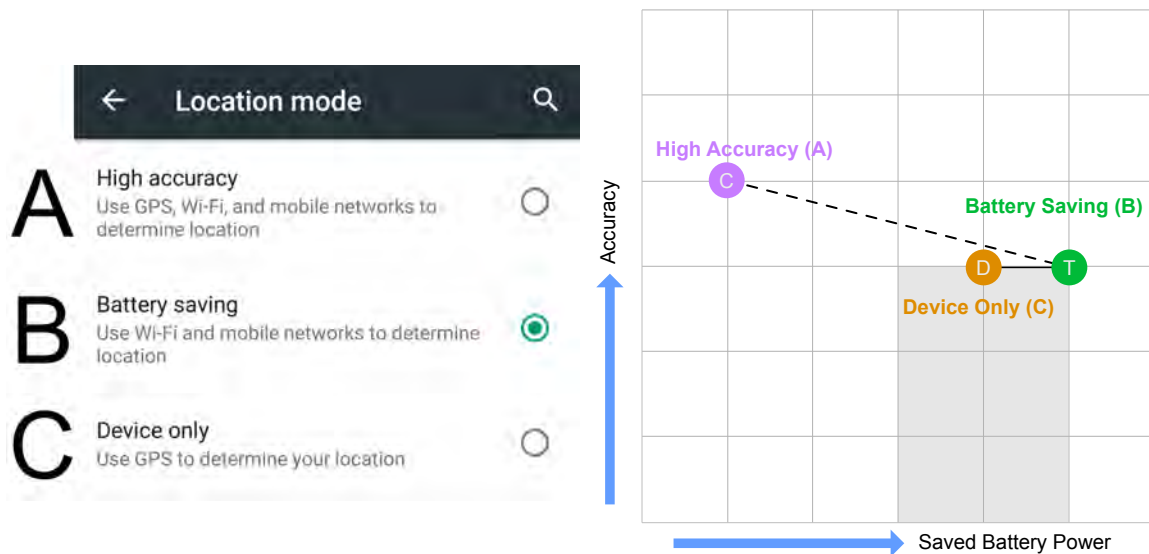


**Figure 10.1:** General choice architecture overview to generate asymmetric dominance. C = competitor, T = Target. The decoy can be placed at different positions in the spectrum but needs to be dominated by the target (gray area). Placement strategies for the decoy to increase the target’s dominance: R = Range, R\* = extreme range, F = frequency, RF = range and frequency.

### Example: Decoy Effects in the Real World

Huber et al.’s example has often been adapted and used in the design of user interfaces. The vendor, i.e. a service provider, tries to steer users towards a particular direction that they see as favorable. For instance, location settings on an Android device show signs of a decoy pattern (see Figure 10.2). From bottom to top, the “device only” mode activates the GPS module to determine the device’s location. It is thus highly accurate outside of buildings but does not work well inside. On the other hand, the “battery saving mode” works in both environments by using Google’s online location services based on triangulation between network cells and surrounding WiFis. In urban areas and even inside buildings, it can achieve great accuracy, while it only roughly estimates locations in rural areas. The two accuracy-levels are thus comparable, but the “battery saving mode” does not require powering up additional antennae and modules, giving it a graspable advantage. The “high accuracy” mode combines both approaches and is thus the most battery consuming, but also the most versatile option. “Battery saving” can be seen as the target, because it provides the best trade-off in most situations. The “device only” mode is the decoy because it uses more battery, while the “high accuracy” mode is the competitor. Google requires the user to allow the collection of technical sensor data for the “high accuracy” and “battery saving” mode. Battery consumption is likely more important to users than location accuracy, which further suggests that the “battery saving” mode is indeed the targeted setting.



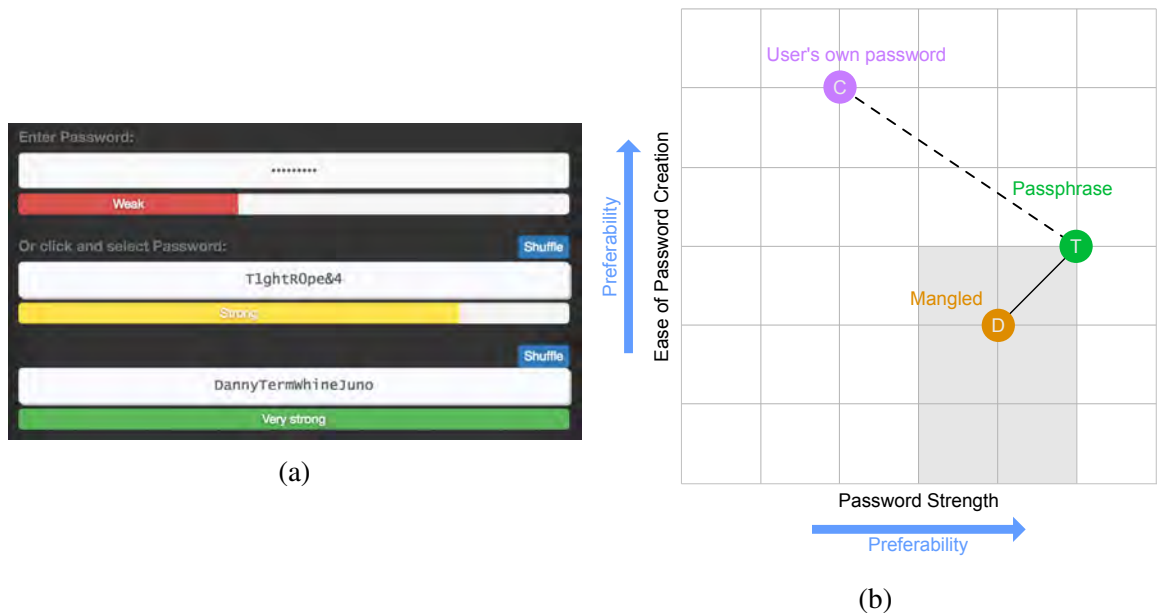


**Figure 10.2:** Location settings in Android show signs of a decoy pattern. The “battery saving” mode is targeted because it does not activate the GPS module and can achieve comparable accuracy. Google benefits from collecting information on WiFi hotspots and network cells to improve their location service.

### 10.1.3 Choice Architecture in Security and Privacy

The USEC community has started to investigate the feasibility of behavioral economics principles in the design of security and privacy mitigations. Egelman et al. showed that choice architecture is highly relevant for privacy settings on mobile devices [93]. They explored how users value privacy-respecting apps and how they make decisions from a list of applications. To measure preferences, they put monetary values and discounts on permissions like accessing the Internet or using device location. Participants in their study showed clear decision-making patterns that were influenced by the price and type of permission. Therefore, Egelman et al. concluded that a certain choice architecture can guide users towards a more “rational behavior”.

Knijnenburg et al. [196] explored the *choice proliferation* phenomenon in location-privacy settings. The principle indicates that people become choice averse with an increasing number of options. In their study, they observed that participants were strongly influenced by the number of available options to share their location. Without specifically mentioning it, they also used a *decoy* option that was extremely unfavorable but triggered a change in preference. In another study, Korff and Böhme showed that the granularity of privacy settings on a business social network can have similar effects: participants in their study tended to stick with default settings and were also less satisfied with their choices if there were too many options [201]. Acquisiti et al. showed that such architectures can nudge users towards certain settings [1].



**Figure 10.3:** Choice architecture and decoy placement as evaluated in the online experiment.

Regarding passwords, only a few publications mention the use of choice architectures. Renaud et al. evaluated a wide range of nudges to make users of a university platform pick a stronger password. They concluded that the tested architectures were fairly ineffective in achieving this goal, but there might have been other more effective solutions beyond their designs. Before our investigation, no study that we are aware of has explored the decoy paradigm for passwords.

## 10.2 Designing Password Choice Architecture

We aimed to craft a nudge that persuades users to create stronger passwords. Respecting the principle of the opportune moment, we opted for account creation contexts. To emulate a situation that is comparable to buying one out of several product alternatives, we show suggestions beneath the password field where the user enters their choice (see Figure 10.3). We ended up with this design after identifying opportunities for strength feedback on the user's password and on the suggestions (i.e. feed-forward). To that end, we had created several prototypical choice architectures and rapidly evaluated them in the lab and online (see technical report [286]). The final architecture implements the propositions in Huber et al.'s framework (see Figure 10.3b).

The key to password choice architecture is the user's own password. It has to be seen as the **competitor**. Service providers will want to accept the user's own password to make sure users sign up. On the other hand, they want to avoid attacks on user accounts, which is why

they likely encourage using a stronger password. This is the **target**. In our design, we opted for a *passphrase* as the target for several reasons: only a few users create secure passphrases as their primary credentials [345]; passphrases provide usability and memorability benefits [189]; self-selected passphrases often do not provide the desired security benefits [37], which can be mitigated by supplying users with a combination of random words [297].

To make the passphrase preferable over the competitor, a decoy needs to be carefully positioned along the two dimensions such that it is closer to the target than to the competitor. Moreover, it needs to be dominated at least in one dimension by the target. To achieve this, we identified “ease of use” and “security benefits” as two feasible dimensions of the choice architecture. We opted for a range-frequency decoy (see Figure 10.1a) where the target dominates the decoy in *both* dimensions because effects are more likely to be detected this way. A typical-length *mangled* password fulfills the criteria. The richer character set includes symbols that require the Shift/Alt keys to enter them. Thus, the ease of creation is lower than that of a passphrase (dimension 1). At the same time, typical mangling strategies only slightly contribute to password strength, which we confirmed in Chapter 5. Passphrases are considered stronger by several estimators. Consequently, mangled passwords are dominated by passphrases regarding strength (dimension 2). Figure 10.3b illustrates the positioning along the two dimensions.

## 10.3 Quantitative Evaluation

We ran an online experiment to evaluate the efficacy of our decoy choice architecture for passwords. We formed the following hypotheses about the outcome:

- H0** Participants’ self-selected passwords are comparable in strength and memorability, regardless of the presence of a password suggestion (Null hypothesis).
- H1** The presence of a single password suggestion will lead to slightly stronger passwords.
- H2** The presence of two suggested passwords that follow a decoy choice architecture regarding strength and ease of use will lead to stronger passwords that resemble the target suggestion.

### 10.3.1 Method

The study implemented a between groups design. The main task was creating a new password under one of four treatments. “Suggestion architecture” served as the independent variable with three levels: **Passphrase**, **Mangled**, and **Decoy**. Each level was tested with a separate participant group. In the Passphrase condition, participants were suggested a single passphrase consisting of four words. Analogously, a higher-complexity password is displayed in the Mangled condition. The Decoy group received both the target passphrase and

---

the complex decoy password as alternative suggestions to their own password. Including study conditions with single suggestions allowed us to compare different choice architectures and measure the impact of the decoy effect. All suggestions are accompanied by visual and textual representations of zxcvbn scores (see Figure 10.3b). The labels for the different strength scores were *very weak*, *weak*, *ok*, *strong*, *very strong*. To obtain a baseline for comparison, there were no password suggestions for the **Control** group. However, participants in this group also received the same kind of strength feedback, so we could isolate the impact of adding suggestions, i.e. feedforward.

We did not collect plaintext passwords to ethically deal with participants disclosing their real passwords. Instead, we modified the zxcvbn library by stripping all sensitive information<sup>1</sup>, and analyzed participants' passwords on the fly. These *meta statistics* served as dependent variables and were saved to a database. Most notably, they describe the password topology (length, number of upper-/lowercase, digits, symbols, and chunks), estimated guess numbers and the zxcvbn score. Scores range from 0 (weak) to 4 (strong), while guess number estimates are open-end. Another dependent variable was memorability which was measured by a successful authentication three days after password selection. To achieve this, we hashed passwords with a secure one-way function (PHP's `password_hash()`) and compared hashes afterward. If they matched, the password was correct. Thus, this is a binary metric.

## Prototype

For the study, we implemented the concept as web-prototype based on HTML and JavaScript. As a foundation for generating passwords in all conditions, we relied on the Diceware dictionary<sup>2</sup> consisting of 5823 words. It provides a good spectrum of common and uncommon words of varying length. To generate the passphrase (target), four words were randomly combined, and each word was capitalized. As shown in Chapter 5, zxcvbn rates four-word passwords with its highest score. The randomness of generated passwords allows us to calculate their entropy. Each word has an estimated entropy of  $(\log_2(5823) \approx 12 \text{ bits})$ . Combining four words randomly thus results in a total entropy of approximately  $(2^{12})^4 = 48$  bits for passphrases.

For the decoy and the mangled password condition, the prototype modifies a randomly selected word from a smaller subset of the Diceware list. The subset only includes words longer than eight characters, giving a total of 687 candidates. The first letter of the word and a second randomly chosen character are capitalized. Two letters are substituted by similarly-looking digits to inspire a *l33t* character. For instance, the letter "o" was replaced with the digit "0". Finally, a random symbol and digit are appended to the word. The resulting decoy password consists of four different character classes (LUDS).

---

<sup>1</sup> <https://gist.github.com/TobiasSeitz/e27a867535b82f6cf9a6ae6140da8b81> (last accessed 06.03.2018)

<sup>2</sup> <http://world.std.com/~reinhold/diceware.wordlist.asc> (last accessed 06.03.2018)

We anticipated that the generated passwords are exceptionally unattractive in some cases, e.g. if passphrases include uncommon, hardly memorable words, e.g. *GirthInflxThineAegis*. To make them more appealing, the prototype provides a *shuffle* button that creates a new combination of different words (see Figure 10.3a). The mangled password can also be re-generated. Another feature to lower the barriers to take one of the suggested passwords was the opportunity to transfer it to the password field with a single click. However, to facilitate memorization, participants needed to manually type the password into the confirmation field.

## **Procedure**

The experiment consisted of two parts that were carried out on separate days. The initial step included the password selection task and usability assessment among other qualitative metrics. Participants were invited to return for the second part of the study three days later. This follow-up step included memorability assessment and further qualitative feedback to help us interpret the data.

The first part started with a thorough briefing about the collected data and asked for consent. The same web-page introduced the scenario for the task: Participants were asked to imagine creating a new password for their already existing email account under a basic8 policy. The page displayed the password fields and suggestions. Participants were randomly assigned to one of the four treatment groups, so the type and number of suggestions depended on this assignment. Once the password was successfully confirmed, participants were asked to fill out a brief questionnaire mostly consisting of 5-point scale items on attitudes and password behaviors. At the end of the first part, the web-page displayed a confirmation code that participants had to copy over to the prolific platform to mark the survey as done.

Three days later, we invited the participants to return and complete the second part. They were asked to provide the password they had remembered, but if they failed, they could continue anyhow.

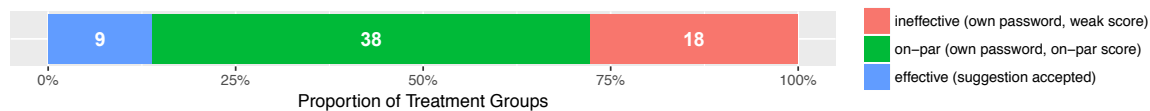
## **Recruitment and Sample**

Following best practices in password research, we leveraged crowd-sourcing tools to elicit the data. Recruitment took place through the research platform Prolific<sup>3</sup>, which is comparable to Amazon's Mechanical Turk solution. Participants were screened for age (older than 18 years), and they had to be located in either the UK or in the USA. The region restriction was introduced because Prolific has a larger user base in those countries. To ensure the quality of the data, we required a past survey completion rate of at least 95%. Such rates are a common metric for the reliability of a crowd-worker [271].

From the 106 respondents who started the experiment, we had to reject seven because the study completion code was wrong or missing, because the questionnaire was insufficiently filled out, or completion times were outliers ( $> 3 * SD$ ). The remaining 99 participants

---

<sup>3</sup> <https://prolific.ac> (last accessed 06.03.2018)



**Figure 10.4:** Nudge efficacy in treatment groups (n=65)

received invitations to return, which 97 people did. There were another fourteen incomplete responses or wrong study codes. Hence, the resulting  $N$  of our data set is  $N = 83$  valid, complete responses in both parts. Group sizes were  $n = 18$  in Control,  $n = 24$  in Passphrase,  $n = 21$  in Mangled, and  $n = 20$  in Decoy. On average, participants were 30 years old ( $SD = 10$ ). 42% were female. The majority (78%) was employed, 12% were students, 10% were unemployed. Participants reportedly possessed nine online accounts that they regularly use ( $SD = 5.6$ ). Thus, the sample provides a sufficient degree of diversity.

## 10.3.2 Results

Through statistical tests, we found significant differences across the groups. Although the decoy effect did not work as intended, there were notable side-effects. In the following, we break down the findings. For non-parametric continuous data, we used Kruskal-Wallis and Mann-Whitney tests, whereas frequencies were analyzed with chi-squared tests. Significance levels were set to  $\alpha = 0.05$  unless multiple comparisons required a Bonferroni correction.

### Efficacy of Suggestions

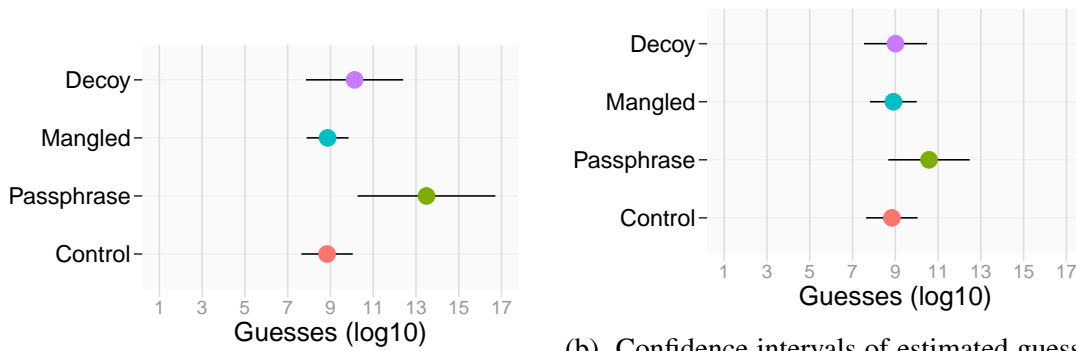
Across all treatment groups, nine respondents out of 65 ( $\approx 14\%$ ) accepted a suggested password verbatim – four in the Passphrase group, two in the Mangled group, and three in the Decoy group (two targets, one decoy). Suggestions were thus effective for one in ten participants. From the remaining 56 self-selected passwords, 18 (27.7%) were weaker than the suggestions. In other words, the nudge was ineffective for roughly a third of participants judged by their choice to pick a password that was weaker than the suggestion. For the remaining 38 participants (58.4%), the nudge could have influenced them to create passwords that are as strong as the suggestions, but we lack data to investigate this. Figure 10.4 visualizes the efficacy of suggesting passwords.

### Impact on Password Strength

Table 10.1 lists descriptive statistics on password strength metrics. Taking the entire sample into account, omnibus tests did not show significant differences between the four groups. However, we plotted the confidence intervals of all metrics to examine the data visually. Here, it is visible that guess numbers did notably differ between the groups: the Passphrase treatment led to the strongest passwords overall (see Figure 10.5a). If we remove the samples

**Table 10.1:** Summaries of password metrics from the online experiment. Arranged by group (columns) and metric (rows)

	Control		Mangled		Passphrase		Decoy	
	M	SD	M	SD	M	SD	M	SD
length	11.33	3.53	11.8	2.74	13.87	3.8	11.9	2.69
score	2.88	1.02	2.9	0.76	3.29	0.9	2.95	0.88
guesses <sub>log10</sub>	8.84	2.41	8.86	2.15	13.48	7.63	10.12	4.85
digits	2.61	2.06	2.28	1.27	2.16	2.18	2.6	2.34
special	0.22	0.64	0.52	1.16	0.2	0.5	0.3	0.57
uppercase	1.77	0.8	1.42	0.59	2.45	2.35	1.75	1.11
lowercase	6.55	3.91	7.38	3.21	8.91	4.09	6.95	3.42



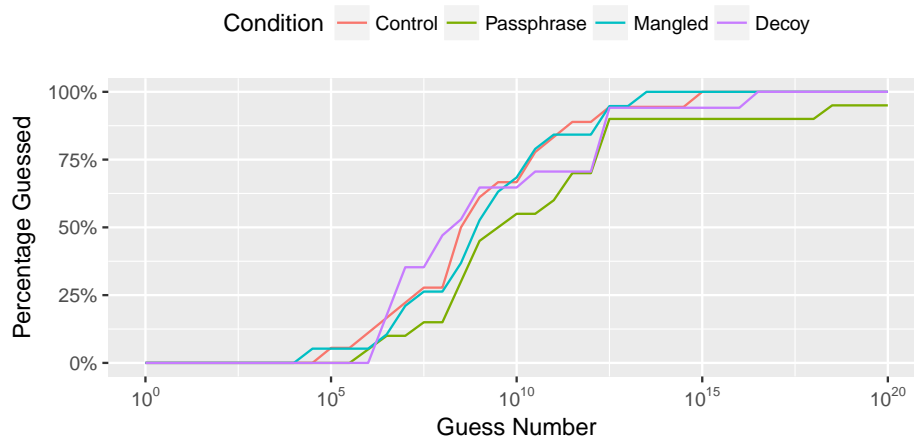
(a) Confidence intervals of estimated guess-numbers (log 10) for all participants (N=83). On average, the Passphrase group created significantly stronger passwords than the Control and Mangled group.

(b) Confidence intervals of estimated guess-numbers (log 10) for self-selected passwords (N=74). Although the difference between the Passphrase group and the others is not as big as in the overall sample, the average guess numbers are still  $\approx$  two orders of magnitude apart.

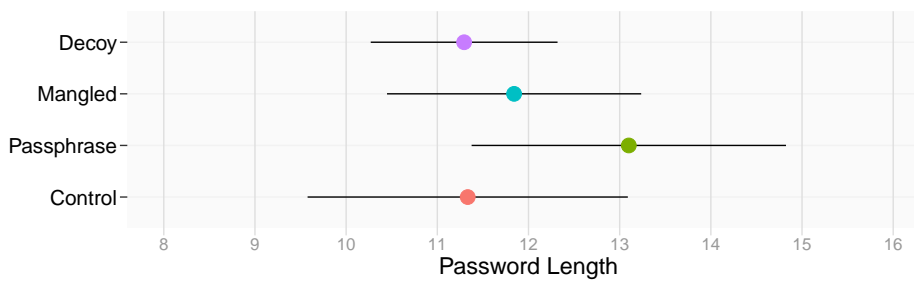
**Figure 10.5:** Arithmetic mean of guess numbers (log 10) and 95% confidence intervals.

where a suggestion was accepted, the differences become smaller (see Figure 10.5b). Since guess numbers are shown on a logarithmic scale, the difference might appear smaller than it actually is: guess numbers in the passphrases group were about 50 times higher than in the control and mangled groups. This is also visible in Figure 10.6, where the percentage of cracked passwords is consistently smaller than in the other groups. Fitting a smoothed generalized additive model to the cracked percentages confirms that passwords are less likely to be cracked after any given number of guesses if they were created in the Passphrase condition ( $intercept = Control, B = -7.9\%, p < 0.001, R^2_{adj} = 0.99$ ). The rise in strength most likely is due to increased password length in the Passphrase group, see Figure 10.7.





**Figure 10.6:** Estimated guessability of user-selected passwords across the four conditions.



**Figure 10.7:** Average password length of self-selected passwords and 95% confidence intervals

**Table 10.2:** Policy fulfillment of submitted passwords. Most participants used at least three character classes.

	comp8	3class8	3class12	3class16	basic8	basic12	basic16
Control	2	9	4	0	1	1	1
Mangled	6	7	3	3	1	1	0
Passphrase	4	5	4	2	1	1	7
Decoy	5	7	3	1	1	1	2
Σ	17	28	14	6	4	4	10



## Password Composition and Policy Adherence

It is possible to sort participants' self-selected passwords into "policy buckets", i.e. the most stringent policy they would fulfill, which makes the effort to create a stronger password visible. Table 10.2 shows the distribution of this analysis in commonly used policy taxonomy [299]. A chi-squared test showed no significant differences across the groups ( $\chi^2(18) = 16.93, p > 0.5$ ). Yet, it is interesting that even though only a basic8 policy was enforced, all participants formed passwords with at least two character classes. The majority (78%) even used three character classes. Participants in the Mangled condition were about twice as likely to adhere to one of the complex policies (comp8, 3class12, 3class16) than the Control group. On average, the length requirement was exceeded by  $\approx 4$  characters.

## Memorability

Roughly 40% ( $n = 34$ ) of participants succeeded to provide their password from the first part of the study. A chi-squared test on group differences was not significant ( $\chi^2(3) = 3.84, p > 0.05$ ). Among the participants whose passwords matched three days later, 76% ( $n = 26$ ) reported to have memorized it, while the rest had their browser store it (2), or put it into a digital file (5), or wrote it down on paper (1). Participants who had opted for a suggestion were mostly unable to authenticate. The one respondent in the Decoy group who correctly entered the mangled password in the second part did so reportedly from memory.

## Qualitative Findings & Feedback

A thematic co-analysis of the qualitative data from the exit questionnaires showed that there were no observable differences across groups. Therefore, we do not report group-specific details but rather the general picture.

In the three treatment groups, we elicited data on the subjective perceptions of the suggestions. Participants were shown adjectives from which they could select multiple items and provide additional text. The most common reactions were "neutral" ( $n = 25$ ), "surprised" ( $n = 23$ ) and "pleased" ( $n = 11$ ). There was no observable tendency as to perceived security improvements for email accounts by the suggestions. 20 participants (24%) might be annoyed if their main email provider implemented a sign-up form like the one in the study, but they did not clarify this further. 30 respondents (36%) acknowledged that suggestions could facilitate email account creation. The main reason to decline the suggestions was the need for passwords with personal meanings (43% agreement rate). Besides, P81 said that "the main reason I don't use generated passwords, is that if a hacker finds out how they're generated, they've basically figured everyone's passwords out. Then its only a matter of time before they brute force into accounts using the same method the system uses to generate them." This quote highlights the distrust regarding password generators, while at the same time assuming attackers are incapable of predicting human behavior. Finally, there were interesting qualitative statements about the memorability of passwords (all *sic*): "sorry, i would have written it down if i knew there was a second part to the study. thats what i do for passwords that i will use again, until i memorize it, then i throw away the paper i write

---

it on.” (P33, *sic*) “I’m not sure if it was because I don’t type this password I used here often or if it [is] because I used one of my lesser very weak passwords that I failed to remember it quickly. Regardless, I need a few better habits. ” (P26) “I forgot my password as I didn’t think I’d need it again. Had I known I would likely have tried harder to remember it.” (P47). From these statements, we take away three important points: a) Some participants were unaware that they would need the password in the future although they received according instructions, b) they rely on their own memorization techniques, and c) they feel guilty of bad password habits which echoes the results in Chapter 8. In summary, those factors had contributed to the low memorability rates of the study passwords.

### 10.3.3 Limitations

There are a few important aspects to consider in the interpretability of our study findings. The sampling method, although state of the art, still targets users who are open-minded enough to participate in studies. Only those crowd-workers with a successful track record were screened in, thus the sample might not be representative for the entire population. However, rough trends can be seen in any case. In terms of sample size, we had initially hoped for a larger data set, but had to omit a significant proportion due to low quality of the data. This reduced the achievable statistical power, which prevented us to narrow down confidence intervals and accordingly find significant effects. This limitation is aggravated by the fact that many study participants chose passwords that were much stronger than what is to be expected from real-world passwords [226]. Such behavior reduces ecological validity and shrinks potential effect sizes induced by different nudges. Despite these limitations, it is astonishing to still detect notable differences across groups. Finally, we relied on the zxcvbn strength metric, as in many other studies throughout this thesis. We could have collected plain-text passwords and used a more robust approach like PGS [346]. However, this would have been too risky, because participants might have disclosed their real email passwords in the scenario, which we regard as unethical. Therefore, the zxcvbn metrics are one of the best solutions to work around this issue while providing sufficient robustness.

## 10.4 Discussion

In the following, implications and opportunities from using persuasive suggestions are highlighted.

### 10.4.1 The Ineffectiveness of the Decoy Effect

The data did not provide any indication that the presence of an additional decoy-suggestion influenced participants. Therefore, we **reject H2** and conclude that the decoy choice architecture was not effective. There are several explanations about the failure of the architecture.

First, in order to induce a decoy effect, the two preferability dimensions (cf. Figure 10.1a) need to be salient and clear, e.g. the price and quality of an item. In our case, the dimensions were *ease of password creation* and *strength*. While strength was perhaps effectively conveyed, the user interface did not explicitly highlight the usability of the password. Advantages in typing speed and memorability over the mangled password were probably to **vague or unclear**. The persuasive strategy failed to trigger deliberate thought processes (cf. System 2), and intuitive thinking (System 1) might be responsible for participants' preference to stick to their self-selected passwords.

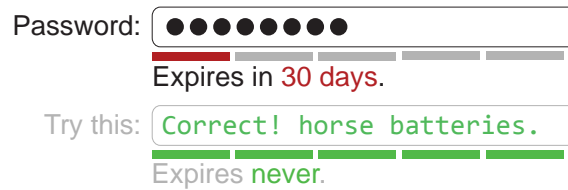
Moreover, Huber et al. measured the decoy effect in a different study design. They relied on repeated measures to understand preferential shifts induced by the presence – or absence – of the decoy. For password studies, repeated measures are only seldom a good fit, because users are unlikely to show different behaviors if they create passwords multiple times in a row. To get closer to the original study design, a real-world field test is conceivable. When users create an account, the decoy might be present and it is missing when they reset the password at any later point in time. As a caveat, this approach vastly increases the duration of the study.

Finally, the participants' self-selected passwords were unexpectedly strong. This did not provide enough room to nudge them towards an even stronger alternative, because the “price” that it entails is unreasonably high. As a consequence, the target group for future studies in password suggestions should be narrowed down to people who are less likely to create a strong password. For instance, we know that the passwords that users select as teenagers tend to be weak [354]. At the same time, these are reused very often and are in use for many years. Thus, teenagers make for an interesting target user group to investigate the efficacy and design of nudges.

## 10.4.2 Framing the Goal Effectively

Although the decoy effect did not prevail, we observed that participants who were suggested a single passphrase as an alternative to their own passwords ended up creating longer and stronger passwords – even if they rejected the suggestion. Thus, **H1 is supported**. What makes this finding particularly interesting is the fact that a mangled password failed to produce a notable impact on password strength.

This is a potential consequence of latent mental models. As we have seen in Chapter 5, users have a rather robust mental model of a strong password. There is much evidence that users see the addition of digits and symbols as highly advantageous for password strength. In this study, the mangled password, however, did not receive the *best* strength rating but was only “second best”. The yellow color of the strength meter might have added to the discouragement because it is not on the positively connoted “green” spectrum. This might have already influenced participants' mental model about mangling strategies, so they proceeded with what they intended to pick in the first place. On the contrary, the passphrase *did* receive the best rating and its strength bar was colored green. Thus, the group who saw only this



**Figure 10.8:** Suggestion accompanied by a graspable benefit.

type of suggestion might have realized that it is **easy to achieve** the highest rating simply by making passwords longer, or that it is possible even without digits and symbols. Put simply, the passphrase seemed to encourage participants to “go higher” because the goal was within reach. *Goal gradient effects*, as discussed in Section 3.4.1, support this argument. Putting both suggestions side-by-side obviously negated these effects. The cognitive load might just be too high in that case.

### 10.4.3 Application Areas and Prospective Designs

Participants in our study indicated that their preferred way to handle passwords is memorization, echoing many other larger studies, e.g. [64, 247]. Thus, users cannot be expected to create a unique strong password for every account. Using a persuasive suggestion is thus only reasonable in a limited number of situations. For instance, users often categorize passwords by their importance (see Chapter 8, [316, 363]). For **important accounts** they are willing to pick stronger passwords, so providing a good alternative like in our study could prove helpful. To make the strength benefits even more graspable, the service could reward complying behavior with longer expiration intervals (in case there are sufficient reasons to use such measures). For instance, a suggestion can illustrate that stronger passwords do not expire (see Figure 10.8). We put this idea forward in [291] and it was very recently evaluated by Renaud and Zimmermann in a longitudinal study [266]. Their findings conclusively show that **making benefits graspable** is the primary key to effectively nudge users towards stronger passwords. A special application area for graspable benefits is password managers. Users easily recognize that a **master password** for password managers need to meet higher strength standards because it constitutes a single point of failure in protecting many accounts. The benefit is therefore graspable and can be made salient very easily. Thus, creating a master password is an opportune moment in which users are probably more receptive to recommendations.

## 10.5 Conclusion

We set out to nudge users towards a password that is stronger than the first choice from their “comfort zone”. From theories in behavioral economics, we carefully derived a choice

architecture that translates marketing strategies to “password advertisement”. This direct translation, however, did not produce the envisioned decoy effects in our online study with 83 participants. Instead, we found that encouraging people through an easily achievable goal fostered a positive outcome: If participants saw that a passphrase containing no digits and symbols could be “very strong” they managed to create stronger passwords on average. We conclude that simplifying the sub-task to compare one’s own performance to a suggested alternative is a key to successfully nudging users in this direction. To avoid that such nudges wear off over time, they should only be used in situations where the benefit of choosing a stronger password is graspable and evident to users.

Although the decoy effect did not show, this does not mean it does not exist for passwords. It could simply mean that the direct adaptation of choice architectures in other fields is insufficient and needs more fine-tuning. In a different UI design, where the dimensions are more clearly framed for the users, the hypothesized preferential shift might be induced. A possible direction towards this is to refrain from generating completely new passwords and instead nudge users towards a modified version that significantly increases strength. Ur et al. have already provided first positive results about suggesting password alterations. It would be interesting to find out if an additional alteration that is “unattractive by design” could boost adoption rates of the target suggestion. We see many opportunities to further explore persuasive password suggestions, especially if they remove usability pain points of trying to create a memorable yet strong secret.

### Take Aways

- The decoy effect was **not** visible when participants received two password suggestions. They proceeded with creating their own passwords. The most plausible explanation is that the benefits of the target over the decoy were too vague to overcome the inertia to follow the recommendation.
- Demonstrating that strong passwords do not have to be overly complicated by suggesting a passphrase made our participants select stronger passwords. Using feed-forward in conjunction with feedback appears to encourage people to come up with a better secret.



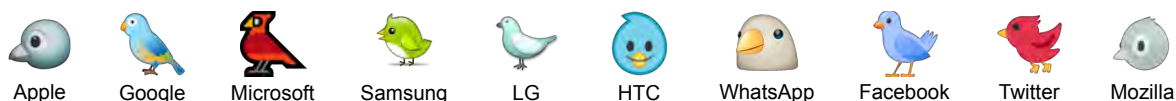
## Extending the Password Space with Emojis

Our previous efforts to support users focused on suggestions, feedback, and feed-forward to help users create stronger passwords. We now continue to explore a different dimension of persuasive design that was supposed to make it easier for users to create both strong and highly memorable passwords. To that end, we looked at **emojis** as possible solution, since we have found positive attitudes towards using them inside passwords (see Chapter 7). Emojis are *pictographs* used to express emotions or communicate words with pictures. The Unicode consortium has made them part of the de-facto digital encoding standard<sup>1</sup>, which led to widespread adoption very quickly. In 2015, the “face with tears of joy” emoji (😄) became the Oxford dictionary’s *word of the year*<sup>2</sup>, which shows the cultural impact of these colorful symbols. A recent report claims that roughly three quarters of Internet users communicate with emojis on a regular basis [100]. While image-based passwords were proposed decades ago (see Section 2.5.1), the recent success of emojis has given graphical authentication new momentum.

Emojis are an opportunity to increase security, because they can add more complexity to passwords. Adding an emoji to a weak password like `iloveyou` could already make it less predictable, e.g. `iloveyou😄`. We have also seen that users’ mental models are built around complexity, and not necessarily around length. Thus, an emoji-password like `Corr3ct🏠📁🔑` might be perceived as more secure than the original passphrase. Past studies have also shown that graphical authentication can benefit from the *picture superiority effect*. Hence, using emojis inside passwords might have usability advantages, and we also see them as an **enabling technology** that empowers users to pick more creative and memorable secrets, i.e., a persuasive design strategy. At the same time, existing authentication back-ends do not require significant changes to make emoji-passwords work. Since emojis are encoded as regular characters, the only potential change is to use a different version of UTF. Theoretically, 90% of web-pages are compatible with emojis in password fields already now (see

<sup>1</sup> <http://www.unicode.org/reports/tr51/> (last accessed 10.03.2018)

<sup>2</sup> <http://blog.oxforddictionaries.com/2015/11/word-of-the-year-2015-emoji/> (last accessed 10.03.2018)



**Figure 11.1:** Although the “bird” emoji is encoded with the same unicode character (U+1F426), its visual representation differs strongly across platforms and vendors. This has specific ramifications on the use of emojis inside passwords, which we address in this project.

Footnote 1). Consequently, some service providers have started to allow emojis as part of user-chosen passwords also on the back-end. Twitter, Slack and StackOverflow are among these pioneers.

However, the adoption comes with certain risks. Emojis are primarily used on mobile devices [100]. Hence, adding emojis to passwords on a smartphone is easy, but there might be problems when the user tries to enter their emoji-password on a desktop: Typical input solutions possible with *software* keyboards get left out on the desktop. Some applications have enabled indirect emoji input through the mouse in a graphical user interface, so it is not impossible to enter an emoji-password across different platforms. On the other hand, the specific images used to render emojis are platform-dependent: a bird emoji on a native iOS soft-keyboard looks different on Android (see Figure 11.1). This *fragmentation*<sup>3</sup> introduces new issues not typically found with other graphical authentication schemes, where there is more control over the images. Thus, we wanted to understand the usability factors and constraints of emoji-passwords to gauge the feasibility of this authentication mechanism.

In this chapter we explore pragmatic and hedonic qualities of emoji-passwords. We aim to answer the following research questions:

RQ1 Do emojis in passwords help users create more **memorable** passwords?

RQ2 How do platform-dependent renderings (**fragmentation**) affect memorability?

RQ3 What is the most feasible **user interface** to enter emoji-passwords?

RQ4 What are users’ password **selection strategies** and behaviors?

RQ5 What are users’ general **attitudes** towards using emojis inside passwords?

This chapter reports on a mixed-method experiment carried out to investigate the usability of alphanumeric passwords that contain emojis. In the following, we put the design space for emoji-passwords into context, describe an empirical research study, and derive implications on the effective use of emoji-passwords.

---

<sup>3</sup> <https://blog.emojipedia.org/2018-the-year-of-emoji-convergence/> (last accessed 16.02.2018)



## Publication Statement

Beside myself, Florian Mathis and Heinrich Hussmann made contributions to the project. The results of the study were published at OzCHI 2017 [290]. The chapter elaborates on the results of that paper and provides significant additional analyses:

- We show a detailed quantitative breakdown of selection strategies, in particular regarding the position of the emojis and relations to the passwords.
- We take a deep-dive into successful and unsuccessful authentication events in the second part of the study.
- We provide new thoughts on the ramifications of using emojis inside alphanumeric passwords.

## 11.1 Background and Context

### 11.1.1 Emojis

Literally translated from Japanese, emojis<sup>4</sup> are “picture words” [325]. They are primarily used in mobile messaging applications like WhatsApp, iMessage, and alike, to express emotions and add subtext in messages. Before emojis arrived, this has been possible with *emoticons*, i.e. combinations of regular characters that convey emotions like :-), <3 and :-0. However, emojis are visually richer and much more versatile. The current version of the Unicode standard lists around 2800 emojis (see Footnote 7) categorized by topics, and roughly 150 are added each year. In 2015, emojis saw a stark increase in usage numbers. Both iOS and Android added special software keyboards that allowed users to easily enter emojis through direct touch input. Often, users replace entire words or blocks of text with emojis. Although emojis are useful to overcome language barriers, their meaning is not always clear [234, 335]. Researchers have started to investigate misinterpretation in many ways, and have proposed solutions like a “disambiguation API” [375].

### 11.1.2 Emojis in Authentication

The growing adoption rates of emojis in various application areas has not gone unnoticed by the USEC community, who started trying to improve authentication schemes with emojis.

<sup>4</sup> The plural form for “emoji” is both “emoji” (Japanese) and “emojis” (adopted to English). For clarity reasons the chapter sticks to “emojis”.

---

One of the first solutions was presented by Intelligent Environments<sup>5</sup>. Their *emoji-passcode* system would allow customers to select four emojis from a 9x5 grid to log into mobile banking apps. However, the concept has not been widely adopted. Golla et al., respectively Kraus et al., proposed a similar system that was targeted at screen-locks for mobiles [136, 203]. In essence, their EmojiAuth system replaces each digit on a PIN-pad with an emoji. In two user studies, they evaluated the concept and showed that both security and usability of unlock mechanisms can be improved this way. On the other hand, we were able to find only one publication about using emojis as part of an alphanumeric password [9]. Al-Husainy and Mali focused on user-account passwords on desktop computers but did not present an empirical evaluation of the concept. We believe web pages and mobile applications are the primary use case for emoji-based authentication. This application area is still underexplored.

### 11.1.3 Research Opportunities

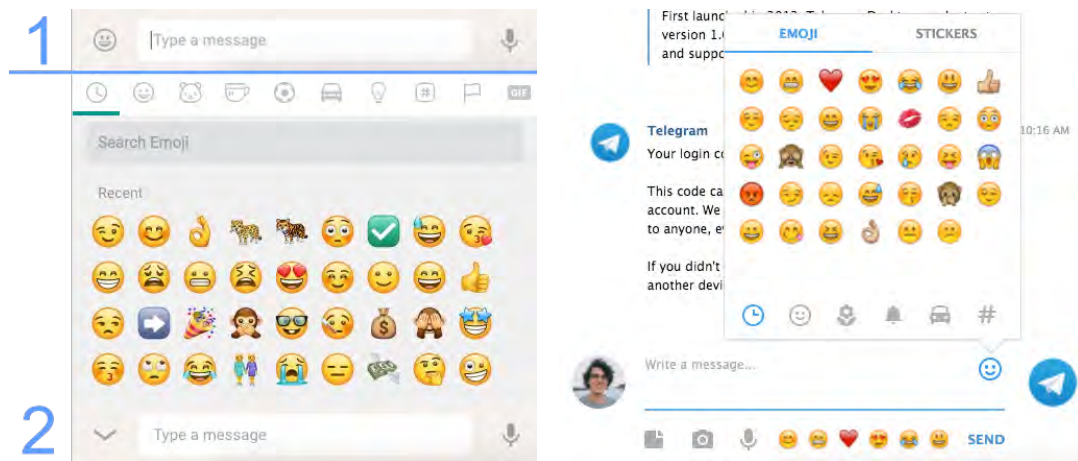
The combination of emojis and alphanumeric characters generates a wide range of research questions, and we can only answer a small subset of them. In particular, we explore password selection strategies that may hint at potentially risky behavior. It is likely that emoji-passwords share some traits of alphanumeric passwords. For instance, the topology of a password is an important metric to gauge its guessability. Therefore, we would expect to observe some topologies reported by Weir et al. [365], Ur et al. [345], and Kuo et al. [206]. The latter indicated that some users had already used emoticons inside passwords in 2005. Although we cannot realistically measure guessability of emoji-passwords at this time, we can explore emerging patterns. For instance, we can elicit where users put their emoji and how they choose it. Emoji-passwords are a hybrid between knowledge- and recognition-based authentication, so they demand effort from both the lexical and visual memory [265]. Therefore, this approach could turn out to be either a memorability advantage or disadvantage. Since the answer to this question is very open, we refrain from generating hypotheses altogether.

## 11.2 User Study

The goal of our project was to understand the human factors and usability constraints of emoji-passwords, which can be used to evaluate the idea of empowering users as persuasive strategy. To achieve this goal, we created a prototype that allowed entering emojis inside password-fields and evaluated it with a mixed-methods user study to cover a large spectrum of opportunities and caveats. The first part took place in a controlled lab environment, while the second was carried out remotely without moderation. To explore different dimensions of usability and to follow common practice, password selection and recall were spread out

---

<sup>5</sup> <https://www.intelligentenvironments.com/now-you-can-log-into-your-bank-using-emoji/> (last accessed 10.03.2018)



(a) Two-step point-and-click interface in WhatsApp. (b) Telegram shows the picker after the user hovers the emoji button. Moreover, recently used emojis are shown beneath the text-field.

**Figure 11.2:** Examples for point-and-click interfaces (“emoji picker”). *Progressive disclosure* is used to access the list of available emojis: The user first needs to interact with a control element (emoji button) or start typing a special character (mostly “:”), before an emoji can be selected by clicking or auto-completion of the shortcode. The emoji is then inserted into the text field.

across different days. In the following we describe the prototype, the procedure of the two methods, the corresponding variables, and the sample of our study.

### 11.2.1 Prototype

The project focused on using emoji-passwords on web sites, therefore we built a web-based prototype with standard technologies (PHP, HTML5, JavaScript). We identified two solutions to enter emojis on a desktop computer: via a point-and-click interface, and “shortcodes”. Most web-versions of messenger applications, e.g. WhatsApp Web, Telegram Web, Hangouts, etc., use the point-and-click approach (see Figure 11.2). A few communication tools also allow entering a predefined word that is then translated into an emoji. This *shortcode* often needs to be put into braces, e.g. (smile) on Skype, or stand between two colons, e.g. :smile: on Slack (see Figure 11.3). We implemented a prototype based on point-and-click selection and Slack-style shortcodes. However, shortcodes were not auto-completed, which is generally discouraged for passwords [232] and there was no “recent emoji” feature. After clicking an emoji, the prototype did not use the unicode character inside the password field for technical reasons<sup>6</sup>. Instead, the shortcode was automatically inserted and masked. To allow checking the entered password, it was displayed in plain text beneath the input fields (see Figure 11.4b).

<sup>6</sup> emojis typically break the masking of password fields, because their encoding differs in byte-size

---

While Unicode v11.0 contained 2789 emojis<sup>7</sup>, we reduced the number of available emojis to 50 for several reasons. First, we found through iterative testing that there was a selection bias, if the full range of emojis was offered; testers mostly included emojis from the first page. Second, Golla et al. used a similar approach for their EmojiAuth system [136, 203]. Last, reducing the number very likely facilitates recognition and potentially increases the memorability of the emoji-password.

The 50 emojis were selected with particular features in mind. To evaluate issues arising from similarity, around a third of emojis should mutually resemble each other. Fragmentation issues during authentication can only be seen for emojis whose appearance strongly differs on other platforms. Moreover, the emojis should appeal to users, e.g. because they are familiar with them. To achieve this, different emoji-categories should be considered. We identified 50 suitable candidates from the most-used<sup>8</sup> emojis on Twitter from multiple categories: *smileys & people* (14), *animals & nature* (7), *food and drink* (6), *activities* (5), *objects* (6), and *symbols* (7). We opted to include more smileys, because this roughly 50% of Unicode emoji characters fall into this category. Our emoji-picker randomly arranged the emojis in a 10x5 grid to isolate selection-by-position effects.

The prototype allowed to switch between two versions of emojis. The default version was based on the emojis from iOS 9.3 (see Figure 11.4a). This default was chosen, because WhatsApp used the same emoji-style across all platforms at the time of the study. Thus, we could expect participants to be familiar with them. The second style was based on Android 7.0 (“blob emojis”<sup>9</sup>).

### 11.2.2 Password Selection in the Lab

For the first part, participants were invited to a lab at the media informatics research group. The primary task was to create a password that participants could remember well. There was no independent variable for the password selection task, thus participants all received the same study instructions.

#### Metrics

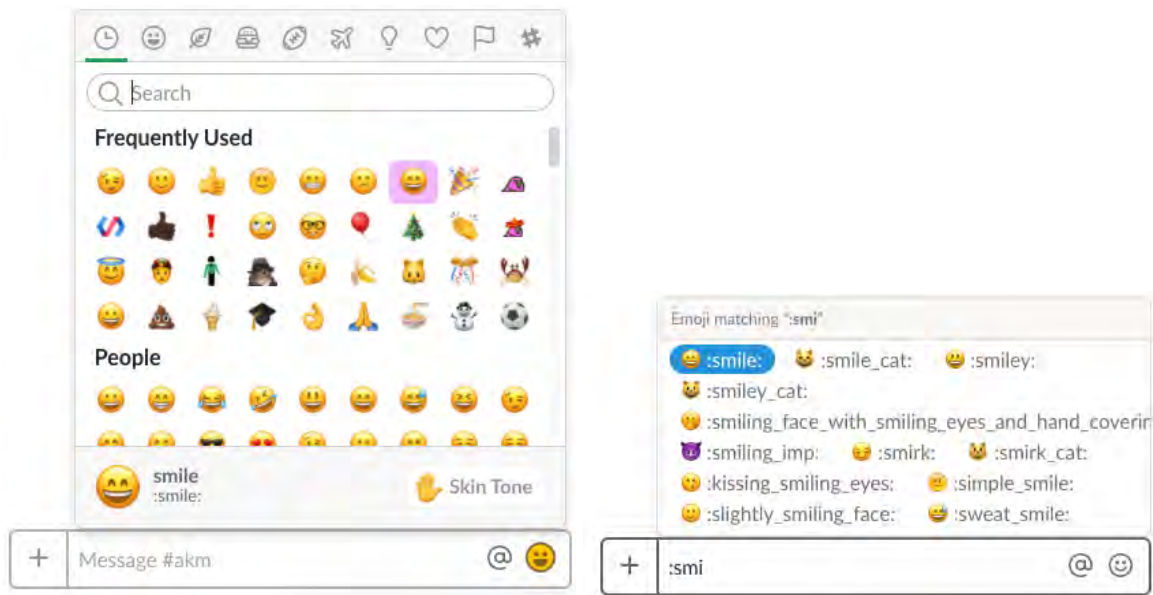
We logged the chosen emojis and their positions inside the passwords. Moreover, we analyzed password characteristics with `zxcvbn` and stored this to the database along with the hashed password. As an indicator for usability of either the picker or the shortcodes, we measured the time taken to select the password. Here, we used the “focus” and “blur” events as start and end points.

---

<sup>7</sup> <https://unicode.org/emoji/charts/full-emoji-list.html> (last accessed 08.03.2018)

<sup>8</sup> <http://emojitracker.com/> (last accessed 08.06.2018)

<sup>9</sup> <https://medium.com/google-design/redesigning-android-emoji-cb22e3b51cc6> (last accessed 08.03.2018)



(a) The picker shows the short-code of the emoji (b) The user can guess the shortcode. Slack offers matching emojis.

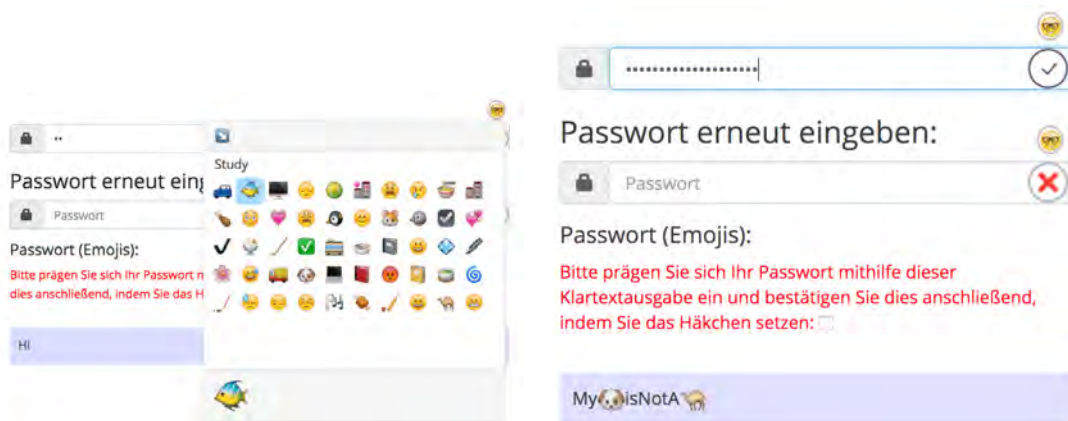
**Figure 11.3:** Slack allows the user to utilize both a point-and-click interface and shortcodes to enter emojis.

On the qualitative side, we used ordinal five-point scales to collect attitudinal data about using emojis inside passwords, the picker, and shortcodes. Demographic data and self-reported password behavior helped us put measurements into context. Everything that participants said during the study was protocolled.

## Procedure

After an in-depth briefing on the purpose of the study and data collection practices, an experimenter explained each step of the study. We provided a standard desktop PC to complete the tasks, which were mostly self-guided. First, participants created a user ID. Since they were going to need this ID later on to allow us matching the lab and field data, we provided a simple algorithm to create an ID. Participants were asked to take the first letter of both their parents' first names and their own birth place, appended by the digit of their birth month (e.g. "IKP10"). This algorithm, albeit not perfectly random, was sufficient to protect personally identifiable information.

This was followed by a questionnaire on demographic data, password coping strategies and attitudes towards emojis in passwords, e.g. *"how likely would you consider using emojis in a password?"* At this point of the study, participants had not yet created an emoji-password, so their attitudes were not biased by the tasks that followed. We also briefed them about the difference between "emojis" and "emoticons" to avoid misinterpretation.



(a) Point and click interface. It is opened by clicking on the 🤖 button. The order of emojis is randomized. Upon selecting an emoji, it automatically closes.

(b) Screenshot of the user study. The selected password needed to be re-created. Additionally, participants need to tick a check-box to confirm that they had memorized their password.

**Figure 11.4:** Prototype as used in the study.

Afterwards, participants completed the password selection task using the prototype. A significant part of the screen was dedicated to list all emojis and their short codes. To provide sufficient background information and introduce a realistic risk [205], the task included a scenario. It asked participants to imagine that they had used WhatsApp for some time and now a new security precaution is introduced. As a safety measure, they were now required to prove their identity with a password upon activating WhatsApp on a new device. The password needed to consist of at least eight alphanumeric characters and at least one emoji. Participants then had to repeat the selected password, and tick a box to confirm that they had memorized it (see Figure 11.4b). This was followed-up through a reflective self-assessment of their behavior during the study (cf. [103]), and a structured questionnaire on their selection strategies.

The study concluded with an interpretation task of two given emojis, namely the “information desk person” (🧑) and “folded hands” (🙏). Those were not available during password selection, and we intended to assess their suitability for future inclusion. In total, the whole session duration was below 15 minutes.

### 11.2.3 Unmoderated Remote Memorability Study

Exactly one week after completing the first study part in the lab, participants were invited to return for the second round of tasks on-line. It primarily focused on memorability metrics and a reassessment of attitudes.



## Independent Variable

To gauge the influence of emoji fragmentation, we used one independent variable “*rendering*” with two levels. For the *control* group, emojis in the picker were rendered as before. In the experimental group, we replaced the iOS emojis with the Android 7.0 version. No other variations were made.

## Dependent Variables

We measured the number of attempts participants needed to log in. If they failed to log in on the first try, we counted this as an error. Moreover, we collected memorization and recall techniques. Finally, subjective usability ratings on the overall concept and qualitative feedback were gathered. This part of the study took about five minutes.

## Procedure

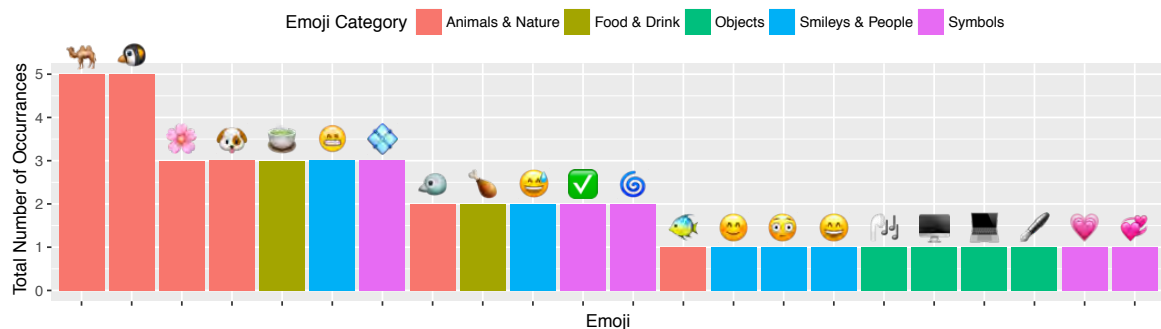
Participants were randomly assigned to one of the two experimental groups. We emailed the corresponding link to the online study and requested completion within two days. The web page instructed them to recreate their user ID, providing the same algorithm as in the first part. Afterwards, participants were asked to log-in via their previously selected password. After three unsuccessful attempts, the log-in counted as failed and the study proceeded automatically. However, as a memory support tool, we displayed the list of short codes after the first failed attempts. The study concluded with an attitudinal questionnaire about the perceived usability of the concept.

### 11.2.4 Recruiting and Demography

We spread a registration link for a “study on emojis” via social networks and an official university newsletter. A 5€ shopping voucher served as incentive. The study was announced to take around 15 minutes. 40 people were screened in and all showed up to their study appointment. All of them were students at the LMU, and aged between 19 and 44 ( $M = 23$ ). Users in this age range are most likely to use emojis on a regular basis [100]. 39 participants returned for the second study part on-line. The control group was formed by 20 participants, and the experimental group by 19.

## 11.3 Results

In the following, we explain participants’ sentiment regarding the usage of emoji-passwords before we report empirical observations and qualitative analyses.



**Figure 11.5:** Histogram of the chosen emojis and their categories. Patterns emerge already in our sample with 40 participants, which hints at potential security problems.

### 11.3.1 Sentiment

Sentiments were assessed with agreement levels to five-point scale items (1 = strongly disagree, 5 = strongly agree). Before participants selected an emoji-password for the first time, they were already reserved towards the statement “I would consider adding an emoji to a password” ( $M = 2.7, SD = 1.4, Md = 2$ ). They did not regard emojis as a way to make passwords more memorable, either ( $M = 2.8, SD = 1.3, Md = 3$ ). After completing the first part, the statement “I liked adding an emoji to my password” was rated slightly more positively with an average of 3.6 ( $SD = 1.19, Md = 4$ ). Having fulfilled all tasks, participants saw general benefits to add emojis in passwords ( $M = 3.5, SD = 1.2, Md = 4$ ). Eleven found the enforced emoji-policy annoying. Hence, attitudes towards using emojis for their personal passwords in the future were rather negative ( $M = 2.6, SD = 1.3, Md = 2$ ), although they were slightly more positive about potential memorability benefits than before ( $M = 3.1, SD = 1.3, Md = 3$ ). In summary, participants were reserved towards adopting emoji-passwords, but their sentiment covered the full spectrum.

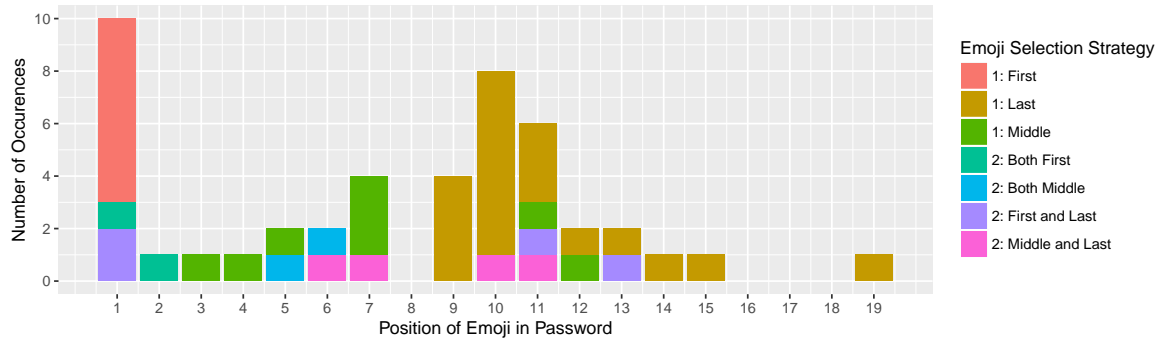
### 11.3.2 Emoji Selection

#### Statistics

Our 40 participants chose a total of 22 different emojis. Most commonly, participants chose the camel (:camel: 🐪,  $n=5$ ), penguin (:penguin: 🐧,  $n=5$ ), grinning face (:grin: 😄,  $n=3$ ), tea (:tea: 🍵,  $n=3$ ), dog (:dog: 🐕,  $n=3$ ), diamond (:diamond\_shape\_with\_a\_dot\_inside: 💠,  $n=3$ ), and cherry blossom (:cherry\_blossom: 🌸,  $n=3$ ). The remaining 15 emojis were chosen less than three times each. Six participants added two emojis to their password though this was not required. On average, password fields had focus for 52.9 seconds ( $SD = 55.00$ ).

Before passwords were hashed, we determined the position of the emojis. The majority of participants who only selected one emoji put it at the end ( $n = 19$ ), while seven started with it and eight put it somewhere in the middle. Adding the emoji there is the most cumbersome





**Figure 11.6:** Histogram of absolute positions of emojis inside passwords. The prefix (1:, 2:) denotes the total number of emojis in the corresponding password. Most participants either started or finished their password with the emoji.

approach due to the double modality switch between mouse and keyboard if the picker is used. The six participants who picked two emojis either put both at the start ( $n = 1$ ), as the first and last characters ( $n = 2$ ), only in the middle as two consecutive characters ( $n = 1$ ), or as middle and last character ( $n = 2$ ). Figure 11.6 shows a detailed breakdown of the chosen positions.

### Self-Reported Selection Strategies

Apart from the emoji, our participants mostly claimed to have created a password like they normally would ( $M = 3.4, SD = 1.45, Md = 4$ ). We elicited selection strategies in two ways: a list of probable strategies that we identified in pre-tests, and an open question where participants were asked to describe their method in detail. Half of the participants indicated that they used an emoji that fits the alphanumeric part of the password. Four said that they preferred an emoji that they frequently use, while another four chose an emoji at random. The remaining participants either associated the picture to a life event ( $n = 3$ ) or hobby ( $n = 1$ ). Eight participants elaborated on their tactics in high detail, and their strategies were more individual than the predefined categories.

Through collaborative thematic analysis, a method similar to affinity diagramming, we identified themes in all qualitative statements and the think-aloud protocol. After the first round of coding, there were 42 codes that were further reduced in an axial coding step. The resulting overall themes were:

- **Internal consistency:** The emoji semantically matches the alphanumeric part of the password, e.g. putting a camel emoji 🐪 after the Greek word for “heat” (P26).
- **Context cues:** The password contains a hint to the participant’s location or to its purpose. For instance, participants used the computer emoji 💻 because they created the password on a PC (P34). Another participant picked the check mark ✅ because it stands for *completing* the study (P38).

- 
- **Replacement:** The emoji replaces either a letter or an entire word. For instance, P17 replaced the letter “p” in their password with a penguin 🐧 because both *words* begin with the same letter.
  - **Appeal:** The emoji visually or emotionally appealed to the participants (mentioned twice for the flower emoji 🌸)
  - **Liking:** Participants had an affection or personal connection to the emoji, e.g. penguins 🐧.
  - **Usability and Security:** An emoji that serves to increase usability or security of the password. Usability could be improved either with a particularly easy-to-type shortcode (:grin:), or by choosing emojis that “stand out from the rest, because they look too similar” (P32, chose the penguin). Security was achieved through perceived randomness or unpredictability (mentioned twice for the diamond 💎).

Within those six themes, we can see a common story line: The selection strategies can be read as an attempt to **improve memorability** of the password. Most participants intuitively focused on creating a password that they could easily recall. None of them mentioned the option to write down the emoji-password or store it externally. The selected emojis thus matched individual memorization approaches.

### 11.3.3 Input Methods

The majority ( $n = 32$ ) intuitively turned to the point-and-click interface to enter their emoji. Five participants used the short-codes and three tried both modalities. Both the picker and the short codes received positive usability ratings from the respective participants ( $M_{picker} = 3.8, SD_{picker} = 1.1, M_{short} = 4.2, SD_{short} = 1.3$ ). Thus, it was easy for participants to enter emojis on a desktop computer in both modalities. Interestingly, the three participants who tried both methods started with the picker and then continued to use the short codes, because they found it more convenient.

Through a qualitative analysis of the think aloud protocol we found that participants considered the advantages and disadvantages of both input methods. The picker was perceived as easy and fast to use, while being less prone to typing errors. Often, participants mentioned that they were already familiar with this entry method (e.g. from WhatsApp web), so they did not have to learn anything new. On the other hand, they described the progressive disclosure paradigm as potential problem, because the button that triggers the picker could be overlooked. Also, some mentioned that it is cumbersome to switch between mouse and keyboard. Those who used the shortcodes appreciated the speed of entry and low effort to add the emoji to the password. However, learning and memorizing the corresponding codes were seen as the main drawbacks. The sentiments did not significantly change in the second part of the study ( $M = 3.5, SD = 1.2, Md = 4$ , overall happiness ratings). In summary, we

can conclude that participants made a deliberate choice about the chosen input method and they were happy with it.

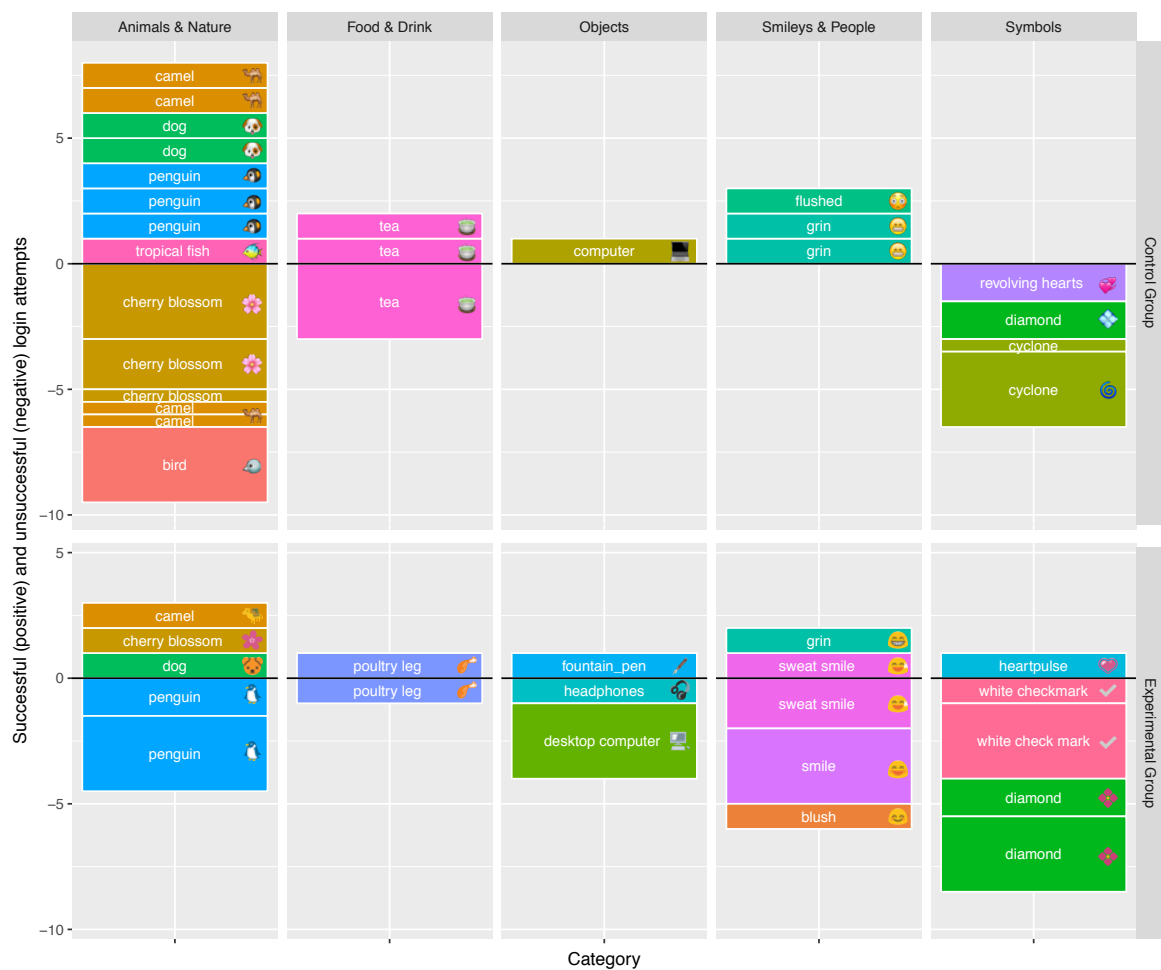
### 11.3.4 Memorability and Recognition

#### Success Rates

Login success rates in the second session did not significantly differ between the two study groups. In the control group, who saw the identical emoji set, five participants failed to log in after a week and 15 succeeded. The experimental group, who received the Android version of the emojis, counted six log-in failures, while 13 participants were able to authenticate despite the change of rendering. This difference was not statistically significant ( $\chi^2(1) = 0.21, p > 0.5$ ). From those who succeeded, 12 participants authenticated on the first try in the control group, whereas this number was slightly lower in the experimental group ( $n = 8$ ). Again, the change was not statistically significant ( $\chi^2(1) = 1.25, p > 0.1$ ), which is probably owed to the sample size. The medians of failed attempts, which are potentially more descriptive in this situation, did differ: the control group's median was 0, while it was 1 in the experimental group. Figure 11.7 visualizes successful and unsuccessful login attempts in high detail. There, we can trace back the emoji that were included in incorrectly entered passwords. Success and failure rates in the “Animals & Nature” category were balanced in both groups. However, participants were twice as likely to fail in the experimental group, if their password contained an emoji from the “Smileys & People” category. Figure 11.8 shows that emojis differing strongly from the iOS version were more associated with login failures. “Symbols” were the category with the highest failure rate across both groups.

#### Textual Feedback

To assess the influence of the rendering modification, we asked whether participants had noticed any differences in the appearance of the emoji. Although the emojis remained identical in the control group, 7 out of the 20 participants indicated that they had noticed a change. This is hard to interpret, but we assume that they meant the order of the emoji inside the 10x5 grid. In the experimental group, where emojis did in fact *look* different, 16 out of 19 participants (84%) talked about this change in the response field. Eight of them did not perceive the alteration as troublesome, while another eight felt that identifying the right emoji was challenging. For three participants, the change of emoji-type was insurmountable: One participant was very sure to have selected the correct emoji, because she picked a “happy face” to express herself. However, she reported to have failed because she was unable to pick the correct emoji in the second study session. Another participant, who successfully authenticated with his emoji-password, indicated that he could only find the right emoji after a Google search. Finally, one participant mentioned that he was troubled by the different versions of the check mark, so he used a trial and error approach until he succeeded.



**Figure 11.7:** Detailed overview of the emojis inside passwords. Columns represent emoji category. The top half shows the control group’s attempts while the bottom depicts the experimental group who had to log in with a different emoji style. Successful logins appear on the positive side of the y-axis, and erroneous login attempts on the negative side. Smileys & People are error prone in the experimental group. Symbols generally led to more errors.



**Figure 11.8:** Effective emoji-set of the experimental group. Error-prone emojis are highlighted with a pink outline. We can see that these differ strongly from the iOS version from session 1. We take this as evidence for the usability problems caused by fragmentation.

### 11.3.5 Memorization Techniques

The emoji-picker and the number of emojis was helpful to recognize and recall the password for 18 participants. The short codes did not achieve this, because only one person reportedly used the code as a cue to the password. A priori agreement levels to the statement “Using emojis in a password makes it more memorable” were neutral with a tendency to disagreement ( $M = 2.8, SD = 1.4, Md = 2$ ). In the exit survey, we again probed this sentiment and noticed a small, statistically insignificant upward trend ( $M = 3.1, SD = 1.3, Md = 3$ ).

### 11.3.6 Interpretation Task

The final task in the lab-session was to provide two word associations for two emojis (🤨, 🙏). Thematic analyses showed a wide range of themes for the “service person” emoji. A total of 14 distinct themes were visible, each mentioned by at least two participants. Most often, this emoji was associated with “female” ( $n = 6$ ) and “pointing” ( $n = 5$ ). The emoji depicts a bell-hop’s “tipping gesture”, which was interpreted as “sassy” ( $n = 2$ ) or “bossy” ( $n = 4$ ), too. Regarding the “praying hands” emoji, participants reached higher consensus on its meaning. Seven themes emerged from the word associations. Most commonly, the participants mentioned “pray” ( $n = 19$ ) and “beg” ( $n = 9$ ). These anecdotal examples highlight the problems produced by unclear meanings of emojis. For passwords, on the other hand, a bigger range of interpretations opens up more opportunities to create a story with emojis.

### 11.3.7 Limitations

The results presented above need to be interpreted in the light of a few important limitations. First, the **sample size** was not large enough to bring about statistically significant test results. Naturally, the null hypotheses (e.g. fragmentation does not affect recall) could be true, but with 40 participants, we lack some statistical power to draw conclusive inferences, especially with a frequentist approach (which emoji was chosen *how often*). Nevertheless, we saw emerging patterns that can be followed up with a quantitative user study. Our primary goal was the exploration of fragmentation and input issues, as well as attitudes. Our participants belonged to the user group who would be most likely to try out emoji-passwords. This makes us confident the trends at least point into the right direction. To explore additional dimensions of the problem space around emoji-passwords, considering a more diverse sample is going to be useful in the future.

Moreover, much of the elicited data is attitudinal, potentially affecting ecological validity of the study. However, we tried to mitigate issues by providing a realistic scenario and had them gauge their behavior. Here, most claimed they acted like they normally would, which has been shown to be a useful indicator as to the trustworthiness of the data [103]. Nonetheless, emoji-passwords constituted an unfamiliar paradigm for all participants, despite their familiarity with both emojis and passwords separately. Therefore, novelty effects are possible in

---

that participants might have spent more time to explore the possibilities than they normally would. However, subjective usability assessments on the feasibility of emoji-passwords were reserved nonetheless. This indicates that respondents critically balanced the pros and cons and did not only focus on the benefits.

## 11.4 Discussion

In the following, the results are put into context. We derive data-driven assessments about the feasibility of emoji-passwords.

### 11.4.1 Selection Strategies and Their Implications

It was evident that participants were keen on creating a **memorable password**. So although people were enabled to select a richer and more diverse password, we observed well-known reactions. The selection strategies were a direct translation of their usual patterns. Many of them noted that they were able to easily include emojis in their long-established strategies and coping behavior. As a consequence, we observed predictable patterns in selection strategies leading to simple attack vectors: participants favored items that helped them maintain their existing selection strategies, which can be exploited to adapt cracking techniques. We can read this as a bad sign, because emojis do not appear to break existing behavior, and only increase the theoretical space of creation strategies. At this point it is difficult to gauge practical security advantages entailed by enlarging the character set. Learning from past roll-outs of new authentication schemes, password *strength* is unlikely to increase drastically by the use of emojis. On the other hand, user-selected passwords have always shown predictable patterns and adding emojis probably will not aggravate the situation further. Therefore, the only major advantage of allowing users to pick emoji-passwords can be an improvement in how password authentication is perceived. In other words, it might help the **user experience (UX)**. Our data is helpful to identify the pitfalls that need to be considered to achieve good UX of emoji-passwords.

### 11.4.2 Factors in Good UX of Emoji Passwords

Improving the UX of a technology is a strategy to drive its persuasiveness [121]. Therefore, we can address a number of aspects to foster a positive user experience of emoji-passwords.

**Freedom to choose** Good UX starts with respect and empathy. Users should not be forced to use emojis in their passwords. Our data discourages the use of a policy that mandates emojis, which a quarter of participants found annoying. Our first study on personality factors showed a large variety of thinking styles about policies (cf. Section 7.2). Therefore, we argue

in favor of autonomy and leaving it up to users to decide whether they want to use an emoji as part of their password(s). So, users are *empowered* to change their behavior and might be persuaded to do so.

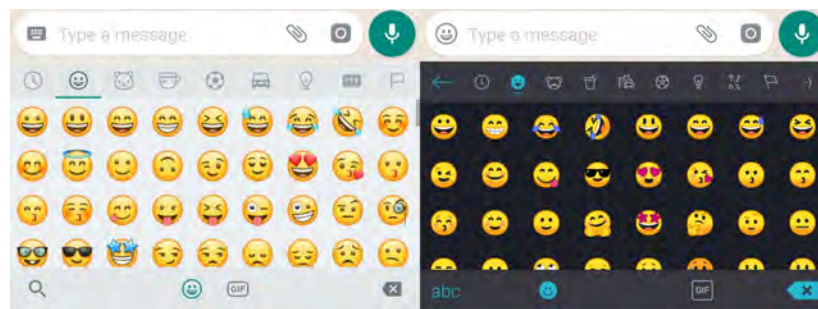
**Learning Curve** Once users understand that they have the freedom to use emoji-passwords, the learning curve for this task looks fairly gentle. Our sample consisted of users who were already familiar with entering emojis on the desktop and therefore immediately knew what to do. Less experienced user groups should be offered assistance in case they decide to experiment with emoji-passwords. We also found that the short codes were mainly used by participants who aimed for efficiency. Thus, there is nothing wrong with enabling short codes for input, but it should be tailored to an advanced user group.

**Pragmatic Constraints for Passwords** Although our sample came from a population of young adults who are highly experienced with emojis, some participants clearly struggled to identify the correct emoji after we exchanged the renderings. Therefore, inconsistencies across platforms could become a show-stopper. At the moment, there are a few problems that contribute to creating inconsistent experiences. First, although emojis are standardized Unicode *code-points*, vendors usually need to create custom emoji-*pictures* for legal reasons. Sets under a public domain or nonrestrictive licenses do exist, but many vendors prefer shipping custom on-brand experiences with a shared design language across their products. On the other hand, it could be useful to form alliances like the World Wide Web Consortium (W3C) to standardize graphical assets of emojis specifically for password-fields. Software keyboards on mobiles would also need to match this hypothetical “password-emoji standard”. As of now, emojis offered by built-in software keyboards often look different to those of certain apps (see Figure 11.9). To solve this inconsistency, a native keyboard could offer an interface that lets the app inject a consistent set of emojis.

After settling on a shared visual representation of emojis, distinctiveness is another critical factor. Some emoji-tuples that look very similar to each other can be troublesome. One of our participants mentioned this issue already in his selection strategy, because he was looking for an emoji that “stands out” and did not “look like the others”. To avoid login failures from confusion, there needs to be a whitelist of emojis that only contains the highly distinctive ones. We can first derive a candidate list based on geometric parameters like shape and color, and then proceed to evaluate them through quantitative user research with a diverse sample.

A final challenge pertaining to the constraints of emoji-passwords are hard usability metrics like efficiency. Higher degree of freedom and creativity support are ineffective if users need to browse through large lists of emojis only to find the correct item. Reducing the emoji-space to a subset emerges as a useful approach to speed up recognition and efficiency. At the same time, it is conceivable that vendors stick to a limited, static subset that is randomly selected from the whitelist of distinct emojis. Distinct subsets, i.e. a different random permutation per vendor, can also mitigate password reuse.





**Figure 11.9:** Screenshots of entering emojis on WhatsApp. While the software provides a set of emojis that is consistent across all platforms supported by WhatsApp (left), users can still turn to the native software keyboard of their OS (right). In this case, the renderings, number, and order are inconsistent.

## 11.5 Conclusion

In this project, we empirically evaluated the usage of emojis in regular alphanumeric passwords to gauge their suitability as persuasive alternative to existing passwords. Our mixed-model user study with forty participants focused on attitudinal and usability aspects. From the selection behavior, memorability results and qualitative feedback we can conclude that emojis do not necessarily lead to memorable passwords (RQ1). Participants wanted to translate their usual password selection behavior to the new paradigm and at least tried to create memorable passwords (RQ4). When participants failed to log in, this could be partially traced back to fragmentation, i.e. differences in the visual representation of the same emoji characters (RQ2). Our sample population was experienced with emojis, but still struggled to match Android emojis to a previously selected iOS emoji. Their personal experience allowed them to easily understand how to enter emoji-passwords on desktops through a graphical point-and-click interface (RQ3). Nevertheless, our prototype made potential usability problems salient for participants and thus their attitudes towards adopting emoji-passwords in the future were reserved (RQ5). Thus, the persuasive power of emoji-passwords was limited, which we attribute to the user experience of the study setting.

Some vendors and service providers have already enabled emoji support for passwords. For instance, Twitter<sup>10</sup> allows users to take advantage of this rich character set. The issues we explored in our study thus affect a growing number of users already now. Roadmaps to address fragmentation and input issues do not exist at this point. However, more and more users are going to find out about the capabilities, perhaps because the feature is presented in blog- and news articles [69]. Therefore, the HCI community should act quick to create a better authentication experience to **scale solutions before problems are going to scale**. Realistically, the propositions and requirements discussed in Section 11.4.2 would require intensive negotiation substantiated with much more user testing data. It is unclear whether vendors are willing to make this investment. We argue that a standardized *emoji-password*-

<sup>10</sup> <https://www.twitter.com> (last accessed 10.03.2018)

*picker* with a random sub-set of white-listed, distinctive emojis is a desirable goal. The “longer term solutions” that are part of the Unicode standard (see Footnote 1) embrace “embedded graphics”, which paves the way to mitigate fragmentation issues. Since leading tech companies have worked together on the standardization of emojis in the past, it would be fruitful to collaborate on authentication issues of emojis, too.

**Future Work** Our prototype did not deliver the best possible user experience. This caveat was partially owed to isolating confounding factors. For instance, a finished solution would not shuffle the order of emojis as we did. Future studies on emoji-passwords need to intensely study different dimensions of user experience issues and potentials. To better understand the current state of emoji-passwords, a diary study might be worthwhile. It would be possible to answer interesting questions on how this novel kind of authentication influences well-established coping strategies. For instance, is it possible to **write down emoji-passwords**, and **share** them with somebody else? Can **password managers** already handle them? Do users take more care to protect emoji-passwords from **shoulder surfers**? Such questions can further help judge the feasibility of this approach and identify important pain points.

At the same time, we saw how existing coping strategies might lead to weak emoji-passwords. Therefore, as a next step, the security benefits should be quantified. Perhaps, this is relatively easy to do with an mTurk study. If our hunch about weak selection strategies is confirmed, persuasive strength feedback for emoji-passwords is an important future research direction. Real-time password meters, particularly those based on zxcvbn, are currently unable to realistically gauge the strength of emoji-passwords. Therefore, further work on modeling strength is necessary. Neural networks, as shown by Melicher et al., could respect the predictability of user-chosen emojis in guess-number estimates [233]. All in all, some more work is going to be necessary to reach the high persuasive power of emoji-passwords that we had hoped for. Our work has laid the foundations for such studies on the quantitative side.

---

## Take Aways

- Users' intuitive reaction to creating an emoji-password was resorting to established strategies with memorability in mind. This is early evidence that the envisioned security benefits might come off very small.
- Emojis did not significantly improve password memorability.
- Simulating log-ins on a different platform than where the emoji-password was selected resulted in notable usability issues, because users fail to recognize the previously selected emojis.
- A point-and-click interface works for emoji-passwords, but emojis must be reduced to a small sample and must not look very similar to each other.
- Most participants were not eager to adapt emoji-passwords. The current user experience reduced the persuasive effects of empowering users to become more creative.

# IV

## SYNTHESIS



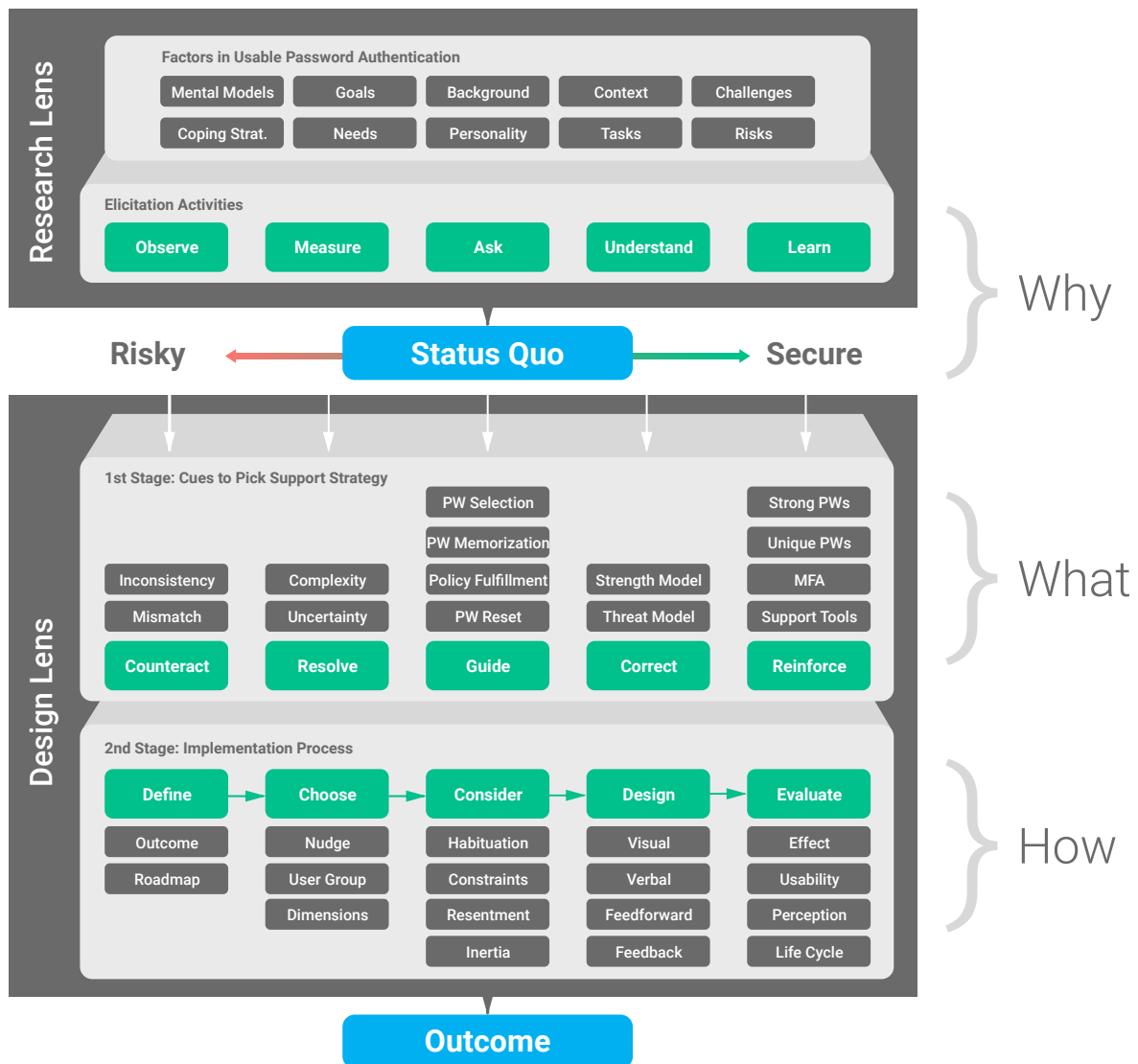
# 12

## Persuasive Design for Password Support

Having reviewed the literature (Part I), and explored both contextual factors (Part II) and novel designs (Part III), we can now take a step back and synthesize common streams, results, and lessons learned. As a result, this chapter presents a new framework for researchers and designers to find and evaluate persuasive strategies that support users in any task that involves passwords. We can find numerous research papers that evaluate a specific aspect without much context information or covariates. For instance, many types of password meters have been designed and evaluated, but most focused on measuring impact in the form of resulting password strength and usability metrics alone. However, aspects like user's preexisting mental models of password strength or the deployment environment were often untouched. At the same time, it is important to focus on the users and choose the support strategy that works best for them under a given configuration of environmental variables. Moreover, we find that many persuasive strategies stay below their potentials [267], or show unexpected effects (see Chapter 10). Therefore, I establish a design process to avoid past shortcomings in nudging effects and to aid strategic decision-making for persuasive design.

The **Persuasive Design for Password Support (P4P)** framework broadens the view on exploring and designing support systems (see Figure 12.1). It is divided into two “lenses” for the task. The “research lens” covers the ever-changing, heterogeneous environmental factors in usable password authentication and how to elicit them. It helps explain *why* users act in certain ways and can provide pointers as to *why* we should try to change the current state. The “design lens” assesses the status quo of password authentication and helps find solutions of different obtrusiveness levels (first stage), i.e. *what* should be the focus of a new persuasive strategy. The second stage defines a user-centered process to implement once the required level of support has been decided. It aids in finding out *how* the persuasive strategy should look. In the following, I explain these elements and show a design exercise where I apply the framework to create a novel approach for a password manager.

# Persuasive Design for Password Support



**Figure 12.1:** Persuasive Design for Password Support (P4P) is a specialized design framework for strategic intervention development.

## 12.1 Research Lens

The first step to create a novel persuasive support strategy is to elicit a number of factors that are involved in usable password authentication. Some of them have been addressed in previous research, e.g. coping strategies, while others are a direct consequence of the findings presented in this thesis (e.g. mental models or personality). Researchers can try to answer the following questions to obtain an overview of the status quo of risky and secure factors:

Question	P4P Element
What are current mental models, e.g. of password strength or support tools?	Mental Models
How do different user groups cope with passwords?	Coping Strategies
What are users primarily trying to achieve when they authenticate?	Goals
What are the dimensions of user needs in this context, also in regard to support?	Needs
How does demographic background impact password attitudes and behavior?	Background
How much is personality visible in attitudes and behavior regarding authentication?	Personality
What is the broader context of the authentication task, e.g. personal vs. work environment?	Context
What (sub-)tasks does the user need to perform to reach their goals?	Tasks
What are the challenges that different users face in the task?	Challenges
What risks are there if the user acts insufficiently secure?	Risks

Some of these questions already hint at the dynamics of certain factors. For instance, users' mental models about threats might adapt over time, e.g. after falling prey to a phishing scam. Adoption rates of password managers are also volatile and the percentage of people relying on them certainly affects the type of novel support strategies one can implement. Although many questions and aspects can be answered by desk research methods, the dynamic evolution of the factors requires ongoing elicitation. Thus, traditional empirical research methods can be used to **observe** behavior and problems, **measure** interactions, explicitly **ask** users about certain aspects, **immerse** in the context of different user groups, and **learn** from all these activities. The result of this exploration is a detailed picture of the status quo.

### Status Quo

The status quo is a snapshot of the levels and relative importance of the factors that contribute to usable password authentication. Although the data elicitation should be generalizable and objective, the judgment of the result is not. It is up to researchers to collaboratively assess the risk level of a certain user group in various contexts. Here, the framework posits that some aspects that have traditionally been considered “risky” behavior need to be seen in a different light as attacks and countermeasures evolve. This follows the argumentation of Florêncio, Herley, Bonneau, and Van Oorshot among others [34, 117, 165, 389]. For instance, not all passwords need to withstand offline attacks. Therefore, it is unnecessary to nudge users to create passwords that would withstand them, because the usability drawbacks are unbearable for many users in most situations. The status quo thus informs the level of intervention in the subsequent design phase.



---

## 12.2 Design Lens

The second half of the framework consist of two distinct stages: 1) Identifying the right level of support and 2) creating novel persuasive solutions. The first stage is strongly influenced by the results of the status-quo analysis, while the second deals with the specific questions during the implementation of a persuasive strategy.

### Cues to Pick the Right Support Strategy

The interventions in Figure 12.1 are ordered by their level of justifiable obtrusiveness. On the left, very risky behavior should be tried to **counteract**. The risk-level assessment is context dependent and can be read from the status quo analysis, where context is one dimension. We can read the elements like “if X then Y” in vertical direction. In particular, the framework follows this pattern:

If <user group>, who are <list of factors>, do <specific action>, then this can be seen as <risky or secure> and justifies <intervention level>.

For example, the framework suggests that if *users of online web-services* show *clear patterns of inconsistent reuse*, i.e. reusing a password from a high-value account on a low-value website, then this is *risky* behavior and should be **counteracted** with obtrusive strategies. The same goes for mismatches of account value and password strength. Other aspects can be **resolved** with less user-involvement and obtrusion. For instance, if status quo analysis shows that current password policies in companies are too complex, they should thus be replaced by simpler versions. Decisions under uncertainty, e.g. when a user needs to assess how valuable an account is, can be resolved with different persuasive patterns, too. Users can be **guided** in password selection, memorization, and reset tasks, as well as fulfilling a given policy. Moreover, it might be reasonable to **correct** users’ mental models of password strength, respectively threats, to empower them to behave more consistently in the long term. Finally, there are a number of aspects that are generally beneficial in terms of security, e.g. using strong and unique passwords, activating multi-factor authentication (MFA), or relying on support tools like password managers. If these are part of the status quo for a given user group, there is no need to change that behavior for the time being. Rather, a support strategy should then positively **reinforce** these actions.

### Implementation Process

Once the correct level of support has been identified, the P4P framework helps with implementing it. First, it is feasible to **define** the envisioned outcome of the strategy, e.g. “stronger passwords” or “changing novices’ mental models of password managers”. Moreover, the general context and roadmap for the implementations are defined at this point. Afterwards,

the designer **chooses** a number of parameters for the strategies. Most notably, the nudging strategy to achieve the target outcome should be chosen and matched to the required level of user support. At the same time, the target user group should be narrowed down, e.g. with personas as shown in Section 7.5.1. The different dimensions of user needs are already addressed at this stage, i.e. “show”, “explain”, “help”, “empower”, because they guide subsequent design stages. When the parameters have been fixed, it is important to **consider** a number of pitfalls in persuasion strategies. For instance, how quickly would users become *habituated* to the nudge and how can one counteract that? What are the *constraints* in different contexts, e.g. deploying the strategy at a company versus a consumer-oriented website for mobile devices? Users also often *resent* attempts to be persuaded, thus the design needs to be more empathetic and ethical at the same time. Moreover, people prefer sticking to current behaviors although they theoretically want to act differently. This discrepancy results in a certain level of *inertia* that a persuasive strategy needs to overcome to be effective. The list of considerations is not exhaustive but covers the most important aspects. If these limitations are too strong, it might be necessary to return to the previous step in the process to resolve them.

Once there is reason to assume that the overall nudging strategy might be feasible, it needs to be substantiated with different **designs**. Here, password interventions have four degrees of freedom: *visual* or *verbal* nudges (e.g. in password meters), and feedforward or feedback techniques (e.g. password suggestions vs. strength assessment of the current password). It is important to consider different versions of the nudge in each of these dimensions to better exhaust the design space. Moreover, it allows for more nuanced options to **evaluate** the overall strategy. Typically, the *effect* or impact of the nudge are compared to a control group, or between different configurations, e.g. the resulting password strength with various feedback mechanisms. The *usability* and *subjective perception* can be evaluated both quantitatively and qualitatively. Finally, it should be investigated how the nudge impacts the *password life cycle* as a whole.

## Outcome

When the process has been completed in full, we are able to judge the feasibility of the strategy and may or may not recommend adopting it. If it is adopted, the status quo is likely to change after a given amount of time. This warrants exploring further opportunities and updating the factors in the “research lens”. Thus, the framework needs to be seen as an iterative process, but it is flexible enough to take shortcuts and start at a later stage. In the following, I discuss an application of the P4P Framework to shed more light on its feasibility.

## 12.3 A Design Exercise with the P4P Framework

Dealing with a multitude accounts, users mostly resort to password reuse. Although some reuse behavior involves considerable risk, Florêncio et al. indicated that it is a neces-

---

sary strategy if users refrain from using password managers [116]. Consequently, Zhang-Kennedy et al. updated common recommendations on reuse [389]. In essence, they recommend to “*strategically* reuse passwords”. Moreover, Wash et al. stated that “defining **appropriate categories** of websites for re-use of passwords of varying strengths is an open area of research;” [363]. We used this as a starting point for a design exercise with the P4P framework. The section is partially based on a Bachelor thesis by Magdalena Siferlinger [306] and a Master thesis by Martin Prinz [256]. I provided the ideas, supervised both students, and guided them through their projects.

### 12.3.1 Phase 1: Research Lens

The goal of the first phase to get a full picture of the status quo. This involves desk research, as well as empirical research methods.

#### Users: Mental models and Strategies

One of the missing puzzle pieces are the specific categories that form reuse strategies. Although related work mentioned distinct exemplary categories, we wanted to narrow down the theoretical space. For this purpose, we planned and executed a mixed-method study to elicit re-use strategies. 35 people completed an online survey that was distributed on social networks. Five passers-by were interviewed in a public location in Munich. The questionnaire was identical in both methods, but the interviews allowed us to ask follow-up questions. Apart from demographics, we mainly inquired the number of accounts, password coping and reuse strategies, and the biggest challenges that respondents face in them. To elicit selection strategies, we provided four scenarios that involved creating accounts for email, social networking, banking, and a news page.

Unsurprisingly, most respondents reported that they reuse passwords either directly or with modifications (77.2%). The remainder relied on a password manager to generate new random passwords. Mangling strategies were predictable (swapping letters for digits, varying appended symbols, and capitalizing letters). More importantly, a thematic analysis of the statements about reuse strategies revealed the following nine themes: Account type (e.g. email accounts), importance (e.g. very important, “throw-away”), strength (e.g. one or two strong and a few weak passwords that are used depending on the perceived threat), time of creation (e.g. the year), frequency of reuse (e.g. a go-to password and a few less frequently typed ones), purpose (e.g. all accounts that were created for a project), base-password plus mangling algorithm, policy-driven (e.g. if the go-to password is rejected, a more complex password is reused), or generating completely new one depending on certain cues. These themes contribute to the status quo of password reuse in this design exercise.

#### Password Managers

Other respondents mentioned using password managers, but there are different types of managers: they can be classified as either *retrieval-based* or *generative*. Retrieval-based pass-

word managers store the user's passwords securely and allow, e.g., to automatically fill out log-in forms on behalf of the user. Once a password is in the user's manager, they can choose to never type it again. Most commercial solutions work this way. On the downside, users do not "practice" their password as much and might not remember it in situations where the password manager is unavailable or too risky to use, e.g. at public computer at an airport. The generative password manager approach solves this problem, because no passwords are stored [228]: it provides the user with an algorithm to re-produce unique passwords (e.g. PwdHash<sup>1</sup>)

**Table 12.1:** Popular password managers: Comparison of features beyond password storage and retrieval. ✓ = available (✓) = available with restrictions, (✗) = with workaround, ✗ = not available

Category	Feature	Dashlane	LastPass	1Password	KeePassX
Automation	form-detection	✓	✓	✓	(✗)
	autofill passwords	✓	✓	✓	(✗)
	autofill personal info	✓	✓	✓	✗
	autofill payment info	✓	✓	✓	✗
	facilitate password reset	(✓)	(✓)	✗	✗
	auto-update password reset	✓	(✓)	(✓)	✗
Organization	grouping / tags / folders	✓	✓	✓	✗
	default groups	✓	✗	✗	✗
	memorization support	(✓)	(✓)	(✓)	✗
	password hints	(✓)	✗	✗	✓
	cross device synchronization	✓	✓	✓	(✗)
	personal info wallet	✓	✓	✓	✗
Security Audit	strength feedback (ad hoc / in client)	✓	✓	✓	✗
	overall score	✓	✓	(✓)	✗
	ad hoc guidance	✓	✓	✓	✗
	security alert	✓	✓	✓	✗
	negative feedback on reuse	✓	✓	✗	✗
	warn about inconsistent reuse	(✗)	✗	✗	(✗)
Password Generation	ad hoc random password	✓	✓	✓	✓
	context aware (policy constraints)	✗	✗	✗	✗
Integrations	Desktop	✓	✓	✓	✓
	Browser	✓	✓	✓	(✓)
	Smartphone	✓	✓	✓	(✓)
Collaboration	Grant others access	(✓)	✓	(✓)	✗
Pricing	Free Version	✓	✓	✗	✓
	Open Source	✗	✗	✗	✓

In Chapter 8, we have already presented an exploration of the mental models of password managers. The most important take-away was that only those who have started using one actually know the benefits, while non-users see password managers as a black box. In chapter 7, we found that using a password manager was associated with demographic factors and some personality traits: Older participants and those with an IT background were more likely to rely on one, while people scoring high on "openness" were slightly less likely. To

<sup>1</sup> <https://pwdhash.github.io/website/> (last accessed 23.03.2018)

---

understand the status quo, we can compare real-world PWMs by their features (see Table 12.1). Additional aspects were part of a recent coverage in the *c't* magazine<sup>2</sup>.

Relying on data from previous chapters, related work, and an additional small-scale empirical evaluation, we can frame the status quo like this:

### **Status Quo of Password Reuse and Password Managers**

**Mental Models.** Users have very vague mental models about password managers.

**Coping Strategies.** Although more people are starting to rely on password managers, the foremost coping strategy is reuse. Here, users often categorize accounts with varying granularity. We have identified nine distinct reuse strategies. **Goals.**

Reusing facilitates recall. Some users are aware of the security risks of reuse and try to mitigate them by strategically reusing passwords. Current password managers try to steer users away from reuse altogether. **Needs.** Users need to be able to re-

call their most important passwords, while the rest could be handled by a password manager. **Background.** Older users appear to be more likely to reuse passwords,

but also more willing to use a password manager. Having an IT background also often goes along with using PWMs to be able to handle a multitude of strong, more random passwords. **Context.** If users encounter PWMs at work, they are likely

to continue using it in private. **Tasks.** Password managers can follow either a generative or a retrieval-based approach which differ in the tasks that the user has to complete. Retrieval-based managers require less cognitive effort. Strategically

reusing passwords might include a deliberate strategy, but is more likely developed intuitively. **Challenges.** Users are challenged with picking the “right” password

manager, switching from one manager to another, and learning how to use them best. Reuse does not challenge users strongly, but recalling the correct category is

sometimes troublesome. **Risks.** Reusing a password too often increases the risk of the “domino effect” if one account is successfully attacked. Forgetting the master-

password of a password manager, e.g. after a long period of inactivity, can lock users out of all their online accounts at once.

## **12.3.2 Phase 2: Design Lens**

With the status quo as starting point, we put on the design lens and pick the focus areas in which to support users with password managers. Particularly, we identified that current PWMs try to “fix the user” by giving negative feedback on reused passwords in case they analyze all saved credentials and display a security score. Thus, there is an opportunity to “fix the system” and better support users in strategic reuse. Hence, our idea is a “Password Reuse Manager” (PWRM).

---

<sup>2</sup> <https://heise.de/-3992417> (last accessed 22.03.2018)

## First Stage

We can address the following cues to figure out *what* to support: Inconsistent reuse should be **counteracted** by the PWRM, while consistent reuse is acceptable. The PWRM can offer default categories to **reduce** complexity and aid in the decision whether to reuse or generate a new password. If a new password is recommendable, the PWRM can **guide** selection, memorization, and policy fulfillment. If the users realize the benefits, this can **reinforce** their choice to adopt a password manager in the first place. It is evident that a password manager is capable of addressing multiple support strategies at once (counteract, resolve, guide, and reinforce), because it is able to interact with the user at multiple touch points of password authentication. At this point, changing the users' mental models is not the PWRMs responsibility, but it might happen along the way.

## Second Stage

In the second stage of the Design Lens, we try to create a concrete implementation of the support strategies.

**Define.** The envisioned outcome is a password manager that better represents the user's current coping strategies and therefore facilitates adoption. The design should be iteratively improved until all common coping strategies are supported (Roadmap).

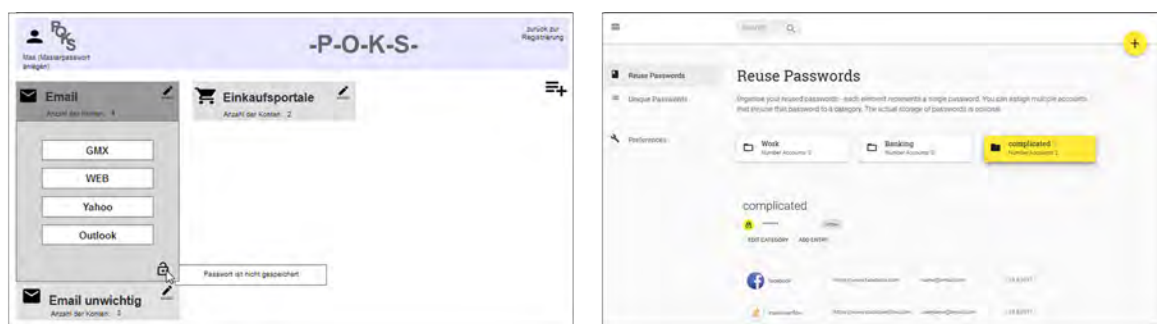
**Choose.** We can choose nudges for each support strategy. Counteracting inconsistent reuse should be done fairly obtrusively, because this kind of behavior generates the greatest risk, respectively the highest amount of effort in recovering from an attack. The "commitment and consistency" principle by Cialdini is perhaps the most promising candidate [57]. Resolving uncertainty and complexity can be addressed with *suggestive* nudges, that facilitate decision-making. The "salience" effect falls into this category [61]. Also, having good *default* categories is a powerful nudge [62]. Similarly, users can be guided by simplifying the password selection process (simplification principle [124]). Finally, using a password manager can be reinforced by offering the service for free and providing a good user experience overall.

In our case, we target users who prefer reusing passwords but also appreciate help in this task. Thus, the target user group consists of novices who have developed their own reuse strategies and prefer maintaining these, but in a simpler way. The "Eliot Elis" persona from Chapter 7 represents this user group well. Therefore, the strategy should fulfill the "show" and "help" dimensions of user needs: Showing problems with inconsistencies and helping with finding alternatives are the foremost requirements we can address with the PWRM. At the same time, the PWRM should *empower* users to maintain their strategy, and to decide not to store certain passwords but only the user-names or a hint.

**Consider.** If the users stick to the PWRM, habituation is actually beneficial, because they learn how to leverage the tool to work for them. In terms of constraints, the PWRM is a

piece of software that would need to pass security audits in company environments. Giving a lot of feedback and interventions might be perceived as authoritative and thus users could resent these strategies. At the same time, the mere presence of the above mentioned nudges might not overcome inertia, so the efficacy needs to be evaluated in multiple steps.

**Design.** In the design phase, we iteratively flesh out the specific information architecture and visuals of the PWRM. In multiple design sessions and evaluations we moved from a paper prototype to clickable wireframes, to a working prototype in the form of a browser extension. Figure 12.2 shows the central page of the PWRM browser extension. The user can put passwords into categories that share the same password. During onboarding, a few default categories based on past reuse strategies are automatically created. They user may also decide to save only a hint for a specific category (see Figure 12.3). This aims to avoid that the user worries about the security of the PWRM – they can simply put the accounts that they do not want managed into one category. In that case, the PWRM still supports them by helping them recall the password, because the category acts as a hint.

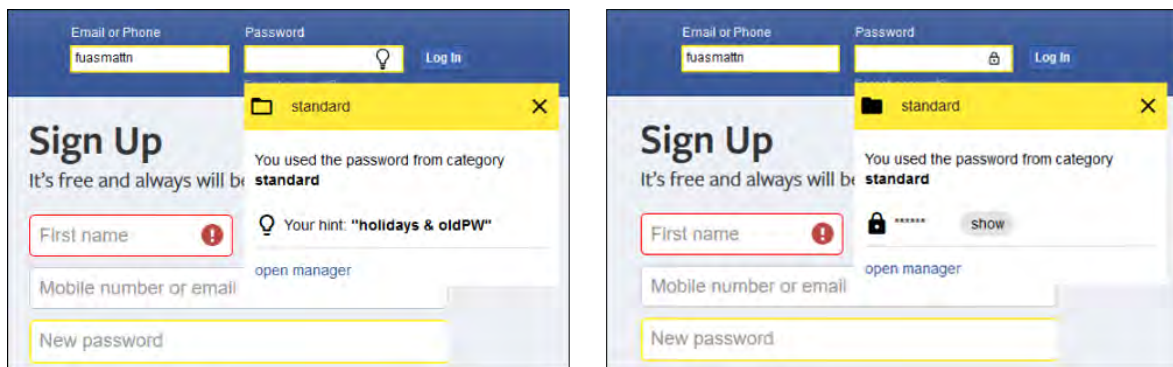


**Figure 12.2:** Different stages of the design iterations. Left: wireframing stage. Right: The main page of the PWRM groups reused passwords into different categories, which makes it easy to indicate that a password has been reused too often. Nudges: **simplification** (through organization), **defaults** (default categories during onboarding)

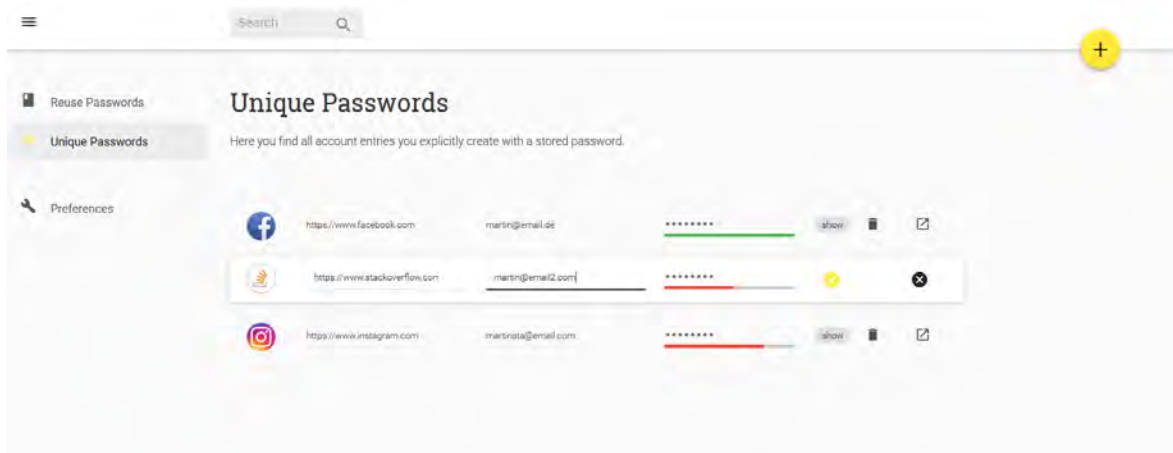
It is also possible to flag passwords as “unique”, i.e. that they do not belong to a group of accounts that share the same password. Figure 12.4 shows the page that also visually indicates the strength of the passwords. The PWRM browser extension detects “submit” events from forms that contain password fields. It opens a small pop-up window to let the user store the password (see Figure 12.5). Here, they can decide whether it should be put into one of the reuse-categories or become a “unique” password, that is not shared with other accounts. Further features and design decisions are indicated in the captions of Figures 12.2, 12.3, 12.4, and 12.5.

**Evaluate.** We iteratively evaluated the concept and the design in a user-centered approach. Table 12.2 shows the four evaluation methods at different stages of the process. In a first wizard-of-oz study, participants found that the categorization feature of the paper prototype was still too complex, although they liked the idea of grouping accounts by passwords. In the





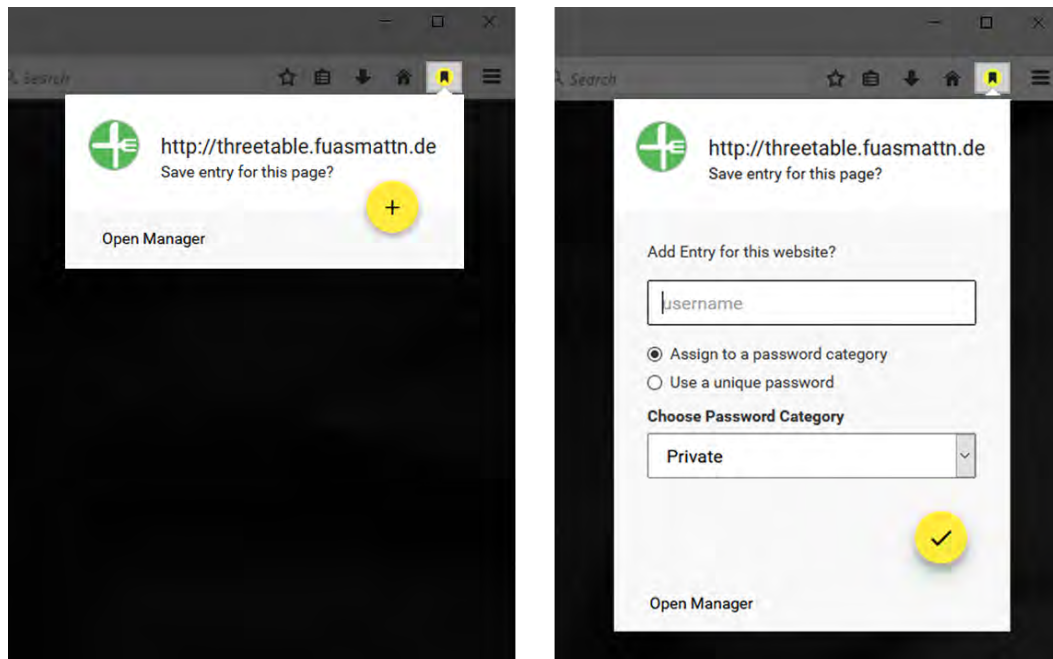
**Figure 12.3:** The PWRM mitigates trust issues by allowing the user to store password hints or categories instead of the passwords themselves. Nudges: **consistency** (stay within the category), **salience** (suggesting categories, visual framing)



**Figure 12.4:** The PWRM holds a separate page of unique passwords. For the user, this reduces uncertainty about the security state of different accounts. Nudges: **commitment** (user should not reuse these passwords.)

second evaluation, the need for more explanation about the potential risks of password reuse became evident. The first usability test with a preliminary implementation of the browser extension showed that participants were able to fulfill all tasks. However, they desired more automation features to speed up the interactions. Finally, we created a beta version of the PWRM extension and recruited 12 participants to test it for ten days. After the ten days, it was clear that the participants only rarely opened the browser with the extension, so we could not conclusively evaluate the solution. The qualitative feedback indicated that participants were browsing the web on their mobiles and also used other browsers during that time. Hence, as a caveat, the effects of the nudges and the impact on the password life cycle would only become visible in a longitudinal, large-scale study, with more careful screening. Unfortunately, this was outside of the scope of this design exercise, and we need to leave this to future work.





**Figure 12.5:** The PWRM browser extension allows the user to save passwords to different categories. Nudges: Defaults, simplification

Stage	Focus	N (m/f)	Key Result
Paper Prototype	Concept	5 (4/1)	Clarify categorization feature
Clickable Wireframes	Information architecture	7 (5/2)	Explain reuse risks
Usability Test	Task success, problems	6 (4/2)	Provide more, automated support
Field Study	Usage patterns	12 (6/6)	Smoother integration necessary

**Table 12.2:** Evaluation iterations of the concept.

## 12.4 Summary

This chapter contributes “P4P” - a framework for persuasive design for password support. Its elements are based on numerous research studies both from related and original work. Other persuasion frameworks for authentication, e.g. the Persuasive Authentication Framework [124], focused on the design space, whereas P4P tries to structure the investigation of the problem (the *why*), design-heuristics (the *what*), and the implementation process (the *how*). So, although it might be seen as an approach to provide a “holistic” view on password support design, I intentionally avoided this term. The elements exhaust much of the research and design space, but there might be other latent factors or different configurations that need more attention in future explorations. Nevertheless, depending on the amount of status-quo information, the structure of the framework allows researchers and designers to start at a later point and follow a structured process. I showed how to use the P4P framework to tackle research questions and explore new opportunities in password support: It led to

the creation of a “Password Reuse Manager” that is now under an open-source license to facilitate distributed development<sup>3</sup>. The design exercise was fairly comprehensive and encompassed a multitude of strategies. However, the framework is also applicable for smaller tasks, e.g. the design of password meters. In the future, P4P can help derive novel solutions outside the current design space of password support.

### Take Aways

- The Persuasive Design for Password Support (P4P) framework structures exploration and design of new solutions to support users in any password-related task.
- It can be applied to big design projects (like a full-blown password manager in our case), but also to smaller aspects (like the design of password meters, help pages, onboarding wizards etc.)

---

<sup>3</sup> <https://github.com/mimuc/pwrm/> (last accessed 23.03.2018)

---

# 13

## Summary

This thesis analyzed a multitude of aspects that need to be considered when we try to support users in password authentication with persuasive design. Hereby, the following research questions were addressed:

- RQ1** What is the role of psychological factors and mental models for password selection and coping strategies?
- RQ2** How can password authentication be simplified for users?
- RQ3** How can we design persuasive strategies to support users in any password-related tasks?

Since password authentication has been under investigation for several decades, we first delimited the landscape of related work in Part I. This helped us find pointers to all three research questions and identify open topics that had still been underexplored (see Chapter 4). In Parts II and III, we reported on a number of empirical experiments and explorative research studies to investigate both human factors and environmental constraints of password authentication. *RQ1* was mainly investigated with online studies in the wild and with survey methods. We explored the mental models of password strength by innovating a research method to inexpensively collect data in the wild. Moreover, we conducted multiple surveys to explore associations between personality and attitudes and behaviors in authentication. We tried to answer *RQ2* with mixed methods, both on the qualitative and the quantitative side. Here, we explored the needs users have in persuasive feedback with a survey and participatory design approach, and derived a solution based on the Decoy effect and emojis inside text-based passwords. Finally, to answer *RQ3*, we discussed a framework to guide the design of persuasive strategies in Chapter 12, which we then also applied to create a novel password-manager. The following sections discuss the central insights and show how they are connected.

---

## 13.1 Central Contributions and Insights

### Psychological Factors and Mental Models in Authentication

In Chapter 3, we found that most related work *describes* coping strategies. Only sometimes the contributing factors like educational background, psychographics, or mental models that foster certain coping strategies are addressed.

We filled this gap in several ways. First, we investigated the mental models of password strength, because these are believed to be highly influential on actual password choice. For these purposes, we innovated on research methods to understand latent password strength perceptions: *PASDJO*, the password game, helped in showing that passphrases are often underestimated by users, although NIST has started to propagate them in favor of highly complex passwords (see Chapter 5). The long-standing belief that strong passwords must include a wide range of characters was clearly visible in the data collected during one year of public deployment. However, users were by and large capable of judging the quality of passwords. We take this as evidence that users are most likely aware of their actions when they select either strong and weak passwords for different purposes. This insight gives rise to a shift in the way we support users in password selection: In many cases it is unnecessary to provide feedback on *strength*, because users can already estimate it well. Therefore, we can tackle other risky behaviors. In fact, password strength is often a secondary risk factor, if users reuse passwords too carelessly. To investigate the real-world constraints for password reuse, we audited the composition policies of the 83 most visited web-sites in Germany (see Chapter 6). We were able to show that it is easy for users to reuse a single password on most sites: it only has to be nine or ten characters long and consist of lower- and uppercase letters and digits. Hence, this finding is another indication that password policies have shaped the mental model that at least three different character classes are absolutely required to form a strong password – and that reuse is less critical, because it is not prevented. Moreover, users do not need a password manager if their go-to passwords are accepted by most websites. However, as soon as they add symbols in the belief that doing so fosters password strength even more, the success rate to reuse the resulting password drops significantly. So, in that case, rejecting passwords based on certain symbols might leave users wondering why these do not boost password strength. The result is a cognitive dissonance in the users' mental models. To resolve this, websites could provide some kind of explanation as to their policy choice, but most fail to do this. Besides, it is unlikely that they disallow certain symbols to hamper password reuse. As a take-away, service providers need to start accepting Unicode passwords without arbitrary length restrictions to avoid confusing users. This allows automatically generating unique passwords of all kinds, which can drive the adoption of password managers. At the same time, Unicode passwords have more ramifications on the use of emojis, which we discuss in a moment.

Real-world constraints like composition policies shape mental models and coping strategies. On the other hand, there is a spectrum of password coping strategies that cannot be explained by environmental factors alone. We hypothesized that a user's personality might play a role

in password selection and coping behavior. In three studies (see Chapter 7), we examined how personality might be associated with different password tasks. We found that personality was a weak, but non-negligible factor in predicting how different user groups deal with policies, perceive strength, or choose passwords. The data allowed us to create user segments in the form of personas that can be used in the development of authentication schemes and support strategies. As a side effect, we observed that people with a background in an IT-related field were more likely to adopt a password manager. Looking at the generally low adoption rates of such software, we explored how users perceive password managers in Chapter 8. We contributed the observation that users appreciate this kind of tool once they were first exposed to the technology, e.g. at work. If they had never used a password manager before, they were unable to anticipate how it might help them. We distinguished important themes that shape mental models about password managers. However, we found that these models were currently not fully matched by commercial software. Consequently, there are novel opportunities to re-design password-managers to better support users.

In summary, we have to understand additional dimensions of the factors that contribute to mental models, because if we do not, we will fail to help users avoid risky password practices. Environmental constraints like password policies probably have the largest influence. Much traditional advice on how to form “good passwords” has led to a skewed mental model of password strength. Second, professional and educational factors are associated in how well users deal with password tasks. Finally, personality also contributes to the shaping of mental models, and warrants further research in this direction.

## Simplification Strategies

Our exploration of mental models and other psychological factors in password authentication revealed that users meet complex tasks with their individual simplification strategies. Especially if a website implements a complex password policy, users often try to get away with as little effort as they can, which results in predictable secrets. To prevent this and offer alternative simplification strategies, researchers have tried numerous approaches. One of them is based on real time feedback during password selection that shows how well the password meets the policy and how strong it is. Many different variations of feedback have been proposed (e.g., [99, 146, 300, 343, 347]), but the designs have mostly been based on assumptions about the users’ needs in simple password feedback. We took a different approach and first tried to validate our assumptions and identify aspects that fell short in the related literature. Through a mixed methods approach (see Chapter 9) we specified user needs and found four central dimensions of password selection support: *showing* current problems, *explaining* the implications, *helping* with improvement, and *empowering* to become creative. These dimensions can be used to facilitate password selection through feedback and feedforward in future solutions.

Moreover, to address the users’ needs and to simplify password selection, we explored two persuasive strategies. The first was based on *showing* current problems and *helping* with improvement. We introduced a *choice architecture* for password selection based on the Decoy effect (see Chapter 10). Through an online experiment, we observed that the Decoy choice

---

architecture did not influence participants as expected. However, displaying a passphrase and making its benefits more visible and easily comparable did result in stronger and longer passwords. Thus, we believe that such a combination of feedback and feedforward is the key to simplifying selection strategies for stronger passwords. Although it is not necessary to pick a strong password for every single account, it is very recommendable to reduce guessability of master-passwords for password managers. The second strategy we explored aimed to simplify memorization of passwords and *empower* users to become creative in their selection. To that end, we evaluated the usability of using emojis inside text-based passwords in two study sessions (see Chapter 11). We created a prototype to enter emoji-passwords that allowed us to measure selection patterns and issues arising from different visual representations of the same set of emojis across platforms (*fragmentation*). For our participants, the concept brought about the intrinsic desire to create more memorable passwords than what is usually possible, thus the simplification approach went in the right direction. However, once participants faced trouble recognizing the right emojis because their visual style had changed, this fragmentation lowered memorability and participants were reserved towards adopting emoji-passwords in the future. So, although the concept generated interest at first, usability troubles outweighed the anticipated benefits. In one of the personality studies (Chapter 7), we had found that certain user groups were more inclined to adopt emoji-passwords than others: they were more acceptable for participants who strongly showed the *Neuroticism* trait. Hence, it is very likely that those users will try to create emoji-passwords in the near future, because some services like Twitter and Slack already support them. Therefore, the usability issues that we identified need to be addressed soon to avoid user frustration due to account lock-outs and inefficient input. Only then will emoji-passwords become a true *simplification* strategy, because right now they miss this target. In conclusion, the task of password creation can be simplified for users through careful tuning of the password policy, feedback, feedforward, and empowerment.

## Guiding Persuasive Designs

One interesting aspect of the persuasive solutions presented in related work is that they rarely explain the design process in forming them. Reading the literature gives the impression that solutions are solely derived from isolated ideation and/or related work. Important iteration stages of the human-centered design process are often underrepresented and it stays unclear what led to different design choices. To identify new opportunities and exhaust the design space, I argued to structure the process and activities specifically for the design of password support solutions.

To that end, I developed a framework that takes all the insights from related and original work into account. The Persuasive Design for Password Support (P4P) framework addresses specific tasks and questions of password authentication, and guides through different stages of the process. At the same time, it allows taking shortcuts and move directly to a later stage, given that prior work paints a clear picture of the status quo (see Chapter 12). To illustrate its usage and applicability, I demonstrated how it informed different stages of the design of a novel password manager. It embraces the fact that many people desire to reuse passwords

and “stay in charge” of their most treasured accounts. The password manager thus adapts to the user’s coping strategies to make the transition smooth. As this is a large software project in an early stage of its implementation, there is a lot of room for improvement and fine-tuning. We contributed the minimum viable product (MVP) under an open-source license to facilitate further development.

## **Eight Recommendations for the Future**

The above summary allows us to give recommendations on service design and research areas. Some bullets confirm prior work and listing them again should be seen as an emphasis.

1. **Consider the evolution of mental models.** Coping strategies adjust to the task load generated by passwords, which fluctuates throughout the years. Users might see complexity as the primary strength component, but this might change as service providers adjust to the recommendations from empirical usability research.
2. **Put less emphasis on password *strength*.** Some researchers have demonstrated that beyond the threshold for online attacks, the benefits of increased password strength are limited. The most important scenarios that really require a strong (and usable!) password are master-passwords and accounts holding particularly sensitive data, e.g. a Dropbox that is full with health records or credit card details.
3. **Remove restrictions, give autonomy.** Service providers should eradicate unjustified complexity requirements, because they have strongly contributed to unreasonable mental models in the past. Instead, foster password diversity through autonomy, i.e. by empowering users to be creative and make informed decisions. We found that users want to be reasonably secure, but often lack the creativity to come up with adequate passwords. The *show-explain-help-empower* paradigm can overcome this creativity barrier and act as an overall guideline for authentication even beyond passwords.
4. **Prepare for more requests of password replacement schemes.** More and more people are willing to use biometrics as primary authentication method<sup>1</sup>. They will expect this technology from products. However, companies often market biometrics as panacea for usable and secure authentication, and fail to make users aware of the ramifications. Therefore, passwords are going to be met with resistance, and we need to reassure users that passwords have irrefutable benefits in certain situations.
5. **Extend the method space.** We can observe a strong tendency towards studies facilitated through mTurk. While the methodology is robust for eliciting quantitative data, the results are only one side of the truth. MTurk studies answer *what works best*, but often fail to explain *why* things work best. Therefore, resurrecting mixed-methods approaches that address qualitative aspects is recommendable for future research in Usable Security and Privacy (USEC).

---

<sup>1</sup> <https://www-03.ibm.com/press/us/en/pressrelease/53646.wss> (last accessed 26.03.2018)



- 
6. **Stay realistic.** Nudges wear off over time, so we have to constantly create new persuasive strategies. Then again, users quickly resent paternalistic guidance and also prefer things to stay as they are. We have to acknowledge that there is only so much we can do. Persuasive support strategies will not work for all users in the same way, but if they reach even a small target group and make their lives a little easier, I believe that they are impactful enough.
  7. **Follow risky ideas.** If we look at the current landscape of research on password support, the design space appears narrow: most published research tackles password meters in different facets. I argue that taking inspiration from other research areas, e.g. behavioral economics, can generate ideas outside the usual spectrum. They might be risky in terms of predictable effect size, but they certainly can counteract habituation effects.
  8. **Give feedback.** Researchers cannot expect that service providers read academic research papers (let alone dissertations). Therefore, we as a community of user advocates have to become active and point out where things go wrong. For instance, it is important to report issues with password policies to service providers. I have engaged in discussions with globally operating companies and was met with an open ear for improvement areas. In the end, this might translate research into graspable impact.

## Conclusion

This thesis has presented a new perspective on a well-known and perhaps unsolvable problem: coping with passwords is hard and annoying for most of us. Nonetheless, reducing the frustration component stays a highly desirable goal. We contributed new insights into the factors that shape coping strategies (mental models, personality) and how to design for the users' implicit and explicit needs (a structured process to fine-tune the mixture of feedback, feedforward, and empowering technology).

## 13.2 Limitations

The findings should not be interpreted without context. Their main limitations arise from the way data was collected and consequently how the findings apply to other contexts. In the following, I will briefly discuss these aspects and shed light on potential risks of advancing science with persuasion.

### 13.2.1 Generalizability

Although we tried to minimize sampling bias, we cannot fully rule it out in any of the reported studies. In total, the fifteen empirical user studies elicited data from 883 people. The surveys were geographically restricted to Germany, the UK, and the USA, because a) the

recruitment tools included the largest user panels in these areas and b) we were only capable to create questions in German and English. Similarly, we only recruited participants from the Munich area for interviews and lab studies. So although we were careful to control for other demographic factors, the samples were rather homogeneous. Also, 883 participants cannot be representative for the entire population of Internet users, so we have to be very careful not to draw specific conclusions about different demographic groups. Future studies will have to solidify the findings for different contexts and user groups. On the upside, geographic factors have not been a major influence on password-related problems and solutions [351, 359].

### 13.2.2 Study Designs and Analysis Methods

Since most of the studies had an exploratory character, there was little room for confirmatory methods. Hence, we mostly relied on correlations and associations between different *dependent* variables. This type of research has the disadvantage that larger sample sizes are required to minimize confidence intervals and to detect latent effects. We did not always achieve optimal levels of statistical power, which is a caveat that needs to be addressed in further research. Nevertheless, the findings help to inform future hypotheses and to run confirmatory studies based on those. In case an independent variable was required to study a phenomenon, we carefully weighed the benefits and shortcomings of various study designs. For the decoy and the emoji-passwords studies (Chapters 10 and 11), we resorted to between-groups settings that usually require a large sample size to achieve high statistical power. On the plus side, they better show contrasting effects. Moreover, we used within-groups designs in two out of the three personality studies (Chapter 7). This allowed for a better understanding of individual preferences and was suitable for smaller sample sizes like in our studies. Exploratory questions make it challenging to anticipate the outcome in either setting, but additional resources might have enabled us to choose an alternative study design. Since we were able to answer our research questions satisfactorily, the choice of our methods was plausible, but future studies might need to reconsider them.

Moreover, we refrained from collecting plain-text passwords for ethical reasons. While other researchers save passwords in clear text, they also need to provide a higher standard of protective mechanisms, like locking access to the data and analyzing it off-line. Since we did not have the resources for such procedures, we found it more reasonable to hash passwords if they needed to be stored. This limits the available depth of post-hoc analyses, which is a caveat. At the same time, the consistent usage of the zxcvbn estimator provided sufficient and reliable details about passwords for the analyses we required. We did, however, have to modify it in order to strip it from sensitive information.

### 13.2.3 Real-World Measurements

Apart from the log analysis of *PASDJO*, we could not collect data *in the wild*. The concepts we evaluated in Part III were not yet mature enough to warrant production-level deployment

---

of the nudges. We tried to increase ecological validity by following established practices in password research (cf. Section 3.1). While these attempts let us assume that participants immersed themselves in the tasks, there is always a small gap between study and real-world contexts. Therefore, we have to leave deployments of our concepts, e.g. emoji-passwords or feedforward techniques, to future work. Moreover, coming back to recommendation 5, we might be able to assess the ecological validity of the existing data through ethnographic methods, e.g. diary studies or contextual inquiry.

### 13.2.4 Ethics and Risks

Studying the users' psyche, like cognitive biases and personality, to aid the design of persuasive interventions bears certain ethical risks. Often, we investigate unconscious phenomena, for instance, how the decoy effect influences users' decision-making. Therefore, we need to always consider how findings in this area might be exploited. There was a recent episode of questionable analysis of personality profiles: Cambridge Analytica, a British political consulting firm, accessed millions of Facebook users' data without their consent to target political campaigns based on their personality and other factors<sup>2</sup>. Although studying users' personality to support them in password authentication appears less critical than politically motivated manipulation techniques, we still have to weigh the benefits against the risks. For instance, if future research corroborates our findings about the associations between personality and password selection, this might allow adversaries to target attacks more efficiently. Thus, the P4P framework includes this important aspect in the hope of seeing more discussions of ethical risks in the future.

---

<sup>2</sup> <https://www.nytimes.com/2018/03/17/us/politics/cambridge-analytica-trump-campaign.html> (last accessed 27.03.2018)

# 14

## The End - Ideas and Final Remarks

In the last chapter of this thesis, I would like to present some ideas for future research in persuasive password support. Although I would have loved to start working on these already now, they go beyond the scope of this thesis and should be left to future work. As final remarks, I want to share my views on current developments of password authentication, where I see it in the near future, and how its future can be shaped more positively.

### 14.1 Ideas for Future Work

The recommendations in Section 13.1 touched on future work on a high level. However, one recommendation was to follow up on ideas off the beaten path. Addressing this criterion, the following sections provide more specific ideas to work on password support in the future.

#### 14.1.1 Follow-Up Studies

First and foremost, it will be worthwhile to address the limitations of the studies we conducted, or follow up on the leads from initial results. For instance, the PASDJO deployment in the wild suffered from lack of demographic data to understand individual differences in password perception. The second study on personality lacked more randomness of the available passwords and it would thus be a straightforward solution to integrate a short, optional psychometric questionnaire into PASDJO. Alternatively, we could have crowd-workers fill out a more elaborate psychometric survey before they play PASDJO and then fit regression models on the data. In any case, by combining the two approaches we will overcome current limitations of the two studies at once.

Furthermore, it will be vital to evaluate the Password Reuse Manager (PWRM, see Chapter 12) with a longitudinal study. At this point, we used a rapid iterative design method that was focused on qualitative feedback on features. The next step will be to invite a larger audience

---

to test it for at least one month. To boost usage numbers during the field trial, the browser extension can be made compatible with additional browsers with very little effort. It is already built on the WebExtensions API, which is increasingly solid on all major browsers. We had also identified specific research questions that can be answered with A/B testing. One of the issues was that users kept their previous password managers activated and the additional extension simply generated more effort. The follow-up study should therefore screen for participants who do not use a password manager yet (group A), and another group who will be required to move over to the PWRM as single management solution throughout the entire study (group B). To extend the method space (see recommendation 5 in Section 13.1) and better understand how the support works for the users, the study should conclude with the *break up-/love-letter method*<sup>1</sup>. Here, participants decide whether to write a love or a break-up letter to the PWRM, which will allow us to understand the emotional component of password support.

Finally, some of our other studies involved a password *creation* task (see Chapters 7,10, and 11). Although we followed guidelines to establish realistic scenarios, it would have been desirable to study password selection in-situ. Thus, in the future, we should leverage existing web-applications to test concepts when they are mature enough. Establishing collaborations with industry partners is critical to carry out this research approach.

### 14.1.2 Personalizing Password Policies

Personalization is a current big trend in user experience design and persuasive technology<sup>2</sup>, especially since the breakthrough in machine-learning capabilities. It was already an element in the Persuasive Authentication Framework [124]. We have found that personality has a notable influence on how users perceive password policies (see Chapter 7). Therefore, it is conceivable to leverage state-of-the-art computing power to tailor password requirements to individual users instead of relying on a one-fits-all approach. Hypothetically, if the system could detect what policy will *help* the user in finding the best trade-off between password strength and usability, its user experience would not be degraded by password authentication (as it is now). I provided initial pointers about the benefits in a position paper adjunct to *PERSUASIVE'17* [287].

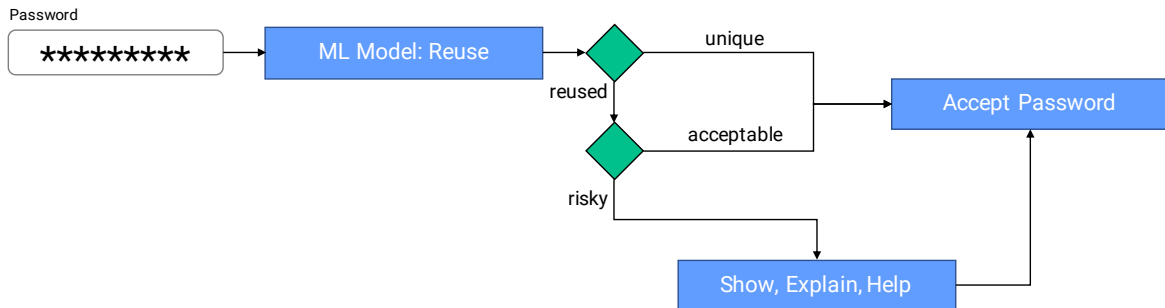
Tackling this challenge is possible with several focus areas, where the P4P framework can guide design decisions. For instance, we could advance algorithms to detect re-used passwords more reliably at registration. Khern-am-nuai et al. have showed an initial proof-of-concept that keystroke dynamics provide useful features [193], but they performed all analyses post-hoc. Machine-learning can potentially enable reuse detection already at enrollment and allow us to tailor feedback more appropriately. For instance, we can design a

---

<sup>1</sup> <https://medialabamsterdam.com/toolkit/method-card/break-uplove-letter/> (last accessed 28.03.2018)

<sup>2</sup> <https://www.forbes.com/sites/shephyken/2017/05/13/recommended-just-for-you-the-power-of-personalization/> (last accessed 28.03.2018)

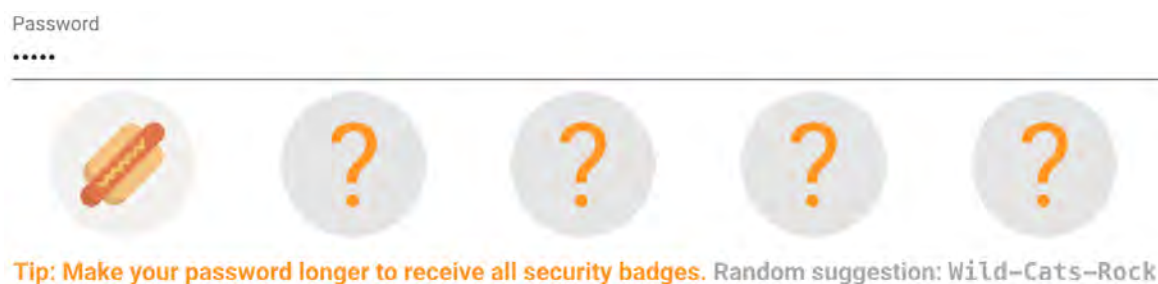
solution that makes alternatives to password reuse more salient, in case reuse practices become too risky (the process is depicted in Figure 14.1). Similar approaches are conceivable with different personality profiles. However, here it is much more likely to tailor personalized support strategies during password *reset*, because the service potentially possesses more data about the user at that point.



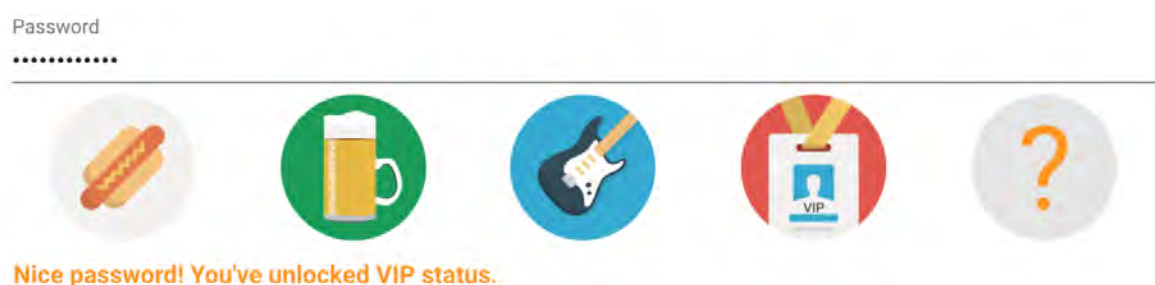
**Figure 14.1:** Personalized intervention for password reuse. The keystrokes inside a password field are analyzed by a machine learning model. If the password is unique, do not intervene. If the password is reused, but there are no signs for overlaps between categories, we can accept the password, too. However, if the password is reused across different categories, we use the show-explain-help paradigm to empower users to make an informed decision. In any case, the password is accepted to avoid being too restrictive.

### 14.1.3 Contextualizing Password Feedback

Beyond personalization, which can be described as context-sensitive adaptation to the user, we should also consider tailoring password feedback to the system’s context. For instance, we can create themed password meters to avoid habituation effects [344]. Kroeze and Olivier proposed using an evolving Pokemon as metaphor for the “evolution state” of the password [204]. This kind of meter should probably be deployed on a game-related service, or directly for the registration page of Pokemon Go. As part of a design study, I created a themed password meter that matches the deployment context: Figure 14.2 shows a password meter that was originally created for a music-festival app. It is based on the *share-explain-help* paradigm and is only one of many design options for this context. For instance, we could also choose a personalized social nudge and say “*only Justin Bieber fans use passwords like that*” – this could trigger in-/out-group biases and motivate some users to act differently. Ideally, such feedback systems are evaluated in situ, but to make it easier, we can use a cover story first, e.g. by creating a mock-website and recruit participants for usability tests thereof. Moreover, to assess the advantages of contextualized feedback, there should be a control group with neutral feedback.



(a) Contextualized feedback and feedforward for a weak password.



(b) When the password is sufficiently strong, provide reinforcing feedback.

**Figure 14.2:** A password meter for users of music festival app that uses context-related imagery to convey password strength. It is designed to drive curiosity. Feedback and feedforward follow the *show-explain-help* paradigm.

#### 14.1.4 Solving Password Breach Aftermath

Database breaches containing passwords occur frequently with different levels of severity depending on security standards at the affected service. Users face an immediate risk, but might be unaware of the situation, so they do not take swift measures to protect their account. To mitigate this, service providers invalidate affected user accounts after they have detected a breach. This action is necessary to reestablish a secure overall system state where attackers cannot impersonate users on the platform. Usually, service providers inform users upon this countermeasure and prompt them to reset their passwords via email campaigns [169]. While these countermeasures are unavoidable, they fall short of addressing the domino effect of reused passwords [173]. Many users do not understand the ramifications of a password breach and might only reset the password on the affected site. All other sites that share the same credentials remain at risk. I propose that service providers constantly monitor password leaks both on their own and on third party websites. In other words, while they might use a blacklist during account creation, there is usually no check-up mechanism that takes the updated risk-level into account. During this routine check-up, accounts that share credentials with affected third parties can be invalidated. The implementation should be straightforward with the aid of breach data freely available from different sources, e.g. <https://haveibeenpwned.com/Passwords>. However, there are a few issues that we need to solve in the future. First, freely available password lists rarely contain user-names

to avoid targeted attacks. Second, if accounts are invalidated because a check-up has found a breached set of credentials, users might misunderstand the reasons for the account lock-down. Phishing attacks often exploit “your account was locked down for security reasons” as a hook to obtain users’ real credentials. Sending out emails with reset prompts is thus especially dangerous because they may raise suspicion, and it is hard for novice users to verify if the prompt is legitimate.

Finally, it would be interesting to study how user behavior changes after a breach. So far, there have been attempts to collect initial data [169], but there is far more that we have not measured, yet. For instance, it will be interesting to find out if personality traits are associated with user behavior after a breach. While it would be fair to assume that *conscientious* users change their passwords more quickly and consistently, we did not necessarily observe such expected behavior in our own studies. Thus, there might be many opportunities for surprising insights.

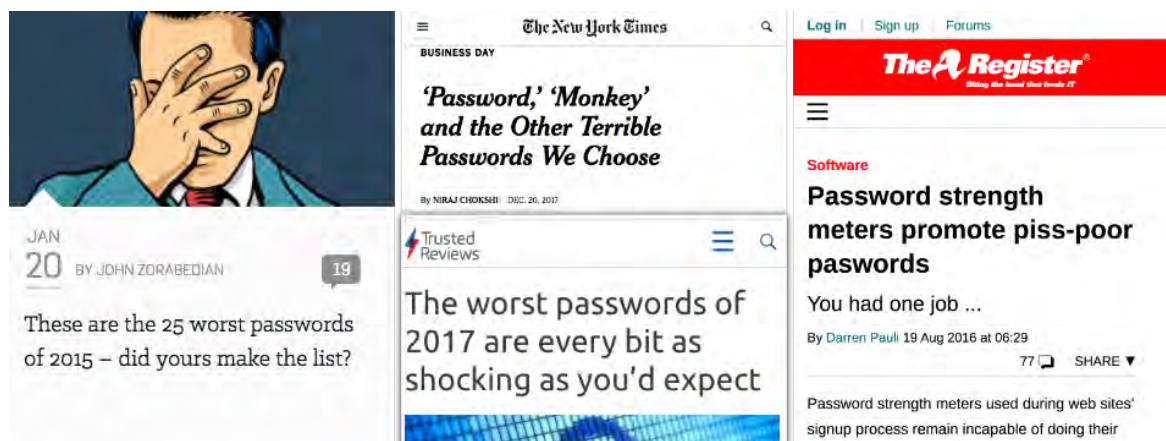
### **14.1.5 Re-locating Password Feedback**

A final problem to address in future work is the fact that many users simply do not *notice* password feedback and therefore cannot re-assess their choice. Many users look down on their hands when typing passwords on regular hardware keyboards, because they have not mastered touch typing. During password entry, on-screen feedback is then out of sight. This could explain why password meters have smaller impact on password selection in the wild: while study participants might take care to follow all on-screen instructions in an online experiment, users are often less focused on the screen in real-world tasks. Therefore, we should study the impact of password feedback if it is relocated closer to the hands. The Touch Bar of current generations of the Apple MacBook Pro models is a visual output close to the hardware keys. It could be leveraged as display for both visual and textual password strength feedback. To the best of my knowledge, this has not been investigated, but might be a feasible solution.

## **14.2 Final Thoughts**

I believe that users are not the weakest link in the authentication chain. There are always security mitigations on the system side that can take responsibility away from users. For instance, if all services had infinite resources to invest into penetration tests and alike, the need for exceptionally strong passwords vanishes. And although some user behavior is arguably risky, people should not be blamed for faulty systems that permit risks. Commentaries on the web and other media do the rest when they name and shame users for weak password behavior (see Figure 14.3). Perhaps, their intentions are good, but the subtext is different: Plenty of users are doing the same, so why would any one user act differently than the rest? This is maladroitness framing of normative messages at best [56].





**Figure 14.3:** Some media are constantly shaming users for their “bad passwords”, arguably as clickbait. However, the reports also act as reinforcement because of the implicit normative message these headlines convey [56]: Many other people behave the same way, so it is fairly normal.

Passwords annoy users for a good reason. Most of us will not grieve, if passwords become obsolete tomorrow. Probably, in the future, we will be able to make machines intelligent enough to independently decide whether a user should be granted access or not, all without explicit authentication. PCs and mobiles will automatically lock when the user is not in the vicinity. Natural interactions will enable us to tell virtual assistants to temporarily grant access to devices, e.g. by saying “*Alexa, my daughter can use my iPad while we are driving to Italy, but make sure that she can only browse safe websites*”, all without the hassle of sharing PINs or enrolling biometric features. Naturally, a few drawbacks will remain, because perfect security cannot be guaranteed, and this will raise trust issues. People want to stay in charge and although passwords appear to be an inferior technology, they can achieve a higher level of perceived control. Thus, there will not only be a need but also a *demand* for knowledge-based authentication in the future. Drawing on an analogy, passwords are perhaps the vinyl records in a world of ubiquitous music streaming: They are a bit clunky, impractical, and can be damaged through careless handling. On the other hand, experts still recommend them despite being an obsolete technology. People treasure them for years, often become collectors and try to find the rare, unique gems of long-lost quality. Once attacks on password alternatives like multimodal biometrics become commonplace, there might be a time when passwords see a revival in popularity like vinyl.

## BIBLIOGRAPHY

- [1] Alessandro Acquisti, Idris Adjerid, Rebecca Balebako, Laura Brandimarte, Lorie Faith Cranor, Saranga Komanduri, Pedro Giovanni Leon, Norman Sadeh, Florian Schaub, Manya Sleeper, Yang Wang, and Shomir Wilson. 2017. Nudges for Privacy and Security: Understanding and Assisting Users' Choices Online. *Comput. Surveys* 50, 3 (2017), 1–41. DOI:<http://dx.doi.org/10.1145/3054926>
- [2] Alessandro Acquisti and Grossklags Jens. 2008. What Can Behavioral Economics Teach Us about Privacy? In *Digital Privacy - Theory, Technologies, and Practices*, Alessandro Acquisti, Stefanos Gritzalis, Costas Lambrinoudakis, and Sabrina De Capitani di Vimercati (Eds.). Vol. 6545. Auerbach Publications, Boca Raton, FL, USA, 363–374.
- [3] Anne Adams and Martina Angela Sasse. 1999. Users Are Not the Enemy. *Commun. ACM* 42, 12 (1999), 41–46. DOI:<http://dx.doi.org/10.1145/322796.322806>
- [4] Anne Adams, Martina Angela Sasse, and Peter Lunt. 1997. Making Passwords Secure and Usable. *People and Computers* 34, 1 (1997), 1–15. DOI:<http://dx.doi.org/10.1145/99977.99993>
- [5] Alexander T. Adams, Jean Costa, Malte F. Jung, and Tanzeem Choudhury. 2015. Mindless Computing : Designing Technologies to Subtly Influence Behavior. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing (Ubicomp '15)*. ACM, 719–730. DOI:<http://dx.doi.org/10.1145/2750858.2805843>
- [6] Seb Aebischer, Claudio Dettoni, Graeme Jenkinson, Kat Krol, David Llewellyn-Jones, Toshiyuki Masui, and Frank Stajano. 2017. Pico in the Wild: Replacing Passwords, One Site at a Time. In *Proceedings 2nd European Workshop on Usable Security*. Internet Society, Paris, France, 1–13. DOI:<http://dx.doi.org/10.14722/eurousec.2017.23017>
- [7] Heikki J. Ailisto, Mikko Lindholm, Jani Mantyjarvi, Elena Vildjiounaite, and Satu-Marja Makela. 2005. Identifying people from gait pattern with accelerometers. In *Proceedings of SPIE - The International Society for Optical Engineering*, Anil K. Jain and Nalini K. Ratha (Eds.). Bellingham, WA, 2005, 1–8. DOI:<http://dx.doi.org/10.1117/12.603331>

- 
- [8] Devdatta Akhawe and Adrienne Porter Felt. 2013. Alice in Warningland: A Large-Scale Field Study of Browser Security Warning Effectiveness. In *Proceedings of the 22nd USENIX Security Symposium*. 257–272. <http://research.google.com/pubs/archive/41323.pdf>
- [9] Mohammed A Fadhil Al-husainy and Raghda Ahmed Malih. 2015. Using Emoji Pictures To Strengthen the Immunity of Passwords Against Attackers. *European Scientific Journal* 11, 30 (2015), 153–165.
- [10] P.S. Aleksic and A.K. Katsaggelos. 2006. Audio-Visual Biometrics. *Proc. IEEE* 94, 11 (nov 2006), 2025–2044. DOI:<http://dx.doi.org/10.1109/JPR0C.2006.886017>
- [11] Nouf Aljaffan, Haiyue Yuan, and Shujun Li. 2017. PSV (Password Security Visualizer): From Password Checking to User Education. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 10292 LNCS. 191–211. DOI:[http://dx.doi.org/10.1007/978-3-319-58460-7\\_13](http://dx.doi.org/10.1007/978-3-319-58460-7_13)
- [12] Patricia Arias-Cabarcos, Andres Marin, Diego Palacios, Florina Almenarez, and Daniel Diaz-Sanchez. 2016. Comparing Password Management Software: Toward Usable and Secure Enterprise Authentication. *IT Professional* 18, 5 (sep 2016), 34–40. DOI: <http://dx.doi.org/10.1109/MITP.2016.81>
- [13] Dan Ariely, Joel Huber, and Klaus Wertenbroch. 2005. When Do Losses Loom Larger Than Gains? *Journal of Marketing Research* 42, 2 (2005), 134–138.
- [14] Dan Ariely and Thomas S. Wallsten. 1995. Seeking Subjective Dominance in Multidimensional Space: An Explanation of the Asymmetric Dominance Effect. *Organizational Behavior and Human Decision Processes* 63, 3 (1995), 223–232. DOI: <http://dx.doi.org/10.1006/obhd.1995.1075>
- [15] Adam J Aviv, Katherine Gibson, Evan Mossop, Matt Blaze, and Jonathan M. Smith. 2010. Smudge Attacks on Smartphone Touch Screens. In *Proceedings of the 4th USENIX Workshop on Offensive technologies (WOOT '13)*. USENIX Association, Washington, DC, USA, 1–10.
- [16] Elizabeth B.-N.Sanders. 2002. From User-centered to Participatory Design Approaches. In *Design and the Social Sciences. Making Connections* (1 ed.), Jorge Frascara (Ed.). Taylor & Francis, London, United Kingdom, Chapter 1, 1–8. DOI: <http://dx.doi.org/10.1201/9780203301302.ch1>
- [17] Daniel V Bailey, Markus Dürmuth, and Christof Paar. 2014. Statistics on Password Re-use and Adaptive Strength for Financial Accounts. In *Proceedings of the International Conference on Security and Cryptography for Networks*. Springer, Amalfi, Italiy, 218–235. DOI:[http://dx.doi.org/10.1007/978-3-319-10879-7\\_13](http://dx.doi.org/10.1007/978-3-319-10879-7_13)

- 
- [18] Ben F. Barton and Marthalee S. Barton. 1984. User-friendly password methods for computer-mediated information systems. *Computers and Security* 3, 3 (1984), 186–195. DOI:[http://dx.doi.org/10.1016/0167-4048\(84\)90040-3](http://dx.doi.org/10.1016/0167-4048(84)90040-3)
- [19] Ulrich Bayer, Imam Habibi, Davide Balzarotti, Engin Kirda, and Christopher Kruegel. 2009. A view on current malware behaviors. In *Proceedings of the 2nd USENIX conference on Large-scale exploits and emergent threats (LEET '09)*. USENIX Association, Boston, MA, USA, 1–8. <https://dl.acm.org/citation.cfm?id=1855684>
- [20] Frank R Bentley and Ying-Yu Chen. 2015. The Composition and Use of Modern Mobile Phonebooks. In *Proceedings of the ACM CHI'15 Conference on Human Factors in Computing Systems*. ACM, 2749–2758. DOI:<http://dx.doi.org/10.1145/2702123.2702182>
- [21] Chandrasekhar Bhagavatula, Blase Ur, Kevin Iacovino, Su Mon Kywe, Lorrie Faith Cranor, and Marios Savvides. 2015. Biometric Authentication on iPhone and Android: Usability, Perceptions, and Influences on Adoption. In *Proceedings 2015 Workshop on Usable Security*. Internet Society, Reston, VA, 1–10. DOI:<http://dx.doi.org/10.14722/usec.2015.23003>
- [22] Kemal Bicakci, Nart Bedin Atalay, Mustafa Yuceel, and Paul C van Oorschot. 2012. Exploration and Field Study of a Password Manager Using Icon-Based Passwords. In *Proceedings of International Conference on Financial Cryptography and Data Security*. Springer, Kralendijk, Bonaire, 104–118. DOI:[http://dx.doi.org/10.1007/978-3-642-29889-9\\_9](http://dx.doi.org/10.1007/978-3-642-29889-9_9)
- [23] Robert Biddle, Sonia Chiasson, and P.C. Van Oorschot. 2012. Graphical passwords. *Comput. Surveys* 44, 4 (aug 2012), 1–41. DOI:<http://dx.doi.org/10.1145/2333112.2333114>
- [24] J. Bigun, J. Fierrez-Aguilar, J. Ortega-Garcia, and J. Gonzalez-Rodriguez. 2005. Combining Biometric Evidence for Person Authentication. In *Advanced Studies in Biometrics*. Number January 2003. Springer Berlin Heidelberg, 1–18. DOI:[http://dx.doi.org/10.1007/11493648\\_1](http://dx.doi.org/10.1007/11493648_1)
- [25] Matt Bishop and Daniel V. Klein. 1995. Improving system security via proactive password checking. *Computers & Security* 14, 3 (1995), 233–249. DOI:[http://dx.doi.org/10.1016/0167-4048\(95\)00003-Q](http://dx.doi.org/10.1016/0167-4048(95)00003-Q)
- [26] Jeremiah Blocki, Anupam Datta, and Joseph Bonneau. 2016. Differentially Private Password Frequency Lists. In *Proceedings 2016 Network and Distributed System Security Symposium*. Internet Society, San Diego, CA, USA, 21–24. DOI:<http://dx.doi.org/10.14722/ndss.2016.23328>
- [27] Jeremiah Blocki, Ben Harsha, and Samson Zhou. 2017. On the Economics of Offline Password Cracking. (2017).

- 
- [28] Jeremiah Blocki, Saranga Komanduri, Ariel D Procaccia, and O R Sheffet. 2013. Optimizing Password Composition Policies. In *Proceedings of the fourteenth ACM conference on Electronic commerce*. ACM, Philadelphia, Pennsylvania, USA, 105–122. DOI:<http://dx.doi.org/10.1145/2482540.2482552>
- [29] Greg E. Blonder. 1996. Graphical password. (1996). <https://www.google.com/patents/US5559961>
- [30] Hristo Bojinov, Elie Bursztein, Xavier Boyen, and Dan Boneh. 2010. Kamouflage: Loss-Resistant Password Management. In *Proceedings of ESORICS*, Dimitris Gritzalis, Bart Preneel, and Marianthi Theoharidou (Eds.). Springer Berlin Heidelberg, Athens, Greece, 286–302. DOI:[http://dx.doi.org/10.1007/978-3-642-15497-3\\_18](http://dx.doi.org/10.1007/978-3-642-15497-3_18)
- [31] Joseph Bonneau. 2012a. *Guessing human-chosen secrets*. PhD Thesis.
- [32] Joseph Bonneau. 2012b. The Science of Guessing: Analyzing an Anonymized Corpus of 70 Million Passwords. In *Proceedings - IEEE Symposium on Security and Privacy*. IEEE Comput. Soc, 538–552. DOI:<http://dx.doi.org/10.1109/SP.2012.49>
- [33] Joseph Bonneau, Elie Bursztein, Ilan Caron, Rob Jackson, and Mike Williamson. 2015. Secrets, Lies, and Account Recovery. In *Proceedings of the 24th International Conference on World Wide Web - WWW '15*. ACM Press, Florence, Italy, 141–150. DOI:<http://dx.doi.org/10.1145/2736277.2741691>
- [34] Joseph Bonneau, Cormac Herley, Paul C. van Oorschot, and Frank Stajano. 2012. The Quest to Replace Passwords: A Framework for Comparative Evaluation of Web Authentication Schemes. In *Proceedings of the IEEE Symposium on Security and Privacy*. IEEE, San Francisco, CA, USA, 553–567. DOI:<http://dx.doi.org/10.1109/SP.2012.44>
- [35] Joseph Bonneau, Cormac Herley, Paul C. Van Oorschot, and Frank Stajano. 2015. Passwords and the Evolution of Imperfect Authentication. *Commun. ACM* 58, 7 (2015), 78–87. DOI:<http://dx.doi.org/10.1145/2699390>
- [36] Joseph Bonneau and Stuart Schechter. 2014. Towards Reliable Storage of 56-bit Secrets in Human Memory. In *Proceedings of the 23rd USENIX Security Symposium*. USENIX Association, San Diego, CA, USA, 607–623. <https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/bonneau>
- [37] Joseph Bonneau and Ekaterina Shutova. 2012. Linguistic Properties of Multi-word Passphrases. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*. Vol. 7398 LNCS. Springer, 1–12. DOI:[http://dx.doi.org/10.1007/978-3-642-34638-5\\_1](http://dx.doi.org/10.1007/978-3-642-34638-5_1)
- [38] Serdar Boztas. 1999. *Entropies, Guessing, and Cryptography*. Technical Report 6. Department of Mathematics, Royal Melbourne Institute, Melbourne, Australia.



- 
- [39] Cristian Bravo-Lillo, Lorrie Faith Cranor, Julie Downs, and Saranga Komanduri. 2011. Bridging the Gap in Computer Security Warnings: A Mental Model Approach. *IEEE Security and Privacy* 9, 2 (2011), 18–26. DOI:<http://dx.doi.org/10.1109/MSP.2010.198>
- [40] Sacha Brostoff and MA Sasse. 2003. “Ten strikes and you’re out”: Increasing the number of login attempts can improve password usability. In *Proceedings of CHI 2003* (2003), 1–4. <http://discovery.ucl.ac.uk/19826/>
- [41] Sacha Brostoff and M Angela Sasse. 2000. Are Passfaces More Usable Than Passwords? A Field Trial Investigation. In *People and Computers XIV — Usability or Else!* Springer London, London, 405–424. DOI:[http://dx.doi.org/10.1007/978-1-4471-0515-2\\_27](http://dx.doi.org/10.1007/978-1-4471-0515-2_27)
- [42] Alan S. Brown, Elisabeth Bracken, Sandy Zoccoli, and King Douglas. 2004. Generating and remembering passwords. *Applied Cognitive Psychology* 18, 6 (sep 2004), 641–651. DOI:<http://dx.doi.org/10.1002/acp.1014>
- [43] R. Brunelli and D. Falavigna. 1995. Person identification using multiple cues. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17, 10 (oct 1995), 955–966. DOI:<http://dx.doi.org/10.1109/34.464560>
- [44] A. Buchoux and N.L. Clarke. 2008. Deployment of keystroke analysis on a Smartphone. In *Proceedings of 6th Australian Information Security Management Conference*. Edith Cowan University, Perth, Australia, 29–39. DOI:<http://dx.doi.org/10.4225/75/57b55a56b876a>
- [45] Bundeskriminalamt. 2016. *Cybercrime - Bundeslagebild 2016*. Technical Report. Wiesbaden. 30 pages. <https://www.bka.de/SharedDocs/Downloads/DE/Publikationen/JahresberichteUndLagebilder/Cybercrime/cybercrimeBundeslagebild2016.html>
- [46] Mark Burnett and Dave Kleiman. 2005. Perfect Passwords. *Perfect Passwords* (2005), 107–112. DOI:<http://dx.doi.org/10.1016/B978-159749041-2/50012-6>
- [47] William E. Burr, Donna F. Dodson, and W. Timothy Polk. 2004. Electronic Authentication Guideline. *Special Publication* 800, 63 (2004), 46–64. <https://csrc.nist.gov/publications/detail/sp/800-63/ver-10/archive/2004-06-30>
- [48] Xavier De Carné De Carnavalet and Mohammad Mannan. 2014. From Very Weak to Very Strong : Analyzing Password-Strength Meters. In *Proceedings of the Network and Distributed System Security Symposium (NDSS’14)*. San Diego, CA, USA, 23–26. DOI:<http://dx.doi.org/10.14722/ndss.2014.23268>
- [49] Nancy J. Carter. 2015. Graphical Passwords for Older Computer Users. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology - UIST*

- 
- '15 *Adjunct*. ACM Press, Charlotte, NC, USA, 29–32. DOI:<http://dx.doi.org/10.1145/2815585.2815593>
- [50] Claude Castelluccia, Markus Duermuth, Maximilian Golla, and Fatma Deniz. 2017. Towards Implicit Visual Memory-Based Authentication. In *Proceedings 2017 Network and Distributed System Security Symposium*. Internet Society, San Diego, CA, USA, 1–16. DOI:<http://dx.doi.org/10.14722/ndss.2017.23292>
- [51] Sonia Chiasson, Alain Forget, Robert Biddle, and P C van Oorschot. 2008. Influencing users towards better passwords: Persuasive Cued Click-Points. In *Proceedings of the 22nd British HCI Group Annual Conference on People and Computers: Culture, Creativity, Interaction*. British Computer Society, Liverpool, United Kingdom, 121–130. <http://dl.acm.org/citation.cfm?id=1531514.1531531>
- [52] Sonia Chiasson, Alain Forget, Elizabeth Stobert, P. C. van Oorschot, and Robert Biddle. 2009. Multiple password interference in text passwords and click-based graphical passwords. In *Proceedings of the 16th ACM conference on Computer and communications security - CCS '09*. ACM Press, Chicago, Illinois, USA, 500–512. DOI: <http://dx.doi.org/10.1145/1653662.1653722>
- [53] Sonia Chiasson and P. C. van Oorschot. 2015. Quantifying the security advantage of password expiration policies. *Designs, Codes and Cryptography* 77, 2-3 (dec 2015), 401–408. DOI:<http://dx.doi.org/10.1007/s10623-015-0071-9>
- [54] Sonia Chiasson, P. C. van Oorschot, and Robert Biddle. 2007. Graphical Password Authentication Using Cued Click Points. In *Proceedings of the 12th European Symposium On Research In Computer Security (ESORICS '12)*. Vol. 4734. Springer, Dresden, Germany, 359–374. DOI:[http://dx.doi.org/10.1007/978-3-540-74835-9\\_24](http://dx.doi.org/10.1007/978-3-540-74835-9_24)
- [55] Yu-Kai Chou. 2015. *Actionable gamification : beyond points, badges, and leaderboards*. CreateSpace Independent Publishing Platform. 499 pages.
- [56] Robert B. Cialdini. 2003. Crafting normative messages to protect the environment. *Current Directions in Psychological Science* 12, 4 (2003), 105–109. DOI:<http://dx.doi.org/10.1111/1467-8721.01242>
- [57] Robert B. Cialdini. 2007. *Influence: The Psychology of Persuasion*. Vol. 55. HarperCollins. 339 pages.
- [58] Ashley Colley, Tobias Seitz, Tuomas Lappalainen, Matthias Kranz, and Jonna Häkkinen. 2016. Extending the Touchscreen Pattern Lock Mechanism with Duplicated and Temporal Codes. *Advances in Human-Computer Interaction* 2016, 8762892 (2016), 1–11. DOI:<http://dx.doi.org/10.1155/2016/8762892>

- [59] Art Conklin, Glenn Dietrich, and Diane Walz. 2004. Password-based authentication: a system perspective. In *Proceedings of the 37th Annual Hawaii International Conference on System Sciences*. IEEE, Big Island, HI, USA, 1–10. DOI: <http://dx.doi.org/10.1109/HICSS.2004.1265412>
- [60] Paul T. Costa and Robert R. McCrae. 1992. Revised NEO personality inventory (NEO PI-R) and NEO five-factor inventory (NEO FFI): Professional manual. *Psychological Assessment Resources* 3 (1992), 101. DOI: <http://dx.doi.org/10.1037//1040-3590.4.1.5>
- [61] Lynne Coventry, Pam Briggs, Debora Jeske, and Aad Van Moorsel. 2014. SCENE : A Structured Means for Creating and Evaluating Behavioral Nudges in a Cyber Security Environment. In *Design, User Experience, and Usability. Theories, Methods, and Tools for Designing the User Experience* (8517 ed.), Aaron Marcus (Ed.). Springer International Publishing, 229–239. DOI: [http://dx.doi.org/10.1007/978-3-319-07668-3\\_23](http://dx.doi.org/10.1007/978-3-319-07668-3_23)
- [62] Lorrie Faith Cranor. 2008. A framework for reasoning about the human in the loop. In *Proceedings of the 1st Conference on Usability, Psychology, and Security*. USENIX Association, San Francisco, CA, USA, 1:1–1:15. <http://portal.acm.org/citation.cfm?id=1387650>
- [63] Heather Crawford. 2010. Keystroke dynamics: Characteristics and opportunities. *PST 2010: 2010 8th International Conference on Privacy, Security and Trust* (2010), 205–212. DOI: <http://dx.doi.org/10.1109/PST.2010.5593258>
- [64] CSID. 2012. *Consumer Survey: Password Habits, A study among American consumers*. Technical Report September. CSID. 10 pages. <http://www.csid.com/wp-content/uploads/2012/09/CS>
- [65] James E. Cutting and Lynn T. Kozlowski. 1977. Recognizing friends by their walk: Gait perception without familiarity cues. *Bulletin of the Psychonomic Society* 9, 5 (1977), 353–356. DOI: <http://dx.doi.org/10.3758/BF03337021>
- [66] Ioannis G. Damousis, Dimitrios Tzovaras, and Evangelos Bekiaris. 2008. Unobtrusive Multimodal Biometric Authentication: The HUMABIO Project Concept. *EURASIP Journal on Advances in Signal Processing* 2008, 1 (dec 2008), 265767. DOI: <http://dx.doi.org/10.1155/2008/265767>
- [67] Anupam Das, Joseph Bonneau, Matthew Caesar, Nikita Borisov, and XiaoFeng Wang. 2014a. The Tangled Web of Password Reuse. In *Proceedings of Network and Distributed System Security Symposium (NDSS 14)*. Internet Society, San Diego, CA, USA, 23–26. <http://www.jbonneau.com/doc/DBCW14-NDSS-tangled>
- [68] Sauvik Das, THJ Kim, LA Dabbish, and JI Hong. 2014b. The Effect of Social Influence on Security Sensitivity. In *Proceedings of the 10th Symposium On Usable Privacy and*



- 
- Security (SOUPS'14)*. 143–157. <http://cmuchimps.org/uploads/publication/paper/147/the>
- [69] Artiom Dashinsky. 2015. Why you should (not) use Emoji in your passwords. (2015). <https://medium.com/@hvost/why-you-should-not-use-emojis-in-your-passwords-b8db0607e169>
- [70] Darren Davis, Fabian Monroe, and Michael K Reiter. 2004. On user choice in graphical password schemes. In *Proceedings of the 13th conference on USENIX Security Symposium - Volume 13*. USENIX Association, San Diego, CA, USA, 1–13. <http://dl.acm.org/citation.cfm?id=1251375.1251386>
- [71] Antonella De Angeli, Mike Coutts, Lynne Coventry, Graham I. Johnson, David Cameron, and Martin H. Fischer. 2002. VIP: A Visual Approach to User Authentication. In *Proceedings of the Working Conference on Advanced Visual Interfaces - AVI '02*. ACM Press, Trento, Italy, 316–323. DOI:<http://dx.doi.org/10.1145/1556262.1556312>
- [72] Antonella De Angeli, Lynne Coventry, Graham Johnson, and Karen Renaud. 2005. Is a picture really worth a thousand words? Exploring the feasibility of graphical authentication systems. *International Journal of Human Computer Studies* 63, 1-2 (2005), 128–152. DOI:<http://dx.doi.org/10.1016/j.ijhcs.2005.04.020>
- [73] Xavier de Carné de Carnavalet and Mohammad Mannan. 2015. A Large-Scale Evaluation of High-Impact Password Strength Meters. *ACM Transactions on Information and System Security* 18, 1 (2015), 1–31. DOI:<http://dx.doi.org/10.1145/2739044>
- [74] Alexander De Luca, Alina Hang, Frederik Brudy, Christian Lindner, and Heinrich Hussmann. 2012. Touch me once and i know it's you!. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems - CHI '12*. ACM Press, Austin, TX, USA, 987–996. DOI:<http://dx.doi.org/10.1145/2207676.2208544>
- [75] Alexander De Luca, Alina Hang, Emanuel von Zezschwitz, and Heinrich Hussmann. 2015. I Feel Like I'm Taking Selfies All Day! Towards Understanding Biometric Authentication on Smartphones. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, Seoul, South Korea, 1411–1414. DOI:<http://dx.doi.org/10.1145/2702123.2702141>
- [76] Alexander De Luca, Marian Harbach, Emanuel von Zezschwitz, Max-emanuel Maurer, Bernhard Ewald Slawik, Heinrich Hussmann, and Matthew Smith. 2014. Now You See Me, Now You Don't – Protecting Smartphone Authentication from Shoulder Surfers. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*. ACM Press, Toronto, ON, Canada, 2937–2946. DOI:<http://dx.doi.org/10.1145/2556288.2557097>

- [77] Alexander De Luca, Katja Hertzschuch, and Heinrich Hussmann. 2010a. ColorPIN. In *Proceedings of the 28th international conference on Human factors in computing systems (CHI '10)*. ACM Press, Atlanta, GA, USA, 1103. DOI:<http://dx.doi.org/10.1145/1753326.1753490>
- [78] Alexander De Luca, Marc Langheinrich, and Heinrich Hussmann. 2010b. Towards Understanding ATM Security – A Field Study of Real World ATM Use. In *Proceedings of the Sixth Symposium on Usable Privacy and Security (SOUPS '10)*. ACM, Redmond, WA, USA, 1–10. DOI:<http://dx.doi.org/10.1145/1837110.1837131>
- [79] Alexander De Luca, Emanuel von Zezschwitz, Ngo Dieu Huong Nguyen, Max-Emanuel Maurer, Elisa Rubegni, Marcello Paolo Scipioni, and Marc Langheinrich. 2013. Back-of-device authentication on smartphones. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press, Paris, France, 2389–2398. DOI:<http://dx.doi.org/10.1145/2470654.2481330>
- [80] Yves Alexandre De Montjoye, Jordi Quoidbach, Florent Robic, and Alex Pentland. 2013. Predicting personality using novel mobile phone-based metrics. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* 7812 LNCS (2013), 48–55. DOI:[http://dx.doi.org/10.1007/978-3-642-37210-0\\_6](http://dx.doi.org/10.1007/978-3-642-37210-0_6)
- [81] Matteo Dell'Amico, Pietro Michiardi, and Yves Roudier. 2010. Password Strength: An Empirical Analysis. In *Proceedings of IEEE INFOCOM*. IEEE, San Diego, CA, USA, 1–9. DOI:<http://dx.doi.org/10.1109/INFOCOM.2010.5461951>
- [82] Matteo Dell'Amico and Maurizio Filippone. 2015. Monte Carlo Strength Evaluation: Fast and Reliable Password Checking. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security (CCS- '15)*. ACM, Denver, CO, USA, 158–169. DOI:<http://dx.doi.org/10.1145/2810103.2813631>
- [83] Rachna Dhamija and Adrian Perrig. 2000. Déjà Vu: A User Study Using Images for Authentication. In *Proceedings of the 9th USENIX Security Symposium*. USENIX Association, Denver, CO, USA, 45–58.
- [84] Rachna Dhamija and J. D. Tygar. 2005. The Battle Against Phishing : Dynamic Security Skins. In *Proceedings of the 2005 Symposium on Usable Privacy and Security (SOUPS '05)*. ACM, Pittsburgh, PA, USA, 77–88. DOI:<http://dx.doi.org/10.1145/1073001.1073009>
- [85] Rachna Dhamija, J. D. Tygar, and Marti Hearst. 2006. Why Phishing Works. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. 581–590. DOI:<http://dx.doi.org/10.1145/1124772.1124861>
- [86] Paul DiGioia and Paul Dourish. 2005. Social navigation as a model for usable security. In *Proceedings of the 2005 symposium on Usable privacy and security - SOUPS '05*.

---

ACM Press, Pittsburgh, PA, USA, 101–108. DOI:<http://dx.doi.org/10.1145/1073001.1073011>

- [87] Paul Dourish, Rebecca E. Grinter, Jessica Delgado de la Flor, and Melissa Joseph. 2004. Security in the wild: user strategies for managing security as an everyday, practical problem. *Personal and Ubiquitous Computing* 8, 6 (nov 2004), 391–401. DOI: <http://dx.doi.org/10.1007/s00779-004-0308-5>
- [88] Paul Dunphy, Andreas P. Heiner, and N. Asokan. 2010. A closer look at recognition-based graphical passwords on mobile devices. In *Proceedings of the Sixth Symposium on Usable Privacy and Security - SOUPS '10*. ACM Press, Redmond, WA, USA, 1. DOI: <http://dx.doi.org/10.1145/1837110.1837114>
- [89] Paul Dunphy and Jeff Yan. 2007. Do background images improve "draw a secret" graphical passwords?. In *Proceedings of the 14th ACM conference on Computer and communications security - CCS '07*. ACM Press, Alexandria, VA, USA, 36–47. DOI: <http://dx.doi.org/10.1145/1315245.1315252>
- [90] David Eargle, John Godfrey, Hsin Miao, Scott Stevenson, Richard Shay, Blase Ur, and Lorrie Cranor. 2015. Poster : You Can Do Better – Motivational Statements in Password-Meter Feedback. In *Symposium on Usable Privacy and Security (SOUPS '15)*. USENIX Association, Ottawa, Canada, 1–2.
- [91] Serge Egelman. 2013. My profile is my password, verify me! The Privacy/Convenience Tradeoff of Facebook Connect. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press, Paris, France, 2369–2378. DOI: <http://dx.doi.org/10.1145/2470654.2481328>
- [92] Serge Egelman, Joseph Bonneau, Sonia Chiasson, David Dittrich, and Stuart Schechter. 2012. It's Not Stealing If You Need It: A Panel on the Ethics of Performing Research Using Public Data of Illicit Origin. In *Proceedings of the 3rd Workshop on Ethics in Computer Security Research (WECSR '12)*, Vol. 7398 LNCS. Springer, Bonaire, Special Municipality of The Netherlands, 124–132. DOI: [http://dx.doi.org/10.1007/978-3-642-34638-5\\_11](http://dx.doi.org/10.1007/978-3-642-34638-5_11)
- [93] Serge Egelman, Adrienne Porter Felt, and David Wagner. 2013. Choice Architecture and Smartphone Privacy: There's A Price for That. In *The economics of information security and privacy*, Rainer Böhme (Ed.). Springer, 211–236. DOI: [http://dx.doi.org/10.1007/978-3-642-39498-0\\_10](http://dx.doi.org/10.1007/978-3-642-39498-0_10)
- [94] Serge Egelman, Marian Harbach, and Eyal Peer. 2016. Behavior Ever Follows Intention? A Validation of the Security Behavior Intentions Scale (SeBIS) Serge. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems - CHI '16*. ACM Press, San Jose, CA, USA, 5257–5261. DOI: <http://dx.doi.org/10.1145/2858036.2858265>

- [95] Serge Egelman, David Molnar, Nicolas Christin, Alessandro Acquisti, Cormac Herley, and Shriram Krishnamurthi. 2010. Please Continue to Hold: An empirical study on user tolerance of security delays. In *Proceedings (online) of the 9th Workshop on Economics of Information Security*. Cambridge, MA, USA.
- [96] Serge Egelman and Eyal Peer. 2015a. Predicting Privacy and Security Attitudes. *Computers and Society: The Newsletter of ACM SIGCAS* 45, 1 (2015), 22–28. DOI: <http://dx.doi.org/10.1145/2738210.2738215>
- [97] Serge Egelman and Eyal Peer. 2015b. 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 - CHI '15*. ACM Press, Seoul, South Korea, 2873–2882. DOI: <http://dx.doi.org/10.1145/2702123.2702249>
- [98] Serge Egelman and Eyal Peer. 2015c. The Myth of the Average User: Improving Privacy and Security Systems through Individualization. In *Proceedings of the New Security Paradigms Workshop (NSPW '15)*. ACM Press, Twente, The Netherlands, 16–28. DOI: <http://dx.doi.org/10.1145/2841113.2841115>
- [99] Serge Egelman, Andreas Sotirakopoulos, Ildar Muslukhov, Konstantin Beznosov, and Cormac Herley. 2013. Does My Password Go Up to Eleven?: The Impact of Password Meters on Password Selection. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, Paris, France, 2379–2388. DOI: <http://dx.doi.org/10.1145/2470654.2481329>
- [100] emogi Research. 2016. 2016 Emoji Report. (2016). <http://cdn.emogi.com/docs/reports/2016>
- [101] Timo Erdelt. 2017. *Untersuchung von Persönlichkeitsfaktoren für Passwortvorgaben*. Bachelor Thesis. Ludwig-Maximilians-Universität München.
- [102] Michael Fagan, Yusuf Albayram, Mohammad Maifi Hasan Khan, and Ross Buck. 2017. An investigation into users' considerations towards using password managers. *Human-centric Computing and Information Sciences* 7, 1 (2017), 12. DOI: <http://dx.doi.org/10.1186/s13673-017-0093-6>
- [103] Sascha Fahl, Marian Harbach, Yasemin Acar, and Matthew Smith. 2013. On The Ecological Validity of a Password Study. In *Proceedings of the Ninth Symposium on Usable Privacy and Security (SOUPS '13)*. 1–15. DOI: <http://dx.doi.org/10.1145/2501604.2501617>
- [104] David C. (Bellcore) Feldmeier and Philip R. (Bellcore) Karn. 1990. UNIX Password Security - Ten Years Later. In *Proceedings of Conference on the Theory and Application of Cryptology (CRYPTO '89) (Lecture Notes in Computer Science)*, Gilles Brassard (Ed.), Vol. 435. Springer New York, Santa Barbara, CA, USA, 44–63. DOI: <http://dx.doi.org/10.1007/0-387-34805-0>

- 
- [105] Adrienne Porter Felt, Alex Ainslie, Robert W. Reeder, Sunny Consolvo, Somas Thyagaraja, Alan Bettis, Helen Harris, and Jeff Grimes. 2015. Improving SSL Warnings: Comprehension and Adherence. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems (CHI'15)*. ACM, Seoul, South Korea, 2893–2902. DOI:<http://dx.doi.org/10.1145/2702123.2702442>
- [106] Adrienne Porter Felt, Robert W Reeder, Alex Ainslie, Helen Harris, Max Walker, Christopher Thompson, Mustafa Emre Acer, Elisabeth Morant, Sunny Consolvo, Adrienne Porter Felt, Robert W Reeder, Alex Ainslie, Helen Harris, Max Walker, Christopher Thompson, Mustafa Emre Acer, Elisabeth Morant, Sunny Consolvo, and U C Berkeley. 2016. Rethinking Connection Security Indicators. In *Proceedings of the Twelfth Symposium on Usable Privacy and Security (SOUPS '16)*. USENIX Association, Denver, CO, USA, 1–14. [RethinkingConnectionSecurityIndicatorsAdriennePorterFelt, RobertW.Reeder, AlexAinslie, HelenHarris, andMaxWalker, Google;](#) [ChristopherThompson, UniversityofCalifornia, Berkeley; MustafaEmreAcer, ElisabethMorant, andSunnyConsolvo, Googl](#)
- [107] Adrienne Porter Felt, Robert W. Reeder, Hazim Almuhammedi, and Sunny Consolvo. 2014. Experimenting at scale with google chrome's SSL warning. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*. ACM Press, Toronto, ON, Canada, 2667–2670. DOI:<http://dx.doi.org/10.1145/2556288.2557292>
- [108] Andy Field. 2005. *Discovering Statistics Using SPSS*. Vol. 2nd. Sage Publications Ltd. 779 pages. <http://www.amazon.com/Discovering-Statistics-Introducing-Statistical-Methods/dp/0761944524>
- [109] I Flechais, M Jirotko, and Deena Alghamdi. 2013. In the balance in Saudi Arabia: security, privacy and trust. *CHI '13 Extended Abstracts on Human Factors in Computing Systems* (2013), 823–828. DOI:<http://dx.doi.org/10.1145/2468356.2468503>
- [110] Ivan Flechais, Jens Riegelsberger, and Martina Angela Sasse. 2005. Divide and Conquer: The Role of Trust and Assurance in the Design of Secure Socio-Technical Systems. In *Proceedings of the New Security Paradigms Workshop (NSPW '05)*. ACM, 33–41. <http://discovery.ucl.ac.uk/19832/>
- [111] Dinei Florêncio and Cormac Herley. 2007. A Large-Scale Study of Web Password Habits. In *Proceedings of the 16th international conference on World Wide Web (WWW '07)*. ACM, 657–665. DOI:<http://dx.doi.org/10.1145/1242572.1242661>
- [112] Dinei Florêncio and Cormac Herley. 2010. Where Do Security Policies Come from?. In *Proceedings of the Sixth Symposium on Usable Privacy and Security (SOUPS '10)*. ACM, Redmond, WA, USA, 10:1—10:14. DOI:<http://dx.doi.org/10.1145/1837110.1837124>



- [113] Dinei Florêncio and Cormac Herley. 2013. Where Do All the Attacks Go? In *Economics of Information Security and Privacy III*. Springer New York, New York, NY, 13–33. DOI:[http://dx.doi.org/10.1007/978-1-4614-1981-5\\_2](http://dx.doi.org/10.1007/978-1-4614-1981-5_2)
- [114] Dinei Florêncio, Cormac Herley, and Baris Coskun. 2007. Do strong web passwords accomplish anything? *Security* (2007), 10. <http://portal.acm.org/citation.cfm?id=1361419.1361429>
- [115] Dinei Florêncio, Cormac Herley, and Paul C Van Oorschot. 2014a. An Administrator’s Guide to Internet Password Research. In *Proceedings of the 28th Large Installation System Administration Conference (LISA14)*. USENIX Association, Seattle, WA, USA, 35–52.
- [116] Dinei Florêncio, Cormac Herley, and Paul C. Van Oorschot. 2014b. Password Portfolios and the Finite-Effort User: Sustainably Managing Large Numbers of Accounts. In *Proceedings of USENIX Security Symposium*. USENIX Association, San Diego, CA, USA, 575–590. <https://www.usenix.org/system/files/conference/usenixsecurity14/sec14-paper-florencio.pdf>
- [117] Dinei Florêncio, Cormac Herley, and Paul C. Van Oorschot. 2016. Pushing on string: The Don’t Care Region of Password Strength. *Commun. ACM* 59, 11 (oct 2016), 66–74. DOI:<http://dx.doi.org/10.1145/2934663>
- [118] Mathis Florian. 2017. *Eignung von Emojipasswörtern für verschiedene Systeme*. Bachelor Thesis. Ludwig-Maximilians-Universität München.
- [119] BJ Fogg. 2009. A Behavior Model for Persuasive Design. In *Proceedings of the 4th International Conference on Persuasive Technology - Persuasive ’09*. ACM Press, Claremont, CA, USA, 1–7. DOI:<http://dx.doi.org/10.1145/1541948.1541999>
- [120] B. J. Fogg. 2003. *Persuasive Technology: Using Computers to Change What We Think and Do*. Morgan Kaufmann, San Francisco, CA, USA.
- [121] B. J. Fogg, Jonathan Marshall, Othman Laraki, Alex Osipovich, Chris Varma, Nicholas Fang, Jyoti Paul, Akshay Rangnekar, John Shon, Preeti Swani, Marissa Treinen, and Cordura Hall. 2001. What Makes Web Sites Credible ? A Report on a Large Quantitative Study. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI ’01)*. 61–68. DOI:<http://dx.doi.org/10.1145/365024.365037>
- [122] Alain Forget and Robert Biddle. 2008. Memorability of Persuasive Passwords. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI ’08)*. ACM, Florence, Italy, 3759. DOI:<http://dx.doi.org/10.1145/1358628.1358926>

- 
- [123] Alain Forget, Sonia Chiasson, and Robert Biddle. 2007a. Helping users create better passwords: Is this the right approach?. In *Proceedings of the 3rd symposium on Usable privacy and security - SOUPS '07*. ACM Press, Pittsburg, PA, USA, 151–154. DOI: <http://dx.doi.org/10.1145/1280680.1280703>
- [124] Alain Forget, Sonia Chiasson, and Robert Biddle. 2007b. Persuasion as Education for Computer Security. In *Proceedings of E-Learn: World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education*. Association for the Advancement of Computing in Education (AACE), Chesapeake, VA, 822–829.
- [125] Alain Forget, Sonia Chiasson, and Robert Biddle. 2015. Choose Your Own Authentication. In *Proceedings of the New Security Paradigms Workshop (NSPW '15)*. ACM, Twente, The Netherlands. DOI: <http://dx.doi.org/10.1145/1235>
- [126] Alain Forget, Sonia Chiasson, P C Van Oorschot, and Robert Biddle. 2008a. Improving Text Passwords Through Persuasion. In *Proceedings of the 4th Symposium on Usable Privacy and Security (SOUPS '08)*. ACM, New York, NY, USA, 1–12. DOI: <http://dx.doi.org/10.1145/1408664.1408666>
- [127] Alain Forget, Sonia Chiasson, Paul C. Van Oorschot, and Robert Biddle. 2008b. Persuasion for stronger passwords. In *Proceedings of the 3rd International Conference on Persuasive Technology for Human Well-Being*. Springer Berlin Heidelberg, Oulu, Finland, 140–150. <http://www.scs.carleton.ca/>
- [128] Marlena R Fraune, Kevin A Juang, Joel S Greenstein, Kapil Chalil Madathil, and Reshmi Koikkara. 2013. Employing User-Created Pictures to Enhance the Recall of System-Generated Mnemonic Phrases and the Security of Passwords. *Proceedings of the Human Factors and Ergonomics Society Annual Meeting* 57, 1 (sep 2013), 419–423. DOI: <http://dx.doi.org/10.1177/1541931213571091>
- [129] Steven Furnell and Rawan Esmael. 2017. Evaluating the effect of guidance and feedback upon password compliance. *Computer Fraud and Security* 2017, 1 (2017), 5–10. DOI: [http://dx.doi.org/10.1016/S1361-3723\(17\)30005-2](http://dx.doi.org/10.1016/S1361-3723(17)30005-2)
- [130] Vaibhav Garg and Jean Camp. 2013. Heuristics and Biases: Implications for Security Design. *IEEE Technology and Society Magazine* 32, 1 (2013), 73–79. DOI: <http://dx.doi.org/10.1109/MTS.2013.2241294>
- [131] Morrie Gasser. 1975. *A Random Word Generator for Pronounceable Passwords*. Technical Report. The MITRE Corporation, Bedford, Massachusetts. 193 pages. <http://www.dtic.mil/cgi-bin/GetTRDoc?AD=ADA017676>
- [132] Shirley Gaw and Edward Felten. 2005. Reuse and Recycle : Online Password Management. In *Extended Abstracts of the Symposium on Usable Privacy and Security (SOUPS '05)*. CMU Usable Privacy and Security Laboratory, Pittsburg, PA, USA, 42–43.

- [133] Shirley Gaw and Edward W. Felten. 2006. Password management strategies for online accounts. In *Proceedings of the second symposium on Usable privacy and security (SOUPS '06)*. ACM, New York, NY, USA, 44–55. DOI:<http://dx.doi.org/10.1145/1143120.1143127>
- [134] Jurijs Girtakovskis, Ken Jacobi, David Kennerley, Kiran Kumar, Grayson Milbourne, Tyler Moffitt, Cameron Palan, and Steve Snyder. 2017. *The Webroot 2017 Annual Threat Report*. Technical Report. Webroot, Broomfield, CO, USA. 24 pages. <https://s3-us-west-1.amazonaws.com/webroot-cms-cdn/8114/8883/6877/Webroot>
- [135] Jeffrey Goldberg. 2015. Unspeakable Passwords - Pronounceable or Random Words (Talk). (2015).
- [136] Maximilian Golla, Dennis Detering, and Markus Dürmuth. 2017. EmojiAuth : Quantifying the Security of Emoji-based Authentication. In *USEC 2017*. Internet Society, San Jose, CA, USA, 1–13. DOI:<http://dx.doi.org/10.14722/usec.2017.23024>
- [137] Samuel D. Gosling, Peter J. Rentfrow, and William B. Swann. 2003. A very brief measure of the Big-Five personality domains. *Journal of Research in Personality* 37, 6 (2003), 504–528. DOI:[http://dx.doi.org/10.1016/S0092-6566\(03\)00046-1](http://dx.doi.org/10.1016/S0092-6566(03)00046-1)
- [138] Jeff Gothelf and Josh Seiden. 2013. *Lean UX*. 1–151 pages. DOI:<http://dx.doi.org/10.1017/CB09781107415324.004>
- [139] C L Grady, A R McIntosh, M N Rajah, and F I Craik. 1998. Neural correlates of the episodic encoding of pictures and words. *Proceedings of the National Academy of Sciences USA* 95, 5 (1998), 2703–2708. DOI:<http://dx.doi.org/10.1073/pnas.95.5.2703>
- [140] Adam M Grant. 2013. Rethinking the Extraverted Sales Ideal: The Ambivert Advantage. *Psychological Science* 24, 6 (jun 2013), 1024–1030. DOI:<http://dx.doi.org/10.1177/0956797612463706>
- [141] Rachel Greenstadt and Jacob Beal. 2008. Cognitive Security for Personal Devices. In *Proceedings of the 1st ACM workshop on Workshop on AISEC - AISEC '08*. ACM Press, Alexandria, VA, USA, 27–30. DOI:<http://dx.doi.org/10.1145/1456377.1456383>
- [142] Thomas Groß, Kovila Coopamootoo, and Amina Al-jabri. 2016a. *Effect of Cognitive Depletion on Password Choice*. Technical Report September. Newcastle University, Newcastle, UK. 1–16 pages. <https://www.usenix.org/system/files/conference/soups2016/way>
- [143] Thomas Groß, Kovila P.L. Coopamootoo, and Amina Al-Jabri. 2016b. Effect of Cognitive Effort on Password Choice. In *Symposium on Usable Privacy and Security - Posters*. USENIX Association, Denver, CO, USA, 1–2.



- 
- [144] Iwan Gulenko. 2014. Improving passwords: influence of emotions on security behaviour. *Information Management & Computer Security* 22, 2 (2014), 167–178. DOI: <http://dx.doi.org/10.1108/IMCS-09-2013-0068>
- [145] Yimin Guo and Zhenfeng Zhang. 2017. LPSE: lightweight password-strength estimation for password meters. *Computers & Security* (2017). DOI: <http://dx.doi.org/10.1016/j.cose.2017.07.012>
- [146] Hana Habib, Jessica Colnago, William Melicher, Blase Ur, Sean M. Segreti, Lujo Bauer, Nicolas Christin, and Lorrie Cranor. 2017. Password Creation in the Presence of Blacklists. In *Proceedings of the 2017 Workshop on Usable Security*. Internet Society, San Diego, CA, USA, 11. DOI: <http://dx.doi.org/10.14722/usec.2017.23043>
- [147] J Alex Halderman, Brent Waters, and Edward W Felten. 2005. A convenient method for securely managing passwords. In *Proceedings of the 14th international conference on World Wide Web - WWW '05*. ACM Press, Chiba, Japan, 471. DOI: <http://dx.doi.org/10.1145/1060745.1060815>
- [148] Tzipora Halevi, James Lewis, and Nasir Memon. 2013. A pilot study of cyber security and privacy related behavior and personality traits. In *WWW 2013 Companion - Proceedings of the 22nd International Conference on World Wide Web*. ACM, Rio de Janeiro, Brazil, 737–744. DOI: <http://dx.doi.org/10.1145/2487788.2488034>
- [149] Tzipora Halevi, Nasir Memon, and Oded Nov. 2015. Spear-Phishing in the Wild: A Real-World Study of Personality, Phishing Self-Efficacy and Vulnerability to Spear-Phishing Attacks. *SSRN Electronic Journal* (2015). DOI: <http://dx.doi.org/10.2139/ssrn.2544742>
- [150] Juho Hamari, Jonna Koivisto, and Tuomas Pakkanen. 2014. Do persuasive technologies persuade? - A review of empirical studies. In *Lecture Notes in Computer Science*, Vol. 8462 LNCS. 118–136. DOI: [http://dx.doi.org/10.1007/978-3-319-07127-5\\_11](http://dx.doi.org/10.1007/978-3-319-07127-5_11)
- [151] Alina Hang. 2015. *Exploiting Autobiographical Memory for Fallback Authentication on Smartphones*. Dissertation. Ludwig-Maximilians-Universität München.
- [152] Alina Hang, Alexander De Luca, Katharina Frison, Emanuel von Zezschwitz, Massimo Tedesco, Marcel Kockmann, and Heinrich Hussmann. 2013. Travel Routes or Geography Facts? An Evaluation of Voice Authentication User Interfaces. In *Proceedings of INTERACT*, Vol. 8119 LNCS. Springer, Cape Town, South Africa, 468–475. DOI: [http://dx.doi.org/10.1007/978-3-642-40477-1\\_29](http://dx.doi.org/10.1007/978-3-642-40477-1_29)
- [153] SMT Haque, Shannon Scielzo, and Matthew Wright. 2014a. Applying Psychometrics to Measure User Comfort when Constructing a Strong Password. In *Symposium on Usable Privacy and Security (SOUPS)*. Menlo Park, CA, USA, 231–242. <https://www.usenix.org/system/files/conference/soups2014/soups14-paper-haque.pdf>

- [154] S. M Taiabul Haque, Matthew Wright, and Shannon Scielzo. 2014b. Hierarchy of users' web passwords: Perceptions, practices and susceptibilities. *International Journal of Human Computer Studies* 72, 12 (2014), 860–874. DOI:<http://dx.doi.org/10.1016/j.ijhcs.2014.07.007>
- [155] Marian Harbach, Emanuel von Zezschwitz, Andreas Fichtner, Alexander De Luca, and Matthew Smith. 2014. It's a Hard Lock Life: A Field Study of Smartphone (Un)Locking Behavior and Risk Perception. In *SOUPS '14: Proceedings of the Tenth Symposium On Usable Privacy and Security*. 213–230. <https://www.usenix.org/conference/soups2014/proceedings/presentation/harbach>
- [156] Garrett Hardin. 1968. The Tragedy of the Commons. *Science* 162, 3859 (1968), 1243–8. DOI:<http://dx.doi.org/10.1126/science.162.3859.1243>
- [157] James A Haskett. 1984. Pass-algorithms: a user validation scheme based on knowledge of secret algorithms. *Commun. ACM* 27, 8 (1984), 777–781. DOI:<http://dx.doi.org/10.1145/358198.358214>
- [158] Eiji Hayashi, Rachna Dhamija, Nicolas Christin, and Adrian Perrig. 2008. Use Your Illusion. In *Proceedings of the 4th symposium on Usable privacy and security - SOUPS '08*. ACM Press, Pittsburgh, PA, USA, 35–45. DOI:<http://dx.doi.org/10.1145/1408664.1408670>
- [159] Eiji Hayashi and Jason Hong. 2011. A diary study of password usage in daily life. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, Vancouver, BC, Canada, 2627–2631. DOI:<http://dx.doi.org/10.1145/1978942.1979326>
- [160] EB Hekler, Predrag Klasnja, JE Froehlich, and MP Buman. 2013. Mind the Theoretical Gap: Interpreting, Using, and Developing Behavioral Theory in HCI Research. <http://www.designinghealth.org/uploads/1/3/8/4/13844497/hci>
- [161] Olaf Henniger, Dirk Scheuermann, and Thomas Kniess. On security evaluation of fingerprint recognition systems. In *Proceedings of the International Biometric Performance Testing Conference (IBPC)*. NIST, Gaithersburg, MD, USA.
- [162] Cormac Herley. 2009. So Long, And No Thanks for the Externalities: The Rational Rejection of Security Advice by Users. In *Proceedings of the New Security Paradigms Workshop (NSPW '09)*. ACM, Oxford, United Kingdom, 133–144. DOI:<http://dx.doi.org/10.1145/1719030.1719050>
- [163] Cormac Herley. 2014. Security, cybercrime, and scale. *Commun. ACM* 57, 9 (sep 2014), 64–71. DOI:<http://dx.doi.org/10.1145/2654847>
- [164] Cormac Herley and Wolter Pieters. 2015. "If you were attacked, you'd be sorry": Counterfactuals as security arguments. In *Proceedings of the New Security Paradigms*

- 
- Workshop on ZZZ - NSPW '15*. ACM Press, Twente, The Netherlands, 112–123. DOI: <http://dx.doi.org/10.1145/2841113.2841122>
- [165] Cormac Herley and Paul Van Oorschot. 2012. A Research Agenda Acknowledging the Persistence of Passwords. *IEEE Security and Privacy* 10, 1 (2012), 28–36. DOI: <http://dx.doi.org/10.1109/MSP.2011.150>
- [166] Moritz Horsch, Mario Schlipf, Stefen Haas, Johannes Braun, and Johannes Buchmann. 2016. Password Policy Markup Language. In *Proceedings of Open Identify Summit*. Gesellschaft für Informatik, Rome, Italy, 135–147.
- [167] Joel Huber, John W. Payne, and Christopher Puto. 1982. Adding Asymmetrically Dominated Alternatives: Violations of Regularity and the Similarity Hypothesis. *Journal of Consumer Research* 9, 1 (1982), 90. DOI: <http://dx.doi.org/10.1086/208899>
- [168] Paul Huber. 2016. *Einfluss des Persönlichkeitstyps auf die Wahrnehmung von Passwortkomplexität*. Bachelor Thesis. Ludwig-Maximilians-Universität München.
- [169] Jun Ho Huh, Hyoungshick Kim, Swathi S.V.P. Rayala, Rakesh B. Bobba, and Konstantin Beznosov. 2017. I’m too Busy to Reset my LinkedIn Password: On the Effectiveness of Password Reset Emails. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press, Denver, CO, USA, 387–391. DOI: <http://dx.doi.org/10.1145/3025453.3025788>
- [170] Jun Ho Huh, Seongyeol Oh, Hyoungshick Kim, Konstantin Beznosov, Apurva Mohan, and S. Raj Rajagopalan. 2015. Surpass: System-initiated User-replaceable Passwords. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security - CCS '15*. ACM Press, Denver, CO, USA, 170–181. DOI: <http://dx.doi.org/10.1145/2810103.2813622>
- [171] Ahsan Imran. 2015. *A Comparison of Password Authentication Between Children and Adults*. Master Thesis. Carleton University, Ottawa, Ontario.
- [172] Philip Inglesant and Martina Angela Sasse. 2010. The True Cost of Unusable Password Policies: Password Use in the Wild. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. 383–392. DOI: <http://dx.doi.org/10.1145/1753326.1753384>
- [173] Blake Ives, Kenneth R. Walsh, and Helmut Schneider. 2004. The Domino Effect of Password Reuse. *Commun. ACM* 47, 4 (apr 2004), 75–78. DOI: <http://dx.doi.org/10.1145/975817.975820>
- [174] Sheena S. Iyengar and Mark R. Lepper. 2000. When Choice is Demotivating: Can One Desire Too Much of a Good thing? *Journal of Personality and Social Psychology* 79, 6 (2000), 995–1006. DOI: <http://dx.doi.org/10.1037/0022-3514.79.6.995>

- [175] David Jaeger, Chris Pelchen, Hendrik Graupner, Feng Cheng, and Christoph Meinel. 2016. Analysis of Publicly Leaked Credentials and the Long Story of Password (Re-)use. In *Proceedings of the 11th International Conference on Passwords (PASS-WORDS2016)*. Springer, Bochum, Germany, 1–19.
- [176] Markus Jakobsson. 2014. *How to Wear Your Password*. Technical Report. Qualcomm Research. <http://www.markus-jakobsson.com/wp-content/uploads/WP-us-14-Jakobsson-HowToWearYourPassword.pdf>
- [177] Markus Jakobsson and Mayank Dhiman. 2013. The Benefits of Understanding Passwords. In *Mobile Authentication* (2 ed.). Springer, 5–24. DOI:[http://dx.doi.org/10.1007/978-1-4614-4878-5\\_2](http://dx.doi.org/10.1007/978-1-4614-4878-5_2)
- [178] Markus Jakobsson, Elaine Shi, Philippe Golle, and Richard Chow. 2009. Implicit authentication for mobile devices. In *Proceedings of the 4th USENIX conference on Hot topics in security (HotSec'09)*. USENIX Association, Montreal, Canada, 1–6.
- [179] Anthony Jameson, Silvia Gabrielli, Per Ola Kristensson, Katharina Reinecke, Federica Cena, Cristina Gena, and Fabiana Vernero. 2011. How can we support users' preferential choice? *Proceedings of the 2011 annual conference extended abstracts on Human factors in computing systems - CHI EA '11* (2011), 409. DOI:<http://dx.doi.org/10.1145/1979742.1979620>
- [180] Ian Jermyn, Alain Mayer, Fabian Monrose, Michael K Reiter, and Aviel D Rubin. 1999. The Design and Analysis of Graphical Passwords. In *Proceedings of the 8th USENIX Security Symposium*, Vol. 8. USENIX Association, Washington, DC, USA, 1–14. DOI:<http://dx.doi.org/10.1109/ICCIIS.2010.35>
- [181] Debora Jeske, Lynne Coventry, and Pam Briggs. 2014. Nudging whom how : IT proficiency , impulse control and secure behaviour. In *Proceedings of the CHI Workshop on Personalizing Behavior Change Technologies*. Toronto, ON, Canada, 1–4.
- [182] Daniel Kahneman. 2003. Maps of bounded rationality: Psychology for behavioral economics. *American Economic Review* 93, 5 (2003), 1449–1475. DOI:<http://dx.doi.org/10.1257/00028280322655392>
- [183] Daniel Kahneman. 2011. *Thinking, fast and slow*. 499 pages. <https://books.google.de/books?id=ZuKTvERuPG8C>
- [184] Daniel Kahnemann and Shane Frederick. 2002. Heuristics of Intuitive Judgment: Extensions and Applications. In *Heuristics of Intuitive Judgment: Extensions and Applications*, D. Griffin T. Gilovich and D. Kahneman (Eds.). Cambridge University Press, New York, New York, USA, 1–30.
- [185] Ruogu Kang, Laura Dabbish, Nathaniel Fruchter, and Sara Kiesler. 2015. "My data just goes everywhere": User mental models of the internet and implications for privacy and security. In *Proceedings of the Symposium on Usable Privacy and Security*

- 
- (SOUPS '15). USENIX Association, Ottawa, Canada, 39–52. <https://www.usenix.org/system/files/conference/soups2015/soups15-paper-kang.pdf>
- [186] Kaspersky. 2016. Consumer Security Risks Survey 2015. *Kaspersky Lab* (2016). <https://press.kaspersky.com/files/2015/08/Kaspersky>
- [187] Christina Katsini, Nikolaos Avouris, Christos Fidas, George Samaras, and Marios Belk. 2017. Influences of Users' Cognitive Strategies on Graphical Password Composition. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems (CHI EA '17)*. ACM, Denver, CO, USA, 2698–2705. DOI:<http://dx.doi.org/10.1145/3027063.3053217>
- [188] Joseph 'Jofish' Kaye. 2011. Self-reported password sharing strategies. In *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*. ACM Press, Vancouver, BC, Canada, 2619. DOI:<http://dx.doi.org/10.1145/1978942.1979324>
- [189] Mark Keith, Benjamin Shao, and Paul Steinbart. 2009. A Behavioral Analysis of Passphrase Design and Effectiveness. *Journal of the Association for Information Systems* 10, 2 (2009), 63–89. <http://aisel.aisnet.org/cgi/viewcontent.cgi?article=1492>
- [190] Patrick Gage Kelley, Saranga Komanduri, Michelle L. Mazurek, Richard Shay, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Julio Lopez. 2012a. Guess Again (and Again and Again): Measuring Password Strength by Simulating Password-Cracking Algorithms. In *2012 IEEE Symposium on Security and Privacy*. IEEE, San Francisco, CA, USA, 523–537. DOI:<http://dx.doi.org/10.1109/SP.2012.38>
- [191] Patrick Gage Kelley, Saranga Komanduri, Michelle L. Mazurek, Richard Shay, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Julio López. 2012b. Guess again (and again and again): Measuring password strength by simulating password-cracking algorithms. In *Proceedings - IEEE Symposium on Security and Privacy*. 523–537. DOI:<http://dx.doi.org/10.1109/SP.2012.38>
- [192] Limor Kessem. 2018. *IBM Security : Future of Identity Study*. Technical Report. IBM Security, Cambridge, MA, USA. 27 pages. [ibm.com/security](http://ibm.com/security)
- [193] Warut Khern-am nuai, Weining Yang, and Ninghui Li. 2017a. Designing Better Password Strength Meters by Incorporating Contextual Information. *SSRN Electronic Journal* 2017, 05 (2017). DOI:<http://dx.doi.org/10.2139/ssrn.2800499>
- [194] Warut Khern-am nuai, Weining Yang, and Ninghui Li. 2017b. Using Context-Based Password Strength Meter to Nudge Users' Password Generating Behavior: A Randomized Experiment. In *SSRN Electronic Journal*. 27. DOI:<http://dx.doi.org/10.24251/HICSS.2017.071>



- [195] Ran Kivetz, Oleg Urminsky, and Yuhuang Zheng. 2006. The Goal-Gradient Hypothesis Resurrected: Purchase Acceleration, Illusionary Goal Progress, and Customer Retention. *Journal of Marketing Research* 43, 1 (2006), 39–58. DOI:<http://dx.doi.org/10.1509/jmkr.43.1.39>
- [196] Bart P Knijnenburg, Alfred Kobsa, and Hongxia Jin. 2013. Preference-based location sharing: Are More Privacy Options Really Better?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press, Paris, France, 2667–2676. DOI:<http://dx.doi.org/10.1145/2470654.2481369>
- [197] John Kohl and Clifford Neuman. 1993. The Kerberos Network Authentication Service (V5). (1993). <http://www.rfc-editor.org/rfc/rfc1510.txt>
- [198] Saranga Komanduri. 2016. *Modeling the Adversary to Evaluate Password Strength With Limited Samples*. Dissertation. Carnegie Mellon University.
- [199] Saranga Komanduri, Richard Shay, Lorrie Faith Cranor, Cormac Herley, and Stuart Schechter. 2014. Telepathwords: Preventing Weak Passwords by Reading Users' Minds. In *23rd USENIX Security Symposium (USENIX Security 14)*. USENIX Association, San Diego, CA, USA, 591–606. <https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/komanduri>
- [200] Saranga Komanduri, Richard Shay, Patrick Gage Kelley, Michelle L Mazurek, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Serge Egelman. 2011. Of passwords and people. In *Proceedings of the 2011 annual conference on Human factors in computing systems - CHI '11*. 2595. DOI:<http://dx.doi.org/10.1145/1978942.1979321>
- [201] Stefan Korff and Rainer Böhme. 2014. Too Much Choice: End-User Privacy Decisions in the Context of Choice Proliferation. In *Symposium on Usable Privacy and Security (SOUPS '14)*. 69–87. <https://www.usenix.org/system/files/soups14-paper-korff.pdf>
- [202] Vijay Kothari, Ross Koppel, Jim Blythe, and Sean Smith. 2017. Password Logbooks and What Their Amazon Reviews Reveal About Their Users' Motivations, Beliefs, and Behaviors. In *Proceedings 2nd European Workshop on Usable Security*. Internet Society, Paris, France, 10. DOI:<http://dx.doi.org/10.14722/eurosec.2017.23018>
- [203] Lydia Kraus, Robert Schmidt, Marcel Walch, Florian Schaub, and Sebastian Möller. 2017. On the Use of Emojis in Mobile Authentication. In *IFIP Advances in Information and Communication Technology*. Vol. 502. Springer, Cham, 265–280. DOI:[http://dx.doi.org/10.1007/978-3-319-58469-0\\_18](http://dx.doi.org/10.1007/978-3-319-58469-0_18)
- [204] Christien Kroeze and Martin S. Olivier. 2012. Gamifying authentication. In *2012 Information Security for South Africa*. IEEE, Johannesburg, Gauteng, South Africa, 1–8. DOI:<http://dx.doi.org/10.1109/ISSA.2012.6320439>

- 
- [205] Kat Krol, Jonathan M Spring, Simon Parkin, and M Angela Sasse. 2016. Towards robust experimental design for user studies in security and privacy. In *Learning from Authoritative Security Experiment Results (LASER '16)*. USENIX Association, 21–32.
- [206] Cynthia Kuo, Sasha Romanosky, and Lorrie Faith Cranor. 2006. Human Selection of Mnemonic Phrase-Based Passwords. In *Proceedings of the second Symposium on Usable Privacy and Security (SOUPS '06)*. Pittsburg, PA, USA, 67–78. DOI:<http://dx.doi.org/10.1145/1143120.1143129>
- [207] Stanley A. Kurzban. 1985. Easily Remembered Passphrases - A Better Approach. *ACM SIGSAC Review* 3, 2-4 (sep 1985), 10–21. DOI:<http://dx.doi.org/10.1145/1058406.1058408>
- [208] LastPass. 2016. *The Password Paradox and why our Personalities will get us Hacked*. Technical Report. 1–6 pages. <http://prod.cdata.app.sprinklr.com/DAM/434/LastPass>
- [209] Yue Li, Haining Wang, and Kun Sun. 2017. Personal Information in Passwords and Its Security Implications. *IEEE Transactions on Information Forensics and Security* 12, 10 (oct 2017), 2320–2333. DOI:<http://dx.doi.org/10.1109/TIFS.2017.2705627>
- [210] Zhiwei Li, Warren He, Devdatta Akhawe, and Dawn Song. 2014. The Emperor's New Password Manager: Security Analysis of Web-based Password Managers. In *Proceedings of the 23rd USENIX Security Symposium (USENIX Security 14)*. USENIX Association, San Diego, CA, USA, 465–479. <http://devd.me/papers/pwdmgr-usenix14.pdf>
- [211] William Lidwell, Kritina Holden, and Jill Butler. 2003. *Universal Principles of Design*. Vol. 2007. Rockport Publishers. 216 pages. DOI:<http://dx.doi.org/10.1007/s11423-007-9036-7>
- [212] Eric Lin, Saul Greenberg, Eileah Trotter, David Ma, and John Aycock. 2011. Does domain highlighting help people identify phishing sites?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '11)*. ACM, New York, NY, USA, 2075–2084. DOI:<http://dx.doi.org/10.1145/1978942.1979244>
- [213] David Llewellyn-jones and Graham Rymer. 2016. Cracking PwdHash: A Bruteforce Attack on Client-side Password Hashing. In *Proceedings of the 11th International Conference on Passwords*. Springer, Bochum, Germany, 1–19.
- [214] Dan Lockton. 2012. Cognitive Biases, Heuristics and Decision-Making in Design for Behaviour Change. (2012). <http://papers.ssrn.com/sol3/papers.cfm?abstract>
- [215] Dan Lockton, David Harrison, and Neville a Stanton. 2010. The Design with Intent Method: A design tool for influencing user behaviour. *Applied Ergonomics* 41, 3 (may 2010), 382–392. DOI:<http://dx.doi.org/10.1016/j.apergo.2009.09.001>



- [216] Ijlal Loutfi and Audun Jøsang. 2015. Passwords are not always stronger on the other side of the fence. In *Proceedings of the Network and Distributed System Security Conference, USEC Workshop*. Internet Society, San Diego, CA, USA, 1–10. DOI: <http://dx.doi.org/10.14722/usec.2015.23005>
- [217] Sanam Ghorbani Lyastani, Michael Schilling, Sascha Fahl, Sven Bugiel, and Michael Backes. 2017. *Studying the Impact of Managers on Password Strength and Reuse*. Technical Report. 1–20 pages. <http://arxiv.org/abs/1712.08940>
- [218] Joseph Maguire and Karen Renaud. 2012. You Only Live Twice or The Years We Wasted Caring about Shoulder-Surfing. In *Proceedings of the 26th Annual BCS Interaction Specialist Group Conference on People and Computers (BCS-HCI '12)*. BISL / ACM, Birmingham, UK, 404–409. <http://eprints.gla.ac.uk/71011/>
- [219] Nathan Malkin, Shriram Krishnamurthi, and David H. Laidlaw. 2013. Waiting Makes the Heart Grow Fonder and the Password Grow Stronger. In *Symposium on Usable Privacy and Security (SOUPS) - Posters*. USENIX Association, Newcastle, UK, 1–2. <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.364.14>
- [220] Fatma AL Maqbali and Chris J Mitchell. 2016. Password Generators: Old Ideas and New. September (2016), 1–20. <http://arxiv.org/abs/1607.04421>
- [221] Emanuela Marasco and Arun Ross. 2014. A Survey on Antispoofing Schemes for Fingerprint Recognition Systems. *Comput. Surveys* 47, 2 (nov 2014), 1–36. DOI: <http://dx.doi.org/10.1145/2617756>
- [222] Simon Marechal. 2008. Advances in password cracking. *Journal in Computer Virology* 4, 1 (2008), 73–81. DOI: <http://dx.doi.org/10.1007/s11416-007-0064-y>
- [223] Davide Marengo, Fabrizia Giannotta, and Michele Settanni. 2017. Assessing personality using emoji: An exploratory study. *Personality and Individual Differences* 112, July (2017), 74–78. DOI: <http://dx.doi.org/10.1016/j.paid.2017.02.037>
- [224] Max-emanuel Maurer, Alexander De Luca, and Sylvia Kempe. 2011a. Using Data Type Based Security Alert Dialogs to Raise Online Security Awareness. In *SOUPS '11 Proceedings of the Seventh Symposium on Usable Privacy and Security*. ACM, Pittsburgh, PA, USA, Paper 2. DOI: <http://dx.doi.org/10.1145/2078827.2078830>
- [225] Max-Emanuel Maurer, Alexander De Luca, and Tobias Stockinger. 2011b. Shining Chrome: Using Web Browser Personas to Enhance SSL Certificate Visualization. In *Proceedings of the international conference on Human-computer interaction (INTERACT'11)*. Springer, Berlin, Heidelberg, 44–51. <http://link.springer.com/chapter/10.1007/978-3-642-23768-3>
- [226] Michelle L. Mazurek, Saranga Komanduri, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Patrick Gage Kelley, Richard Shay, and Blase Ur. 2013.

- 
- Measuring password guessability for an entire university. In *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security - CCS '13*. ACM Press, New York, New York, USA, 173–186. DOI:<http://dx.doi.org/10.1145/2508859.2516726>
- [227] Daniel McCarney. 2013. *Password Managers: Comparative Evaluation, Design, Implementation and Empirical Analysis*. Ph.D. Dissertation. Carleton University.
- [228] Daniel McCarney, David Barrera, Jeremy Clark, Sonia Chiasson, and Paul C. van Oorschot. 2012. Tapas: Design, Implementation, and Usability Evaluation of a Password Manager. In *Proceedings of the 28th Annual Computer Security Applications Conference on - ACSAC '12*. ACM Press, Orlando, FL, USA, 89–99. DOI:<http://dx.doi.org/10.1145/2420950.2420964>
- [229] Robert R. McCrae and Paul T. Costa. 1987. Validation of the Five-Factor Model of Personality Across Instruments and Observers. *Journal of Personality and Social Psychology* 52, 1 (1987), 81–90.
- [230] Pete McEvoy and Jeremiah D Still. 2016. Contextualizing Mnemonic Phrase Passwords. In *Proceedings of the AHFE 2016 International Conference on Human Factors in Cybersecurity*. Springer, Cham, Orlando, FL, USA, 295–304. DOI:[http://dx.doi.org/10.1007/978-3-319-41932-9\\_24](http://dx.doi.org/10.1007/978-3-319-41932-9_24)
- [231] Stefanie Meitner. 2016. *Der Effekt von Passwort-Generatoren auf die Selektion von Passwörtern*. Bachelor Thesis. Ludwig-Maximilians-Universität München.
- [232] William Melicher, Darya Kurilova, Sean M. Segreti, Pranshu Kalvani, Richard Shay, Blase Ur, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Michelle L Mazurek. 2016a. Usability and Security of Text Passwords on Mobile Devices. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*. 527–539. DOI:<http://dx.doi.org/10.1145/2858036.2858384>
- [233] William Melicher, Blase Ur, Sean M. Segreti, Saranga Komanduri, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2016b. Fast, Lean, and Accurate: Modeling Password Guessability Using Neural Networks. In *Proceedings of the 25th USENIX Security Symposium*. USENIX Association, Austin, TX, USA, 175–191.
- [234] Hannah Miller, Jacob Thebault-Spieker, Shuo Chang, Isaac Johnson, Loren Terveen, and Brent Hecht. 2015. “Blissfully happy” or “ready to fight”: Varying Interpretations of Emoji. *GroupLens Research, University of Minnesota* (2015).
- [235] Tehila Minkus and Nasir Memon. 2014. Leveraging Personalization to Facilitate Privacy. (2014). <http://papers.ssrn.com/abstract=2448026>
- [236] Kevin D. Mitnick and William L. Simon. 2002. *The Art of Deception: Controlling the Human Element in Security* (1st editio ed.). Wiley. 352 pages. DOI:<http://dx.doi.org/0471237124>

- [237] Robert Morris and Ken Thompson. 1979. Password Security: A Case History. *Commun. ACM* 22, 11 (1979), 594–597. DOI:<http://dx.doi.org/10.1145/359168.359172>
- [238] Nicole L. Muscanell, Rosanna E. Guadagno, and Shannon Murphy. 2014. Weapons of influence misused: A social influence analysis of why people fall prey to internet scams. *Social and Personality Psychology Compass* 8, 7 (2014), 388–396. DOI:<http://dx.doi.org/10.1111/spc3.12115>
- [239] Arvind Narayanan and Vitaly Shmatikov. 2005. Fast Dictionary Attacks on Passwords Using Time-Space Tradeoff. In *Proceedings of the 12th ACM conference on Computer and communications security - CCS '05*. ACM, Alexandria, VA, USA, 364–372. DOI: <http://dx.doi.org/10.1145/1102120.1102168>
- [240] Mohammad Nauman and Tamleek Ali. 2010. TOKEN: Trustable Keystroke-Based Authentication for Web-Based Applications on Smartphones. In *Communications in Computer and Information Science*. Vol. 76 CCIS. Springer Berlin Heidelberg, 286–297. DOI:[http://dx.doi.org/10.1007/978-3-642-13365-7\\_28](http://dx.doi.org/10.1007/978-3-642-13365-7_28)
- [241] Aline Neumann. 2017. *Effects of Personality on Password Selection*. Bachelor Thesis. Ludwig-Maximilians-Universität München.
- [242] Jakob Nielsen. 1994. Enhancing the explanatory power of usability heuristics. In *Proceedings of the SIGCHI conference on Human factors in computing systems celebrating interdependence - CHI '94*. ACM Press, Boston, MA, USA, 152–158. DOI: <http://dx.doi.org/10.1145/191666.191729>
- [243] Chris Nodder. 2013. *Evil By Design* (1 ed.). Wiley, Indianapolis, IN, USA. 322 pages.
- [244] Don Norman. 1983. Some Observations on Mental Models. In *Mental Models*. Psychology Press, Chapter 1, 7–14.
- [245] Gilbert Notoatmodjo. 2007. *Exploring the ‘ Weakest Link ’: A Study of Personal Password Security*. Master Thesis. University of Auckland, New Zealand.
- [246] Harri Oinas-Kukkonen. 2013. A foundation for the study of behavior change support systems. *Personal and Ubiquitous Computing* 17, 6 (2013), 1223–1235. DOI:<http://dx.doi.org/10.1007/s00779-012-0591-5>
- [247] Kenneth Olmstead and Aaron Smith. 2017. *Americans and Cybersecurity*. Technical Report. Pew Research Center. 42 pages. <http://assets.pewresearch.org/wp-content/uploads/sites/14/2017/01/26102016/Americans-and-Cyber-Security-final.pdf>
- [248] Caroline Olsienkiewicz. 2016. *Verbesserung von verbalem Echtzeitfeedback bei der Passwortselektion*. Bachelor Thesis. Ludwig-Maximilians-Universität München.

- 
- [249] Daniel M. Oppenheimer, Tom Meyvis, and Nicolas Davidenko. 2009. Instructional manipulation checks: Detecting satisficing to increase statistical power. *Journal of Experimental Social Psychology* 45, 4 (2009), 867–872. DOI:<http://dx.doi.org/10.1016/j.jesp.2009.03.009>
- [250] J R B Paiva, V M Gomes, and C Morris. 2017. Passfault : an Open Source Tool for Measuring Password Complexity and Strength. In *Proceedings of the 8th International Multi-Conference on Complexity, Informatics and Cybernetics (IMCIC '17)*. OWASP, Orlando, FL, USA. <https://www.owasp.org/images/1/13/Artigo-Passfault.pdf>
- [251] Allan Paivio, T.B. Rogers, and Padric C. Smythe. 1968. Why are Pictures Easier to Recall Than Words ? *Psychonomic Science* 11, 4 (1968), 137–138. DOI:<http://dx.doi.org/10.3758/BF03331011>
- [252] J.L. Parrish Jr., J.L. Bailey, and J.F. Courtney. 2009. A Personality Based Model for Determining Susceptibility to Phishing Attacks. *Southwest Decision Sciences Institute (SWDSI) annual meeting* October 2015 (2009), 285–296.
- [253] Eyal Peer, Laura Brandimarte, Sonam Samat, and Alessandro Acquisti. 2017. Beyond the Turk: Alternative platforms for crowdsourcing behavioral research. *Journal of Experimental Social Psychology* 70 (2017), 153–163. DOI:<http://dx.doi.org/10.1016/j.jesp.2017.01.006>
- [254] Sean Peisert, Ed Talbot, and Tom Kroeger. 2013. Principles of authentication. In *Proceedings of the 2013 workshop on New security paradigms workshop - NSPW '13*. ACM Press, Banff, Alberta, Canada, 47–56. DOI:<http://dx.doi.org/10.1145/2535813.2535819>
- [255] John O. Pliam. 2000. On the Incomparability of Entropy and Marginal Guesswork in Brute-Force Attacks. In *Proceedings of Progress in Cryptology - INDOCRYPT 2000*. Springer Berlin Heidelberg, Calcutta, India, 67–79. DOI:[http://dx.doi.org/10.1007/3-540-44495-5\\_7](http://dx.doi.org/10.1007/3-540-44495-5_7)
- [256] Martin Prinz. 2017. *Developing a Secure Password-Reuse-Manager*. Master Thesis. Ludwig-Maximilians-Universität München.
- [257] Martin Prinz and Tobias Seitz. 2017. Towards a Mental Model of Password Management Software. In *Extended Abstracts of the Symposium on Usable Privacy and Security (SOUPS EA 2017)*. USENIX Association, Santa Clara, CA, USA, 6.
- [258] Robert W Proctor, Mei-Ching Lien, Kim-Phuong L Vu, E Eugene Schultz, and Gavriel Salvendy. 2002. Improving Computer Security for Authentication of Users: Influence of Proactive Password Restrictions. *Behavior Research Methods, Instruments, & Computers: A Journal of the Psychonomic Society, Inc* 34, 2 (2002), 163–169. DOI: <http://dx.doi.org/10.3758/BF03195438>

- [259] Niels Provos and David Mazieres. 1999. A future-adaptable password scheme. In *Proceedings of the USENIX Annual Technical Conference*. USENIX Association, Monterey, CA, USA, 1–12. <https://www.usenix.org/legacy/event/usenix99/full>
- [260] Kenneth Radke, Colin Boyd, Juan Gonzalez Nieto, and Laurie Buys. 2013. "Who decides?" Security and Privacy in the Wild. In *Proceedings of the 25th Australian Computer-Human Interaction Conference on Augmentation, Application, Innovation, Collaboration - OzCHI '13*. ACM Press, Adelaide, Australia, 27–36. DOI:<http://dx.doi.org/10.1145/2541016.2541043>
- [261] Beatrice Rammstedt and Oliver P. John. 2005. Kurzversion des Big Five Inventory (BFI-K):. *Diagnostica* 51, 4 (oct 2005), 195–206. DOI:<http://dx.doi.org/10.1026/0012-1924.51.4.195>
- [262] Janet Read, Emanuela Mazzone, and Russell Beale. 2009. Under my Pillow – Designing Security for Children’s Special Things. In *Proceedings of the 23rd British HCI Group Annual Conference on People and Computers: Celebrating People and Technology*. BCS Learning & Development Ltd. Swindon, UK, Cambridge, UK, 288–292. <https://dl.acm.org/citation.cfm?id=1671046>
- [263] David Recordon and Drummond Reed. 2006. OpenID 2.0: A Platform for User-Centric Identity Management. In *Proceedings of the second ACM workshop on Digital identity management - DIM '06*. ACM Press, Alexandria, VA, USA, 11–15. DOI: <http://dx.doi.org/10.1145/1179529.1179532>
- [264] Robert W. Reeder, Iulia Ion, and Sunny Consolvo. 2017. 152 Simple Steps to Stay Safe Online: Security Advice for Non-Tech-Savvy Users. *IEEE Security and Privacy* 15, 5 (2017), 55–64. DOI:<http://dx.doi.org/10.1109/MSP.2017.3681050>
- [265] Karen Renaud and Antonella De Angeli. 2009. Visual Passwords: Cure-All or Snake Oil? *Commun. ACM* 52, 12 (2009), 135–140. DOI:<http://dx.doi.org/10.1145/1610252.1610287>
- [266] Karen Renaud and Verena Zimmermann. 2018. Nudging folks towards stronger password choices: providing certainty is the key. *Behavioural Public Policy* (feb 2018), 1–31. DOI:<http://dx.doi.org/10.1017/bpp.2018.3>
- [267] Karen Renaud, Verena Zimmermann, Joseph Maguire, and Steve Draper. 2017. Lessons Learned from Evaluating Eight Password Nudges in the Wild. In *Proceedings of The {LASER} Workshop: Learning from Authoritative Security Experiment Results ({LASER} 2017)*. USENIX Association, Arlington, VA, USA, 25–37.
- [268] Steve Riley. 2006a. It’s Me, and Here’s My Proof: Why Identity and Authentication Must Remain Distinct. (2006). <https://technet.microsoft.com/en-us/library/cc512578.aspx>



- 
- [269] Shannon Riley. 2006b. Password security: what users know and what they actually do. *Usability News* 8, 1 (2006), 2833–2836. <http://dl.acm.org/citation.cfm?id=1240866.1241089>
- [270] Luis Roalter, Stefan Diewald, Andreas Möller, Tobias Stockinger, and Matthias Kranz. 2013. User-Friendly Authentication and Authorization Using a Smartphone Proxy. In *Computer Aided Systems Theory - EUROCAST 2013*. Springer Berlin Heidelberg, Las Palmas de Gran Canaria, Spain, 390–399. DOI:[http://dx.doi.org/10.1007/978-3-642-53862-9\\_50](http://dx.doi.org/10.1007/978-3-642-53862-9_50)
- [271] Joel Ross, Lilly Irani, M Six Silberman, Andrew Zaldivar, and Bill Tomlinson. 2010. Who are the Crowdworkers ? Shifting Demographics in Mechanical Turk. *Chi 2010* JANUARY 2010 (2010), 2863–2872. DOI:<http://dx.doi.org/10.1145/1753846.1753873>
- [272] Scott Ruoti, Brent Roberts, and Kent Seamons. 2015. Authentication Melee: A Usability Analysis of Seven Web Authentication Systems. In *Proceedings of the 24th International Conference on World Wide Web - WWW '15*. ACM Press, Geneva, Switzerland, 916–926. DOI:<http://dx.doi.org/10.1145/2736277.2741683>
- [273] Richard M. Ryan and Edward L. Deci. 2000. Self-Determination Theory and the Facilitation of Intrinsic Motivation. *American Psychologist* 55, 1 (2000), 68–78. DOI: <http://dx.doi.org/10.1037/0003-066X.55.1.68>
- [274] Marlies Rybnicek, Christoph Lang-Muhr, and Daniel Haslinger. 2014. A roadmap to continuous biometric authentication on mobile devices. In *2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, Nicosia, Cyprus, 122–127. DOI:<http://dx.doi.org/10.1109/IWCMC.2014.6906343>
- [275] Martina Angela Sasse. 2015. Scaring and Bullying People into Security Won't Work. *Security & Privacy Economics* May/June (2015), 80–83.
- [276] Martina Angela Sasse, Sacha Brostoff, and Dirk Weirich. 2001. Transforming the "Weakest Link": A Human-Computer Interaction Approach for Usable and Effective Security. *BT Technology Journal* 19, 3 (2001), 122–131. DOI:<http://dx.doi.org/10.1023/A:1011902718709>
- [277] Martina Angela Sasse and Ivan Flechais. 2005. Usable Security: Why Do We Need It? How Do We Get It? In *Security and Usability: Designing secure systems that people can use*, Lorrie Faith Cranor and Simson L. Garfinkel (Eds.). O'Reilly Media, Inc., Sebastopol, CA, USA, Chapter 2, 13–30. <http://discovery.ucl.ac.uk/20345/>
- [278] M Angela Sasse, Matthew Smith, Cormac Herley, Heather Lipford, and Kami Vaniea. 2016. Debunking Security – Usability Tradeoff Myths. *IEEE Security and Privacy* 14, 5 (2016), 33–39.

- [279] Florian Schaub, Ruben Deyhle, and Michael Weber. 2012. Password entry usability and shoulder surfing susceptibility on different smartphone platforms. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia - MUM '12*. ACM Press, Ulm, Germany, 1. DOI:<http://dx.doi.org/10.1145/2406367.2406384>
- [280] Roland Schlöglhofer and Johannes Sametinger. 2012. Secure and usable authentication on mobile devices. In *Proceedings of the 10th International Conference on Advances in Mobile Computing & Multimedia - MoMM '12*. ACM Press, Bali, Indonesia, 257–263. DOI:<http://dx.doi.org/10.1145/2428955.2429004>
- [281] David Schmidt and Trent Jaeger. 2013. Pitfalls in the automated strengthening of passwords. In *Proceedings of the 29th Annual Computer Security Applications Conference on - ACSAC '13*. ACM, New Orleans, Louisiana, USA, 129–138. DOI:<http://dx.doi.org/10.1145/2523649.2523651>
- [282] Bruce Schneier. 2006. Real-World Passwords - Schneier on Security. (2006). <https://www.schneier.com/blog/archives/2006/12/realworld>
- [283] Bruce Schneier. 2013. The Psychology of Security. In *Proceedings of Progress in Cryptology – AFRICACRYPT 2008*, Serge Vaudenay (Ed.). Springer Berlin Heidelberg, Casablanca, Morocco, 50–79. DOI:[http://dx.doi.org/10.1007/978-3-540-68164-9\\_5](http://dx.doi.org/10.1007/978-3-540-68164-9_5)
- [284] Katharina Schwarz. 2016. *Partizipatives Design eines Registrierungsformulars mit non-verbalen persuasiven Elementen zur Passwortverbesserung*. Bachelor Thesis. Ludwig-Maximilians-Universität München.
- [285] Sean M. Segreti, William Melicher, Saranga Komanduri, Darya Melicher, Richard Shay, Blase Ur, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, and Michelle L. Mazurek. 2017. Diversify to Survive: Making Passwords Stronger with Adaptive Policies. In *Proceedings of the Thirteenth Symposium on Usable Privacy and Security (SOUPS 2017)*. USENIX Association, Santa Clara, CA, USA, 1–12. <https://www.usenix.org/system/files/conference/soups2017/soups2017-segreti.pdf>
- [286] Tobias Seitz. 2016. *The Decoy Effect for Passwords - A First Exploration*. Technical Report. Ludwig-Maximilians-Universität München, Munich, Germany. 8 pages. DOI:<http://dx.doi.org/10.13140/RG.2.1.2308.8880>
- [287] Tobias Seitz. 2017. Personalizing Password Policies and Strength Feedback. In *Personalizing Persuasive Technologies Workshop (Persuasive '17 adjunct)*. Springer Berlin Heidelberg, Amsterdam, The Netherlands.
- [288] Tobias Seitz, Manuel Hartmann, Jakob Pfab, and Samuel Souque. 2017a. Do Differences in Password Policies Prevent Password Reuse?. In *Proceedings of the*



---

2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems - CHI EA '17. ACM Press, New York, New York, USA, 2056–2063. DOI: <http://dx.doi.org/10.1145/3027063.3053100>

- [289] Tobias Seitz and Heinrich Hussmann. 2017. PASDJO: Quantifying Password Strength Perceptions with an Online Game. In *Proceedings of the 29th Australian Conference on Human-Computer Interaction (OzCHI 2017)*. ACM, Brisbane, Australia, 9. DOI: <http://dx.doi.org/10.1145/3152771.3152784>
- [290] Tobias Seitz, Florian Mathis, and Heinrich Hussmann. 2017b. The Bird is The Word: A Usability Evaluation of Emojis inside Text Passwords. In *Proceedings of the 29th Australian Conference on Computer-Human Interaction - OZCHI '17*. ACM Press, New York, New York, USA, 10–20. DOI: <http://dx.doi.org/10.1145/3152771.3152773>
- [291] Tobias Seitz, Emanuel von Zezschwitz, Stefanie Meitner, and Heinrich Hussmann. 2016. Influencing Self-Selected Passwords Through Suggestions and the Decoy Effect. In *Proceedings of the 1st European Workshop on Usable Security*. Internet Society, Darmstadt, 2:1–2:7. DOI: <http://dx.doi.org/10.14722/eurousec.2016.23002>
- [292] C. E. Shannon. 1951. Prediction and Entropy of Printed English. *Bell System Technical Journal* 30, 1 (1951), 50–64. DOI: <http://dx.doi.org/10.1002/j.1538-7305.1951.tb01366.x>
- [293] Richard Shay. 2015. *Creating Usable Policies for Stronger Passwords with MTurk*. Dissertation. Carnegie Mellon University.
- [294] Richard Shay and Elisa Bertino. 2009. A Comprehensive Simulation Tool for the Analysis of Password Policies. *International Journal of Information Security* 8, 4 (2009), 275–289. DOI: <http://dx.doi.org/10.1007/s10207-009-0084-3>
- [295] Richard Shay, Adam L Durity, Sean M. Segreti, Blase Ur, Lujo Bauer, and Nicolas Christin. 2016. Designing Password Policies for Strength and Usability. *ACM Transactions on Information and System Security* 18, 4 (2016), 13:1–13:34. DOI: <http://dx.doi.org/10.1145/2891411>
- [296] Richard Shay, Iulia Ion, Robert W Reeder, and Sunny Consolvo. 2014. "My religious aunt asked why i was trying to sell her viagra": Experiences with account hijacking. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*. ACM Press, Toronto, ON, Canada, 2657–2666. DOI: <http://dx.doi.org/10.1145/2556288.2557330>
- [297] Richard Shay, Patrick Gage Kelley, Saranga Komanduri, Michelle L. Mazurek, Blase Ur, Timothy Vidas, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2012. Correct Horse Battery Staple. In *Proceedings of the Eighth Symposium on Usable Privacy and Security (SOUPS '12)*. ACM, New York, NY, USA, 1–20. DOI: <http://dx.doi.org/10.1145/2335356.2335366>

- [298] Richard Shay, Patrick Gage Kelley, Pedro Giovanni Leon, Michelle L. Mazurek, Nicolas Christin, and Lorrie Faith Cranor. 2010. Encountering Stronger Password Requirements : User Attitudes and Behaviors. In *Proceedings of the Sixth Symposium on Usable Privacy and Security (SOUPS '10)*. ACM, Redmond, WA, USA, Article 2, 20 pages. DOI:<http://dx.doi.org/10.1145/1837110.1837113>
- [299] Richard Shay, Saranga Komanduri, Adam L Durity, Phillip Seyoung Huh, Michelle L. Mazurek, Sean M Segreti, Blase Ur, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2014. Can Long Passwords Be Secure and Usable?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '14)*. DOI:<http://dx.doi.org/10.1145/2556288.2557377>
- [300] Richard Shay, Blase Ur, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Alain Forget, Saranga Komanduri, Michelle L. Mazurek, William Melicher, and Sean M Segreti. 2015. A Spoonful of Sugar? The Impact of Guidance and Feedback on Password-Creation Behavior. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '15)*. ACM, New York, NY, USA, 2903–2912. DOI:<http://dx.doi.org/10.1145/2702123.2702586>
- [301] Steve Sheng, Mandy Holbrook, Ponnurangam Kumaraguru, Lorrie Faith Cranor, and Julie Downs. 2010. Who Falls for Phish? A Demographic Analysis of Phishing Susceptibility and Effectiveness of Interventions. *ACM Transactions on Internet Technology (TOIT)* 10, 2 (2010), 373 – 382. DOI:<http://dx.doi.org/10.1145/1753326.1753383>
- [302] Michael Sherman, Gradeigh Clark, Yulong Yang, Shridatt Sugrim, Arttu Modig, Janne Lindqvist, Antti Oulasvirta, and Teemu Roos. 2014. User-generated free-form gestures for authentication. In *Proceedings of the 12th annual international conference on Mobile systems, applications, and services - MobiSys '14*. ACM Press, Bretton Woods, NH, USA, 176–189. DOI:<http://dx.doi.org/10.1145/2594368.2594375>
- [303] M. Shevlin and J.N.V. Miles. 1998. Effects of sample size, model specification and factor loadings on the GFI in confirmatory factor analysis. *Personality and Individual Differences* 25 (1998), 85–90. DOI:[http://dx.doi.org/10.1016/S0191-8869\(98\)00055-5](http://dx.doi.org/10.1016/S0191-8869(98)00055-5)
- [304] Jordan Shropshire, Merrill Warkentin, Allen C. Johnston, and Mark B. Schmidt. 2006. Personality and IT security: An application of the five-factor model. *Americas Conference on Information Systems (AMCIS)* January (2006), 3443–3449.
- [305] Jordan Shropshire, Merrill Warkentin, and Shwadhin Sharma. 2015. Personality, attitudes, and intentions: Predicting initial adoption of information security behavior. *Computers & Security* 49 (2015), 177–191. DOI:<http://dx.doi.org/10.1016/j.cose.2015.01.002>

- 
- [306] Magdalena Siferlinger. 2017. *Unterstützung bei der strategischen Wiederverwendung von Passwörtern*. Bachelor Thesis. Ludwig-Maximilians-Universität München.
- [307] Itamar Simonson. 1989. Choice Based on Reasons: The Case of Attraction and Compromise Effects. *Journal of Consumer Research* 16, 2 (1989), 158. DOI: <http://dx.doi.org/10.1086/209205>
- [308] Supriya Singh, Anuja Cabraal, Catherine Demosthenus, Gunela Astbrink, and Michele Furlong. 2007. Password Sharing: Implications for Security Design Based on Social Practice. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, San Jose, CA, USA, 895–904. DOI: <http://dx.doi.org/10.1145/1978942.1979324>
- [309] Andreas Sotirakopoulos, Kirstie Hawkey, and Konstantin Beznosov. 2011. On the challenges in usable security lab studies. In *Proceedings of the Seventh Symposium on Usable Privacy and Security - SOUPS '11*. ACM Press, Pittsburgh, PA, USA, 1. DOI: <http://dx.doi.org/10.1145/2078827.2078831>
- [310] Sanjay Srivastava, Oliver P. John, Samuel D. Gosling, and Jeff Potter. 2003. Development of personality in early and middle adulthood: Set like plaster or persistent change? *Journal of Personality and Social Psychology* 84, 5 (2003), 1041–1053. DOI: <http://dx.doi.org/10.1037/0022-3514.84.5.1041>
- [311] Clemens Stachl, Sven Hilbert, Jiew-Quay Au, Daniel Buschek, Alexander De Luca, Bernd Bischl, Heinrich Hussmann, and Markus Bühner. 2017. Personality Traits Predict Smartphone Usage. *European Journal of Personality* 31, 6 (nov 2017), 701–722. DOI: <http://dx.doi.org/10.1002/per.2113>
- [312] Frank Stajano and Paul Wilson. 2011. Understanding Scam Victims: Seven Principles for Systems Security. *Commun. ACM* 54, 3 (2011), 70. DOI: <http://dx.doi.org/10.1145/1897852.1897872>
- [313] Michelle Steves, Mary Theofanos, Celia Paulsen, and Athos Ribeiro. 2015. Password Policy Languages: Usable Translation from the Informal to the Formal. In *Proceedings of the International Conference on Human Aspects of Information Security, Privacy, and Trust (HAS '15)*. Springer International Publishing, Los Angeles, CA, USA, 119–130. DOI: [http://dx.doi.org/10.1007/978-3-319-20376-8\\_11](http://dx.doi.org/10.1007/978-3-319-20376-8_11)
- [314] Elizabeth Stobert and Robert Biddle. 2013. Memory retrieval and graphical passwords. In *Proceedings of the Ninth Symposium on Usable Privacy and Security - SOUPS '13*. ACM, Newcastle, UK, 1. DOI: <http://dx.doi.org/10.1145/2501604.2501619>
- [315] Elizabeth Stobert and Robert Biddle. 2014a. A Password Manager that Doesn't Remember Passwords. In *Proceedings of the 2014 workshop on New Security Paradigms Workshop*. ACM, Victoria, BC, Canada, 39–52. DOI: <http://dx.doi.org/10.1145/2683467.2683471>

- [316] Elizabeth Stobert and Robert Biddle. 2014b. The Password Life Cycle: User Behaviour in Managing Passwords. In *Proceedings of the 10th Symposium On Usable Privacy and Security (SOUPS '14)*. USENIX Association, Menlo Park, CA, USA, 243–255.
- [317] Elizabeth Stobert and Robert Biddle. 2015. Expert Password Management. In *Proceedings of Passwords 2015*. Springer International Publishing, 3–20. <https://passwordscon.org/wp-content/uploads/2015/05/preproceedings.pdf>
- [318] Elizabeth Ann Stobert. 2015. *Graphical Passwords and Practical Password Management*. Doctoral Thesis. Carlton University.
- [319] Tobias Stockinger. 2011. *Implicit authentication for mobile devices*. Technical Report. Media Informatics Group, Munich, Germany.
- [320] Tobias Stockinger, Marion Koelle, Patrick Lindemann, Matthias Kranz, and Luis Roalter. 2015. Towards Leveraging Behavioral Economics in Mobile Application Design. In *Gamification in Education and Business*, Torsten Reiners and Lincoln Woods (Eds.). Springer International Publishing, 105–131. DOI:[http://dx.doi.org/10.1007/978-3-319-10208-5\\_6](http://dx.doi.org/10.1007/978-3-319-10208-5_6)
- [321] Anselm Strauss and Juliet M. Corbin. 1990. *Basics of qualitative research: grounded theory procedure and techniques*. Vol. 13. SAGE Publications. 3–21 pages.
- [322] San-tsai Sun, Yazan Boshmaf, Kirstie Hawkey, and Konstantin Beznosov. 2010. A Billion Keys, but Few Locks: The Crisis of Web Single Sign-On. In *Proceedings of the 2010 workshop on New security paradigms - NSPW '10*. ACM, Concord, MA, USA, 61–72. DOI:<http://dx.doi.org/10.1145/1900546.1900556>
- [323] San-tsai Sun, Eric Pospisil, Ildar Muslukhov, Nuray Dindar, Kirstie Hawkey, and Konstantin Beznosov. 2011. What makes users refuse web single sign-on? An Empirical Investigation of OpenID. In *Proceedings of the Seventh Symposium on Usable Privacy and Security - SOUPS '11*. ACM Press, Pittsburgh, PA, USA, 1–20. DOI:<http://dx.doi.org/10.1145/2078827.2078833>
- [324] Joshua Sunshine, Serge Egelman, Hazim Almuhiemedi, Neha Atri, and Lorrie Faith Cranor. 2009. Crying Wolf: An Empirical Study of SSL Warning Effectiveness. In *18th USENIX Security Symposium*. 399–432. DOI:[http://dx.doi.org/10.1016/S1353-4858\(01\)00916-3](http://dx.doi.org/10.1016/S1353-4858(01)00916-3)
- [325] Caroline Taggart. 2015. *New Words for Old: Recycling Our Language for the Modern World*. Michael O'Mara Books. 192 pages. <https://books.google.de/books?id=VP04CgAAQBAJ>
- [326] Michael Cheng Tek Tai. 2012. Deception and informed consent in social, behavioral, and educational research (SBER). *Tzu Chi Medical Journal* 24, 4 (2012), 218–222. DOI:<http://dx.doi.org/10.1016/j.tcmj.2012.05.003>

- 
- [327] Mohammad Tamviruzzaman, Sheikh Iqbal Ahamed, Chowdhury Sharif Hasan, and Casey O'Brien. 2009. ePet: When Cellular Phone Learns to Recognize Its Owner. In *Proceedings of the 2nd ACM workshop on Assurable and usable security configuration - SafeConfig '09*. ACM Press, Chicago, IL, USA, 13–17. DOI:<http://dx.doi.org/10.1145/1655062.1655066>
- [328] Andrew S. Tanenbaum and D. (David) Wetherall. 2011. *Computer networks* (5th editio ed.). Pearson Prentice Hall. 933 pages. <https://books.google.de/books?id=2xWHAQAACAAJ>
- [329] Furkan Tari, A. Ant Ozok, and Stephen H. Holden. 2006. A Comparison of Perceived and Real Shoulder-surfing Risks between Alphanumeric and Graphical Passwords. In *Proceedings of the second symposium on Usable privacy and security - SOUPS '06*. ACM Press, Pittsburgh, PA, USA, 56–66. DOI:<http://dx.doi.org/10.1145/1143120.1143128>
- [330] RH Thaler. 2004. Mental accounting matters. Vol. 206. Princeton University Press, Chapter 3, 183–206. <http://books.google.com/books?hl=en>
- [331] Richard H. Thaler. 1999. Mental Accounting Matters. *Journal of Behavioral Decision Making* 12, 3 (sep 1999), 183–206. DOI:[http://dx.doi.org/10.1002/\(SICI\)1099-0771\(199909\)12:3<183::AID-BDM318>3.0.CO;2-F](http://dx.doi.org/10.1002/(SICI)1099-0771(199909)12:3<183::AID-BDM318>3.0.CO;2-F)
- [332] Richard H. Thaler and Cass R. Sunstein. 2008. *Nudge: Improving Decisions about Health, Wealth, and Happiness*. Yale University Press. <https://books.google.com/books?hl=de>
- [333] Richard H Thaler, Cass R Sunstein, and John P Balz. 2010. Choice Architecture. *Social Science Research Network* April (2010). DOI:<http://dx.doi.org/10.2139/ssrn.1583509>
- [334] Julie Thorpe, Muath Al-Badawi, Brent MacRae, and Amirali Salehi-Abari. 2014. The presentation effect on graphical passwords. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*. ACM Press, Toronto, ON, Canada, 2947–2950. DOI:<http://dx.doi.org/10.1145/2556288.2557212>
- [335] Garreth W Tigwell and David R. Flatla. 2016. “Oh that’ s what you meant!”: Reducing Emoji Misunderstanding. In *Proceedings of the 18th International Conference on Human-Computer Interaction with Mobile Devices and Services Adjunct (MobileHCI '16)*. ACM, Copenhagen, Denmark, 859–866. DOI:<http://dx.doi.org/2957265.2961844>
- [336] Noam Tractinsky, Adi Katz, and D. Ikar. 2000. What is beautiful is usable. *Interacting with Computers* 13, 2 (2000), 127–145. DOI:[http://dx.doi.org/10.1016/S0953-5438\(00\)00031-X](http://dx.doi.org/10.1016/S0953-5438(00)00031-X)



- [337] Harshal Tupsamudre, Vijayanand Banahatti, and Sachin Lodha. 2016. POSTER : Improved Markov Strength Meters for Passwords. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*. ACM, Vienna, Austria. DOI:<http://dx.doi.org/10.1145/2976749.2989058>
- [338] Amos Tversky and Daniel Kahneman. 1974. Judgment under Uncertainty: Heuristics and Biases. *Science* 185, 4157 (sep 1974), 1124–31. DOI:<http://dx.doi.org/10.1126/science.185.4157.1124>
- [339] Sven Uebelacker and Susanne Quiel. 2014. The Social Engineering Personality Framework. In *2014 Workshop on Socio-Technical Aspects in Security and Trust*. IEEE, Vienna, Austria, 24–30. DOI:<http://dx.doi.org/10.1109/STAST.2014.12>
- [340] Sebastian Uellenbeck, Markus Dürmuth, Christopher Wolf, and Thorsten Holz. 2013. Quantifying the security of graphical passwords: the case of android unlock patterns. *Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security - CCS '13* 44, 4 (2013), 161–172. DOI:<http://dx.doi.org/10.1145/2508859.2516700>
- [341] Blase Ur, Felicia Alfieri, Maung Aung, Lujo Bauer, Nicolas Christin, Jessica Colnago, Lorrie Faith Cranor, Henry Dixon, Pardis Emami Naeini, Hana Habib, Noah Johnson, and William Melicher. 2017. Design and Evaluation of a Data-Driven Password Meter. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (CHI '17)*. ACM, Denver, CO, USA, 3775–3786. DOI:<http://dx.doi.org/10.1145/3025453.3026050>
- [342] Blase Ur, Jonathan Bees, Sean M. Segreti, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2016. Do Users' Perceptions of Password Security Match Reality ?. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, San Jose, CA, USA, 3748–3760. DOI:<http://dx.doi.org/10.1145/2858036.2858546>
- [343] Blase Ur, Patrick Gage Kelley, Saranga Komanduri, Joel Lee, Michael Maass, Michelle L. Mazurek, Timothy Passaro, Richard Shay, Timothy Vidas, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2012a. How Does Your Password Measure Up? The Effect of Strength Meters on Password Creation. In *Proceedings of the 21st USENIX Security symposium*. USENIX Association, Bellevue, WA, USA, 5–16. <https://www.usenix.org/system/files/conference/usenixsecurity12/sec12-final209.pdf>
- [344] Blase Ur, Patrick Gage Kelley, Saranga Komanduri, Joel Lee, Michael Maass, Michelle L. Mazurek, Timothy Passaro, Richard Shay, Timothy Vidas, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Serge Egelman, and Julio López. 2012b. Helping Users Create Better Passwords. *login* 37, 6 (2012), 51–57. <http://scholar.google.co.uk/scholar?q=the+science+of+guessing+bonneau>

- 
- [345] Blase Ur, Fumiko Noma, Jonathan Bees, Sean M. Segreti, Richard Shay, Lujo Bauer, Nicolas Christin, and Lorrie Faith Cranor. 2015a. "I Added '!' at the End to Make It Secure": Observing Password Creation in the Lab. In *Proceedings of the Symposium on Usable Privacy and Security (SOUPS '15)*. USENIX Association, Ottawa, Canada, 123–140.
- [346] Blase Ur, Sean M. Segreti, Lujo Bauer, Nicolas Christin, Lorrie Faith Cranor, Saranga Komanduri, Darya Kurilova, Michelle L Mazurek, William Melicher, and Richard Shay. 2015b. Measuring Real-World Accuracies and Biases in Modeling Password Guessability. In *24th USENIX Security Symposium (USENIX Security 15)*. USENIX Association, Washington, DC, USA, 463—481. <https://www.usenix.org/conference/usenixsecurity15/technical-sessions/presentation/ur>
- [347] Anthony Vance, David Eargle, Kirk Ouimet, and Detmar Straub. 2013. Enhancing password security through interactive fear appeals: A web-based field experiment. In *Proceedings of the Annual Hawaii International Conference on System Sciences (HICSS'13)*. IEEE, Koloa, HI, USA, 2988–2997. DOI:<http://dx.doi.org/10.1109/HICSS.2013.196>
- [348] Rafael Veras, Christopher Collins, and Julie Thorpe. 2014. On the Semantic Patterns of Passwords and their Security Impact. In *Proceedings 2014 Network and Distributed System Security Symposium*. Internet Society, Reston, VA, USA, 23–26. DOI:<http://dx.doi.org/10.14722/ndss.2014.23103>
- [349] Rafael Veras, Julie Thorpe, and Christopher Collins. 2012. Visualizing semantics in passwords. In *Proceedings of the Ninth International Symposium on Visualization for Cyber Security - VizSec '12*. ACM Press, Seattle, WA, USA, 88–95. DOI:<http://dx.doi.org/10.1145/2379690.2379702>
- [350] Vilhelm Verendel. 2008. *A Prospect Theory approach to Security*. Technical Report 08. Göteborg University, Göteborg.
- [351] George E Violettas and Kyriakos Papadopoulos. 2014. Passwords to absolutely avoid (A Survey in Greece). In *The Fifth International Conference on the Applications of Digital Information and Web Technologies (ICADIWT 2014)*. IEEE, Bangalore, India, 60–68. DOI:<http://dx.doi.org/10.1109/ICADIWT.2014.6814693>
- [352] Melanie Volkamer and Karen Renaud. 2013. Mental Models – General Introduction and Review of Their Application to Human-Centred Security. Vol. 8260. Springer Berlin Heidelberg, 255–280. DOI:[http://dx.doi.org/10.1007/978-3-642-42001-6\\_18](http://dx.doi.org/10.1007/978-3-642-42001-6_18)
- [353] Emanuel Von Zezschwitz. 2016. *Risks and Potentials of Graphical and Gesture-based Authentication for Touchscreen Mobile Devices*. PhD Thesis. Ludwig-Maximilians Universität München.



- [354] Emanuel Von Zezschwitz, Alexander De Luca, and Heinrich Hussmann. 2013. Survival of the shortest: A retrospective analysis of influencing factors on password composition. In *Human-Computer Interaction – INTERACT 2013, Lecture Notes in Computer Science*, Paula Kotzé, Gary Marsden, Gitte Lindgaard, Janet Wesson, and Marco Winckler (Eds.). Vol. 8119. Springer Berlin Heidelberg, 460–467. DOI: [http://dx.doi.org/10.1007/978-3-642-40477-1\\_28](http://dx.doi.org/10.1007/978-3-642-40477-1_28)
- [355] Emanuel von Zezschwitz, Alexander De Luca, and Heinrich Hussmann. 2014. Honey, I Shrunk the Keys: Influences of Mobile Devices on Password Composition and Authentication Performance. In *Proceedings of the 8th Nordic Conference on Human-Computer Interaction Fun, Fast, Foundational - NordiCHI '14*. ACM Press, Helsinki, Finland, 461–470. DOI: <http://dx.doi.org/10.1145/2639189.2639218>
- [356] Emanuel von Zezschwitz, Alexander De Luca, Philipp Janssen, and Heinrich Hussmann. 2015. Easy to Draw, but Hard to Trace? On the Observability of Grid-based (Un)lock Patterns. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, Seoul, South Korea, 2339–2342. DOI: <http://dx.doi.org/10.1145/2702123.2702202>
- [357] Emanuel von Zezschwitz, Malin Eiband, Daniel Buschek, Sascha Oberhuber, Alexander De Luca, Florian Alt, and Heinrich Hussmann. 2016. On quantifying the effective password space of grid-based unlock gestures. In *Proceedings of the 15th International Conference on Mobile and Ubiquitous Multimedia - MUM '16*. ACM Press, Rovaniemi, Finland, 201–212. DOI: <http://dx.doi.org/10.1145/3012709.3012729>
- [358] Ding Wang. 2016. Targeted Online Password Guessing: An Underestimated Threat. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security (CCS '16)*. ACM, Vienna, Austria, 1242–1254. DOI: <http://dx.doi.org/10.1145/2976749.2978339>
- [359] Ding Wang, Haibo Cheng, and Ping Wang. 2015. *Understanding Passwords of Chinese Users : Characteristics , Security and Implications*. Technical Report.
- [360] Ding Wang, Debiao He, Haibo Cheng, and Ping Wang. 2016. fuzzyPSM: A New Password Strength Meter Using Fuzzy Probabilistic Context-Free Grammars. In *Proceedings of the International Conference on Dependable Systems and Networks (DSN)*. IEEE, 595–606. DOI: <http://dx.doi.org/10.1109/DSN.2016.60>
- [361] Ding Wang and Ping Wang. 2015. The Emperor's New Password Creation Policies. In *Proceedings of the 20th European Symposium on research in Computer Security - ESORICS'15*. Springer, Vienna, Austria, 456–477. DOI: <http://dx.doi.org/10.1007/978-3-319-24177-7>
- [362] Rick Wash. 2010. Folk models of home computer security. In *Proceedings of the Sixth Symposium on Usable Privacy and Security - SOUPS '10*. ACM Press, Redmond, WA, USA, 1. DOI: <http://dx.doi.org/10.1145/1837110.1837125>


- 
- [363] Rick Wash, Emilee Rader, Ruthie Berman, Macalester College, Rick Wash, Emilee Rader, and Ruthie Berman. 2016. Understanding Password Choices: How Frequently Entered Passwords are Re-used Across Websites. In *Symposium on Usable Privacy and Security - SOUPS'16*. USENIX Association, Denver, CO, USA, 175–188.
- [364] Rick Wash, Emilee Rader, and Chris Fennell. 2017. Can People Self-Report Security Accurately? Agreement Between Self-Report and Behavioral Measures. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems - CHI '17*. ACM Press, Denver, CO, USA, 2228–2232. DOI:<http://dx.doi.org/10.1145/3025453.3025911>
- [365] Matt Weir, Sudhir Aggarwal, Michael Collins, and Henry Stern. 2010. Testing metrics for password creation policies by attacking large sets of revealed passwords. *Proceedings of the 17th ACM conference on Computer and communications security - CCS '10* (2010), 162. DOI:<http://dx.doi.org/10.1145/1866307.1866327>
- [366] Matt Weir, Sudhir Aggarwal, Breno de Medeiros, and Bill Glodek. 2009. Password Cracking Using Probabilistic Context-Free Grammars. In *2009 30th IEEE Symposium on Security and Privacy*. IEEE, Oakland, CA, USA, 391–405. DOI:<http://dx.doi.org/10.1109/SP.2009.8>
- [367] Dirk Weirich and Martina Angela Sasse. 2001a. Persuasive Password Security. In *CHI '01 extended abstracts on Human factors in computing systems - CHI '01*. ACM Press, Minneapolis, Minnesota, USA, 139–140. DOI:<http://dx.doi.org/10.1145/634067.634152>
- [368] Dirk Weirich and Martina Angela Sasse. 2001b. Pretty Good Persuasion: A First Step towards Effective Password Security in the Real World. In *Proceedings of the 2001 Workshop on New Security Paradigms (NSPW '01)*. ACM, New York, NY, USA, 137–143. DOI:<http://dx.doi.org/10.1145/634149.634152>
- [369] Daniel Lowe Wheeler. 2016. zxcvbn: Low-Budget Password Strength Estimation. In *Proceedings of the 25th USENIX Security Symposium (USENIX Security 16)*. USENIX Association, Austin, TX, 157–173. <https://www.usenix.org/conference/usenixsecurity16/technical-sessions/presentation/wheeler>
- [370] Andrew M White, Katherine Shaw, Fabian Monrose, and Elliott Moreton. 2014. Isn't that Fantabulous: Security, Linguistic and Usability Challenges of Pronounceable Tokens. In *Proceedings of the 2014 workshop on New Security Paradigms Workshop - NSPW '14*. ACM Press, Victoria, British Columbia, Canada, 25–38. DOI:<http://dx.doi.org/10.1145/2683467.2683470>
- [371] Alma Whitten and J. D. Tygar. 1999. Why Johnny Can't Encrypt: A Usability Evaluation of PGP 5.0. In *Proceedings of the 8th USENIX Security Symposium*. USENIX Association, Washington, DC, USA, 169–184. DOI:<http://dx.doi.org/169-184>

- [372] Susan Wiedenbeck, Jim Waters, Jean Camille Birget, Alex Brodskiy, and Nasir Memon. 2005. PassPoints: Design and longitudinal evaluation of a graphical password system. *International Journal of Human Computer Studies* 63, 1-2 (2005), 102–127. DOI:<http://dx.doi.org/10.1016/j.ijhcs.2005.04.010>
- [373] Susan Wiedenbeck, Jim Waters, Leonardo Sobrado, and Jean-Camille Birget. 2006. Design and evaluation of a shoulder-surfing resistant graphical password scheme. In *Proceedings of the working conference on Advanced visual interfaces - AVI '06*. ACM Press, Venezia, Italiy, 177. DOI:<http://dx.doi.org/10.1145/1133265.1133303>
- [374] Craig Wiggington, Mike Curran, and Terrence Karner. 2017. *2017 Global Mobile Consumer Survey: US Edition*. Technical Report. Deloitte. 1–29 pages. <http://www2.deloitte.com/be/en.html>
- [375] Sanjaya Wijeratne, Lakshika Balasuriya, Amit Sheth, and Derek Doran. 2010. EmojNet : An Open Service and API for Emoji Sense Discovery. (2010).
- [376] Daricia Wilkinson, Saadhika Sivakumar, David Cherry, Bart P Knijnenburg, Elaine M Raybourn, Pamela Wisniewski, and Henry Sloan. 2017. ( Work in Progress ) User-Tailored Privacy by Design. In *Proceedings of USEC'17*. Internet Society, 1–12.
- [377] Naomi Woods and Mikko Siponen. 2018. Too many passwords? How understanding our memory can increase password memorability. *International Journal of Human-Computer Studies* 111, Supplement C (2018), 36–48. DOI:<http://dx.doi.org/https://doi.org/10.1016/j.ijhcs.2017.11.002>
- [378] Min Wu, Robert C. Miller, and Simson L. Garfinkel. 2006. Do Security Toolbars Actually Prevent Phishing Attacks?. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '06)*. ACM, Montréal, Québec, Canada, 601–610. DOI:<http://dx.doi.org/10.1145/1124772.1124863>
- [379] Heng Xu, Mb Rosson, and Jm Carroll. 2007. Increasing the Persuasiveness of IT Security Communication: Effects of Fear Appeals and Self-View. In *Workshop on Usable IT Security Management, Symposium on Usable Privacy and Security (SOUPS)*. Carnegie Mellon University, Pittsburgh, PA, USA, 1–4. <http://cups.cs.cmu.edu/soups/2007/workshop/IT>
- [380] Jeff Yan, Blackwell Alan, Ross Anderson, and Alasdair Grant. 2004. Password Memorability and Security: Empirical Results. *IEEE Security and Privacy* 2, 5 (2004), 25–31. DOI:<http://dx.doi.org/10.1109/MSP.2004.81>
- [381] Weining Yang, Ninghui Li, Omar Chowdhury, Aiping Xiong, and Robert W Proctor. 2016. An Empirical Study of Mnemonic Sentence-based Password Generation Strategies. In *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security - CCS'16*. ACM Press, New York, New York, USA, 1216–1229. DOI:<http://dx.doi.org/10.1145/2976749.2978346>

- 
- [382] Yulong Yang, Janne Lindqvist, and Antti Oulasvirta. 2014. Text Entry Method Affects Password Security. In *Proceedings of the Learning from Authoritative Security Experiment Results Workshop (LASER '14)*. USENIX Association, Arlington, VA, USA, 11–20. <https://www.usenix.org/system/files/conference/laser2014/laser-2014-paper-yang.pdf>
- [383] Zishuang (Eileen) Ye, Sean Smith, and Denise Anthony. 2005. Trusted Paths for Browsers. *ACM Transactions on Information and System Security* 8, 2 (2005), 153–186. DOI:<http://dx.doi.org/10.1145/1065545.1065546>
- [384] Ka-Ping Yee and Kragen Sitaker. 2006. Passpet: Convenient Password Management and Phishing Protection. In *Proceedings of the second symposium on Usable privacy and security - SOUPS '06*. ACM Press, Pittsburgh, PA, USA, 32–43. DOI:<http://dx.doi.org/10.1145/1143120.1143126>
- [385] Indi Young. 2008. *Mental Models: Aligning Design Strategy with Human Behavior*. 299 pages. <http://books.google.com/books?id=b5aLQ>
- [386] Wu Youyou, Michal Kosinski, and David Stillwell. 2015. Computer-based personality judgments are more accurate than those made by humans. *Proceedings of the National Academy of Sciences* 112, 4 (jan 2015), 1036–1040. DOI:<http://dx.doi.org/10.1073/pnas.1418680112>
- [387] Nur Haryani Zakaria and Norliza Katuk. 2013. Towards designing effective security messages: Persuasive password guidelines. In *Proceedings of the International Conference on Research and Innovation in Information Systems (ICRIIS)*. IEEE, Kajang, Malaysia, 129–134. DOI:<http://dx.doi.org/10.1109/ICRIIS.2013.6716697>
- [388] Yinqian Zhang, Fabian Monrose, and Michael K Reiter. 2010. The Security of Modern Password Expiration: An Algorithmic Framework and Empirical Analysis. In *Proceedings of the 17th ACM conference on Computer and communications security - CCS '10*. ACM Press, Chicago, IL, USA, 176. DOI:<http://dx.doi.org/10.1145/1866307.1866328>
- [389] Leah Zhang-Kennedy, Sonia Chiasson, and Paul van Oorschot. 2016. Revisiting password rules: facilitating human management of passwords. In *Proceedings of the Symposium on Electronic Crime Research (eCrime)*. IEEE, Toronto, ON, Canada, 1–10. DOI:<http://dx.doi.org/10.1109/ECRIME.2016.7487945>

# Appendix

## Policy Audit

TLD	min	max	L	U	D	S	<>	C	
google.de	8	100	✓	✓	✓	(✗)	✗	basic	✓
youtube.com	8	100	✓	✓	✓	(✗)	✗	basic	✓
facebook.com	6	0	✓	✓	✓	✓	✓	basic	✓
amazon.de	6	0	✓	✓	✓	✓	✗	basic	✗
google.com	8	100	✓	✓	✓	(✗)	✗	basic	
ebay.de	6	64	✓	✓	✓	(✗)	✗	2class	✓
bing.com	8	0	✓	✓	✓	✓	✗	2class	✓
wikipedia.org	1	0	✓	✓	✓	✓	✓	basic	✗
web.de	8	40	✓	✓	✓	(✗)	✗	basic	✗
gmx.net	8	40	✓	✓	✓	(✗)	(✗)	basic	✗
t-online.de	8	16	✓	✓	✓	(✗)	✗	2class	✗
ebay-kleinanzeigen.de	6	0	✓	✓	✓	✓	✓	basic	
yahoo.com	9	32	✓	✓	✓	✓	✗	basic	✓
msn.com	8	0	✓	✓	✓	(✗)	✗	2class	✓
bild.de	6	0	✓	✓	✓	✓	✓	basic	✗
spiegel.de	8	50	✓	!	✓	(✗)	(✗)	3class	✗
live.com	8	0	✓	✓	✓	(✗)	✗	2class	✓
paypal.com	8	20	✓	✓	✓	✓	✓	paypal	✗
mobile.de	6	0	✓	✓	✓	(✗)	✗	2class	✗
xhamster.com	4	0	✓	✓	✓	✓	✓	basic	✓
chip.de	1	0	✓	✓	✓	✓	✓	basic	✗
twitter.com	6	0	✓	✓	✓	✓	✓	basic	
otto.de	6	12	✓	✓	✓	✓	✗	basic	✗
gutefrage.net	8	0	✓	✓	✓	✓	✓	basic	✗
immobilienscout24.de	5	40	✓	✓	✓	✓	✓	basic	✗
streamcloud.eu	4	32	✓	✓	✓	✓	✓	basic	✗
bs.to	5	0	✓	✓	✓	✓	✓	basic	✗
instagram.com	6	0	✓	✓	✓	✓	(✗)	basic	✓
outbrain.com	8	144	!	!	!	✓	(✗)	outbrain	✗
focus.de	8	0	✓	✓	✓	✓	✓	basic	✓
pornhub.com	6	40	✓	✓	✓	(✗)	✗	basic	✓
bahn.de	6	40	✓	✓	✓	✓	✓	bahn	✗
bongacams.com	6	32	✓	✓	✓	(✗)	(✗)	basic	✓
microsoft.com	8	0	✓	✓	✓	(✗)	✗	2class	✓
xing.com	4	0	✓	✓	✓	✓	✓	basic	✗
netflix.com	4	50	✓	✓	✓	(✗)	(✗)	basic	✗
1und1.de	8	0	✓	✓	✓	(✗)	✗	basic	✗
blogspot.de	8	100	✓	✓	✓	(✗)	✗	basic	✓
pinterest.com	6	0	✓	✓	✓	✓	✗	basic	✓
autoscout24.de	6	80	✓	✓	✓	✓	✓	basic	✗
idealo.de	8	0	✓	✓	✓	✓	✓	idealo	✗



TLD	min	max	L	U	D	S	<>	C	
chefkoch.de	4	32	✓	✓	✓	(X)	(X)	basic	X
wetter.com	8	0	✓	✓	✓	✓	✓	basic	X
kicker.de	6	20	✓	✓	✓	✓	(X)	basic	X
twitch.tv	8	40	✓	✓	✓	✓	✓	basic	X
tumblr.com	8	0	✓	✓	✓	✓	✓	basic	✓
booking.com	8	100	✓	✓	✓	✓	✓	basic	X
welt.de	6	0	✓	✓	✓	✓	✓	basic	X
heise.de	1	0	✓	✓	✓	✓	✓	basic	X
zalando.de	6	0	✓	✓	✓	✓	✓	basic	X
txxx.com	5	0	✓	✓	✓	✓	✓	basic	X
wordpress.com	6	0	✓	✓	✓	✓	✓	basic	✓
linkedin.com	6	0	✓	✓	✓	✓	✓	basic	✓
youporn.com	4	20	✓	✓	✓	✓	✓	basic	X
sueddeutsche.de	6	0	✓	✓	✓	✓	✓	basic	X
leo.org	1	0	✓	✓	✓	✓	✓	basic	X
reddit.com	6	0	✓	✓	✓	✓	✓	basic	X
deutsche-bank.de	8	16	✓	✓	✓	✓	✓	2class	X
arbeitsagentur.de	8	20	!	!	!	✓	✓	3class	X
amazon.com	6	0	✓	✓	✓	✓	✓	basic	X
zeit.de	6	0	✓	✓	✓	(X)	(X)	basic	X
wetteronline.de	6	12	✓	✓	✓	✓	✓	basic	X
ikea.com	7	10	!	✓	!	✓	✓	2class	X
computerbild.de	8	0	✓	✓	✓	✓	✓	basic	X
tvnow.de	6	18	✓	✓	✓	✓	✓	basic	X
hclips.com	5	0	✓	✓	✓	✓	✓	basic	X
apple.com	8	32	!	!	!	✓	✓	3class	X
sport1.de	7	12	✓	✓	✓	✓	✓	basic	X
faz.net	6	48	✓	✓	✓	✓	X	basic	X
imgur.com	6	0	!	✓	!	✓	✓	2class	X
zdf.de	1	50	✓	✓	✓	✓	✓	basic	X
aol.com	8	16	✓	✓	✓	✓	✓	basic	✓
lidl.de	6	0	!	!	!	✓	✓	lidl	X
immowelt.de	7	50	✓	✓	✓	✓	✓	basic	X
stackoverflow.com	8	0	✓	✓	✓	✓	✓	2class	X
adobe.com	8	0	!	!	✓	✓	✓	3class	✓
tchibo.de	8	0	✓	✓	✓	✓	✓	basic	X
mediamarkt.de	8	15	!	✓	✓	✓	✓	mediamarkt	X
dropbox.com	6	0	✓	✓	✓	✓	✓	basic	X
hm.com	6	25	✓	✓	✓	✓	✓	basic	X
tagesschau.de	1	30	✓	✓	✓	✓	✓	basic	X
wetter.de	6	0	✓	✓	✓	✓	✓	basic	X

Table 14.1: Reverse-Engineered Password Policies. **Head Row:** TLD = top level domain. <> = supports unicode UTF8/16, C = complexity,  = utilizes dictionary checks. **Content:** L = lower case letters, U = upper case letters, D = digits, S = symbols, ✓ = allowed/yes, ! = mandatory, (X) = restricted usage, X = forbidden/no.

[illegible]

**Figure 14.4:** List of 46 passwords used to reverse-engineer password policies.



## Password Personality

	emoji12	2word12	3class12
(Intercept)	9.99*** (2.35)	7.04*** (0.75)	6.26* (2.52)
Age	0.02 (0.06)		-0.05 (0.06)
GenderFemale	0.91 (0.58)	1.35* (0.66)	-0.46 (0.61)
ITNo	-0.10 (0.59)	0.60 (0.66)	0.21 (0.63)
Extraversion	-0.05 (0.08)		-0.01 (0.08)
Conscientiousness	0.01 (0.09)		0.08 (0.10)
Neuroticism	-0.22* (0.09)		0.05 (0.09)
emojiPos2	0.52 (0.65)		
emojiPos3	0.23 (0.63)		
twoWordPos2		-0.38 (0.74)	
twoWordPos3		-0.07 (0.73)	
threeClassPos2			1.19 (0.66)
threeClassPos3			0.77 (0.69)
AIC	591.89	615.92	604.37
Deviance	772.78	899.67	890.95
Deviance explained	0.15	0.21	0.10
R <sup>2</sup>	0.04	0.08	0.00
GCV score	8.47	10.44	9.36
Num. obs.	119	119	119
Num. smooth terms	2	6	2

\*\*\*  $p < 0.001$ , \*\*  $p < 0.01$ , \*  $p < 0.05$

**Table 14.2:** GAM summary of difficulty to create a password in Personality study 1.

	emoji12	2word12	3class12
(Intercept)	123.77*** (28.79)	98.71** (34.57)	159.83*** (45.19)
Age	1.55 (0.95)	0.12 (0.90)	
GenderFemale	19.92* (9.87)	29.00** (9.16)	10.99 (10.57)
ITNo	-6.83 (10.11)	14.37 (9.58)	15.64 (10.99)
Conscientiousness	-3.58* (1.58)	-3.40* (1.49)	-2.09 (1.70)
emojiPos1	-37.24*** (10.95)		
emojiPos2	-51.03*** (10.71)		
EDF: s(Extraversion)	1.44 (1.76)		
EDF: s(Agreeableness)	1.10 (1.20)	1.72 (2.16)	
EDF: s(Neuroticism)	2.84 (3.57)	3.30 (4.13)	
EDF: s(Openness)	1.22 (1.41)		
Extraversion		-0.42 (1.22)	-2.71 (1.40)
Openness		0.86 (1.21)	-1.49 (1.35)
twoWordPos1		-10.78 (10.57)	
twoWordPos2		-33.43** (10.52)	
Agreeableness			3.40* (1.64)
Neuroticism			-3.33* (1.64)
threeClassPos1			-7.15 (11.52)
threeClassPos2			-41.44*** (11.99)
EDF: s(Age)			2.76** (3.43)
AIC	1266.94	1252.46	1283.54
BIC	1307.51	1294.21	1321.78
Log Likelihood	-618.87	-611.20	-628.01
Deviance	229220.93	201509.84	267271.98
Deviance explained	0.29	0.26	0.30
Dispersion	2174.73	1919.59	2515.74
R <sup>2</sup>	0.21	0.16	0.22
GCV score	2455.30	2176.04	2817.89
Num. obs.	119	119	119
Num. smooth terms	4	2	1

\*\*\* $p < 0.001$ , \*\* $p < 0.01$ , \* $p < 0.05$

**Table 14.3:** General Additive Models for **Time to Create** passwords in first personality study.

---

A. parametric coefficients	Estimate	Std. Error	t-value	p-value
(Intercept)	62.2292	13.0463	4.7699	< 0.0001
B5_Extraversion	-0.0239	0.1313	-0.1823	0.8557
B5_Agreeableness	0.3349	0.2140	1.5651	0.1211
B5_Neuroticism	-0.0327	0.1630	-0.2003	0.8417
B5_Openness	-0.3655	0.1526	-2.3951	0.0187**
D_Age	-0.1144	0.1150	-0.9954	0.3222
D_GenderFemale	0.8796	2.1791	0.4037	0.6874
D_ComputerScienceBackgroundYes	-2.9237	2.2227	-1.3154	0.1917
B. smooth terms	edf	Ref.df	F-value	p-value
s(B5_Conscientiousness)	1.8036	2.2885	0.4180	0.5944

**Table 14.4:** GAM fit for “overall strength ratings” with Big-Five trait scores as covariates. Conscientiousness was smoothed, because it shows a non-linear association with overall strength ratings.

# GLOSSARY

ASCII	American Standard Code for Information Interchange
CHI	ACM CHI Conference on Human Factors in Computing Systems. Largest venue of research in Human-Computer Interaction
CMU	Carnegie Mellon University (Pittsburgh, Pennsylvania, USA)
ESM	Experience Sampling Method
GAM	generalized additive model
HCI	Human-Computer Interaction
HIT	Human Intelligence Task
IRB	Institutional Review Board
ISO	International Organization for Standardization
LUDS	lowercase, uppercase, digits, symbols
mTurk	Amazon Mechanical Turk. Crowd-Sourcing platforms where workers (“turkers”) complete micro tasks and receive a small payment.
NIST	National Institute of Standards and Technology.
P4P	Persuasive Design for Password Support
PAF	Persuasive Authentication Framework
PANAS	Positive Affect and Negative Affect Scale
password manager	Password Manager. Software that supports a user in the task of managing credentials. Can be standalone or built into web browsers. Famous examples for standalone password managers: LastPass, 1Password, Dashlane, Keepass, RememBear
PCFG	Probabilistic Context-Free Grammar. Statistical grammar model. In password studies, PCFG algorithms can be adapted to measure the guessability of a given password and calculate a guess number.
PD	Persuasive Design
persona	Fictional character that represents a market or user segment during a user-centered design process

---

PGS	Password Guessing Service. Service that ‘estimates plaintext passwords’ guessability: how many guesses a particular password-cracking algorithm with particular training data would take to guess a password. <a href="https://pgs.ece.cmu.edu/">https://pgs.ece.cmu.edu/</a>
PII	personally identifiable information
PIN	personal identification number
PSM	password strength meter
PWM	password manager
REML	Restricted maximum likelihood
SeBIS	Security Behavior Intentions Scale
service provider	Entity providing access to a resource through a specific service, e.g. the operator of a news website.
SSO	Single-Sign On
UI	user interface
Unicode	Standard to consistent encode, represent, and handle digital text
USEC	Usable Security and Privacy
UX	user experience
W3C	World Wide Web Consortium
WPA	WiFi Protected Access. Protocol used in access control for wireless networks.

## Eidesstattliche Versicherung

(Siehe Promotionsordnung vom 12.07.11, § 8, Abs. 2 Pkt. 5)

Hiermit erkläre ich an Eidesstatt, dass die Dissertation von mir selbstständig und ohne unerlaubte Beihilfe angefertigt ist.

München, den 26. April 2016

Tobias Seitz