2024

Mobile Authentication

LEVERAGING MOBILE DEVICES FOR SECURE ACCESS

JAMES R. BEBARSKI

NORTHEASTERN UNIVERSITY - CY5010: INFORMATION ASSURANCE - FINAL PROJECT

Repository: https://bit.ly/securetouch-repo
Web Interface: https://bit.ly/securetouch-web-interface
Android APK: https://bit.ly/securetouch-apk
Presentation: https://bit.ly/mobile-auth-presentation

Table of Contents

Introduction:

Mobile devices have transformed many aspects of our daily lives throughout the last few decades, providing unprecedented social and economic benefits and ease of access to information. Although mobile devices provide wide-ranging benefits, they do not come without cost. The increasing dependence on mobile devices has introduced us to potential vulnerabilities in our security. Despite the potential for jeopardizing our privacy and data, mobile authentication is not perfect. As I have learned, most people do not even take the basic steps to secure their devices. Mobile devices are not just tools but extensions of our personal and professional lives, necessitating strong measures that do not compromise usability.

As someone who enjoys mobile application development and is relatively new to the world of cyber security and information assurance, I decided to explore authentication methods utilized by mobile devices – with a specific focus on Android devices, by assessing existing weaknesses and identifying potential areas for improvement. I hope to find a path toward more secure and user-friendly authentication experiences and, at the very least, apply what I learned in my future application development ventures.

Why does this project interest me?

My interest in mobile authentication stems from its critical role in today's digital landscape, particularly in identity protection, the evolving role of artificial intelligence (AI), and the potential for new MFA methods. As a mobile app development enthusiast new to information assurance and cybersecurity, I find mobile authentication and keeping our devices secure a topic worth exploring further. As I have learned, most forms of authentication we use on our mobile devices are not very secure as single factors, so I was pretty interested in how newer technologies in behavioral analysis or biometrics can offer greater convenience and security when layered with current forms of authentication to provide strong 2FA or MFA solutions.

What do I hope to learn?

Mobile devices, and therefore mobile device authentication (if you even use it), are engrained in our day-to-day lives. I often catch myself just opening my phone out of habit. I think about how many times I open my device and assert that I am who I say I am, many times per day, for something as simple as looking at text, checking Instagram, or even looking at our bank balance. In short, I want to explore how I can ride that fine line of security and convenience, especially as I move forward in the future, developing my own Android and potentially even iOS applications.

What questions do I hope to explore?

- What are some vulnerabilities in various forms of mobile device authentication?
- What new, 2FA, and MFA authentication methods are being developed for mobile devices, and how do they compare to some of the more traditional mechanisms?
- How do we navigate the delicate balance between convenience and security in mobile device authentication, ensuring that security measures protect us without being too overwhelming or overkill?
- In what ways can emerging technologies like AI, biometrics, and behavioral analysis provide a more secure and convenient user experience?

Objectives:

Primary:

- Conduct a review of existing mobile authentication technologies.
- Explore mechanisms for mobile authentication, assessing strengths and weaknesses.
- Leveraging my current Android and web development skills, develop my MFA authenticator application.
- Evaluate the effectiveness and usability of the developed prototype app.

Secondary:

- Investigate user acceptance and readiness for adopting authentication techniques.
- Assess how different Android device capabilities (e.g., sensors, OS versions) affect the feasibility and reliability of new authentication methods, considering ease of use, accessibility, and user feedback mechanisms.

Tertiary:

- Explore the potential for implementing the developed authentication methods across other platforms (iOS, wearables).
- Investigate emerging innovations in hardware that could be integrated into Android devices to support advanced authentication processes and methods, enhancing security without compromising convenience.

Literature Review

Apple Explained. (2022, September 5). *Why iPhones Do not Have Pattern Unlock*. Retrieved from YouTube: https://youtu.be/OPZMNtAW4MM?si=gz7sJQH9loVxn-Zi

- *A short video provides insight into why Apple did not adopt pattern-based authentication.*

Apple Support. (n.d.). *About Optic ID advanced technology*. Retrieved from support.apple.com: https://support.apple.com/en-us/118483

- *Apple support documentation briefly describes how Apple Visions Optic ID works.*

Aratek. (2023, January 1). *The Fingerprint File: 4 Fingerprint Sensor Types*. Retrieved from aratek.com: https://www.aratek.co/news/the-4-fingerprint-sensor-types

- *Resource I used to gain a better understanding of fingerprint sensor hardware.*

Aviv, A. J., Davin, J. T., Wolf, F., & Kuber, R. (2017). Towards Baselines for Shoulder Surfing on Mobile Authentication. *ACSAC '17: Proceedings of the 33rd Annual Computer Security Applications Conference* (pp. Pages 486–498). Association for Computing Machinery.

- *Provided a lot of insight into the effectiveness of PIN and pattern-based lock screens in protecting against shoulder surfing attacks. Cited in my "something you know" section.*

Aviv, A. J., Gibson, K., Mossop, E., Blaze, M., & Smith, J. M. (2010). Smudge Attacks on Smartphone Touch Screens. *4th USENIX Conference on Offensive Technologies.* USENIX Association.

- *I actually discovered this when looking into related works of Aviv, one of the researchers mentioned in the previous work. It educated me on the effectiveness of smudge attacks on smart devices to infer PINs and patterns.*

BBC. (2019, December 20). *Facial recognition fails on race, government study says*. Retrieved from bbc.com: https://www.bbc.com/news/technology-50865437

- *In research into the effectiveness of facial recognition on mobile devices, I stumbled into this when researching algorithmic bias.*

Boonkrong, S. (2020). *Authentication and Access Control: Practical Cryptography Methods and Tools*. Apress.

- *This was probably the most helpful text I read on the topic of authentication. I found its sections on threats to authentication, password-based authentication, biometric authentication, and MFA to be most helpful. The biometric section goes into some decent depth on performance metrics for biometric authentication, like the Failure to Enroll Rate, Failure to Acquire Rate, False Acceptance, and False Rejection.*

Bursztein, E. (2014, March). *Survey: most people do not lock their android phones - but should*. Retrieved from elie.net: https://elie.net/blog/survey-most-people-dont-lock-their-android-phones-but-should

- *I referenced this study in the presentation, and my write-up on authentication is below. This consumer study provided much context as to how much Android users have changed in how they secure their devices in the last 10 years. Especially since the introduction of fingerprint lock screens.*

CPI OpenFox. (2023, July 17). *History of Fingerprinting*. Retrieved from openfox.com: https://www.openfox.com/history-of-fingerprinting/

- *I was pretty interested in the history of fingerprint use in law enforcement and found this to be a great resource.*

Crumb. (2023, March 19). *The Honor Student Caught Stealing $7.5 Million Dollars*. Retrieved from YouTube.com: https://youtu.be/A-nIjUIFARA?si=fOKJw9fRfussZM6T

- *I initially learned about SIM swapping from this video, which led to further research into the origin of SIM swapping with our smart devices in 2018-2019.*

Donnellan, A. (2020, June 24). *How you can a-Void a voice spoofing attack*. Retrieved from csiro.au: https://www.csiro.au/en/news/all/articles/2020/june/how-you-can-a-void-a-voice-spoofing-attack

- *A write-up on a system intended to detect voice recognition spoofing.*

Fanti, M. (2023). *Implementing Multifactor Authentication*. Packt.

- *Although later chapters discuss more industry specifics, I found this text's earlier chapters on authentication in general and different situations when you should use other types of MFA to be pretty helpful. However, none of this text was directly cited in my write-up. Later chapters discuss third-party authenticators like Duo and Azure ID and authentication services.*

Faresse, M. (2020, May 21). *The Most Common Facial Recognition Spoofing Methods and How to Prevent Them*. Retrieved from dormakaba.com: https://blog.dormakaba.com/the-most-common-facial-recognition-spoofing-methods-and-how-to-prevent-them/

- *A very enlightening read on how attackers will spoof facial recognition technologies and discussed static 2D and static 3D spoofing. This is cited directly in my facial recognition section.*

Franceschi-Bicchierai, L. (2019, February 1). *Hacker Who Stole $5 Million By SIM Swapping Gets 10 Years in Prison*. Retrieved from vice.com: https://www.vice.com/en/article/gyaqnb/hacker-joel-ortiz-sim-swapping-10-years-in-prison

- *More specific article on Joel Ortiz, the first person to be charged with SIM swapping in the United States.*

Glover, J. D., Sudderick, Z. R., Bo-Ju Shih, B., Bath-Samblas, C., Charlton, L., Krause, A. L., . . . Headon, D. J. (2023). The Development Basis of Fingerprint Pattern Formation and Variation. *50cell*, Volume 185, ISSUE 5.

- *This research discusses what is called Turing reaction-diffusion systems, which I reference in the section on fingerprint biometrics. They researched how ridges and patterns of fingerprints formed uniquely in mice.*

Grimes, R. A. (2020). *Hacking Multifactor Authentication*. Wiley.

- *Although this text was more specific to hacking MFA generally and not entirely specific to mobile devices, this was still probably my most helpful text for understanding some vulnerabilities in different forms of authentication. The section on social engineering, SMS attacks, and biometric attacks was most beneficial, and I believe I specifically most frequently throughout.*

Han, Q., Mandujano, S., Ports, S., Subrahmanian, V., & Tetali, S. D. (2023). *The Android Malware Handbook*.

- *Another excellent text that covers Android Malware specifically is an excellent resource for learning about the various forms of Malware that threaten mobile authentication. The sections on Spyware and abuse of permissions in Android were very enlightening. I cited this text very frequently in the Something You Have subsection on the threat of Malware to mobile authentication.*

Hristov, V. (2022, August 16). *Android has written off Face ID way too soon*. Retrieved from phonearena.com: https://www.phonearena.com/news/Android-has-written-off-Face-ID-way-too-soon_id141933

- *Provided some background on the history of facial recognition support by Android.*

HYPR Team. (2022, April 28). *Six of the Biggest Problems with Password Managers*. Retrieved from securityboulevard.com: https://securityboulevard.com/2022/04/six-of-the-biggest-problems-with-password-managers/

- *A blog post discussing some significant concerns with password managers.*

IANS. (2023, May 8). *A new Android malware was discovered that stole your passwords and 2FA codes*. Retrieved from business-standard.com: https://www.business-standard.com/technology/tech-news/new-android-malware-discovered-that-steals-your-passwords-2fa-codes-123050800681_1.html

- *Example of malicious apps pretending to be real apps and distributed via phishing email. This malware also requested SMS access and intercepted 2FA codes to hijack accounts.*

Kaspersky. (n.d.). *What is SIM Swapping?* Retrieved from Kaspersky.com: https://www.kaspersky.com/resource-center/threats/sim-swapping

- *More in-depth write-up on how SIM swapping works and ways that organizations and individuals can combat SIM swapping.*

Schleier, S., Holguera, C., Mueller, B., & Willemsen, J. (2023). *OWASP Mobile Application Security*. Retrieved from https://mas.owasp.org/: https://mas.owasp.org/MASTG/

- *A very in-depth technical guide on Android security testing, android architecture, application structure, and attack surfaces. The sections related to Android Data Storage, Cryptographic APIs, and Android Local Authentication helped me significantly in developing the SecureTouch prototype.*

Smith, Z. S. (2022, April 22). *Google Reportedly Bans Dozens Of Apps Containing Spyware*. Retrieved from forbes.com: https://www.forbes.com/sites/zacharysmith/2022/04/06/google-reportedly-bans-dozens-of-apps-containing-spyware/?sh=94ed17d26578

- *An article detailing how even malicious apps can sneak their way into Google Play.*

The Verge. (2024, January 9). *The Rabbit R1 is an AI-powered gadget that can use your apps for you*. Retrieved from theverge.com: https://www.theverge.com/2024/1/9/24030667/rabbit-r1-ai-action-model-price-release-date

- *Article discussing the Rabbit R1, a new AI-powered assistant that actually works by directly accessing your applications without the use of APIs.*

Turgeman, A. (2018, January 18). *Machine Learning And Behavioral Biometrics: A Match Made In Heaven*. Retrieved from forbes.com: https://www.forbes.com/sites/forbestechcouncil/2018/01/18/machine-learning-and-behavioral-biometrics-a-match-made-in-heaven/?sh=283b277e3306

- *Interesting article from Forbes on ML and behavioral biometrics.*

Weatherbed, J. (2023, September 12). *10 years ago. Apple finally convinced us to lock our phones*. Retrieved from theverge.com: https://www.theverge.com/23868464/apple-iphone-touch-id-fingerprint-security-ten-year-anniversary

- *Another article that I cited in my presentation essentially argues that the addition of fingerprint authentication not only inspired people to use fingerprint authentication but created an evolution of users locking their screens through some means of lock screen authentication.*

Weinert, A. (2019, July 9). *Your Pa$$word doesn't matter* . Retrieved from techcommunity.microsoft.com: https://techcommunity.microsoft.com/t5/microsoft-entra-blog/your-pa-word-doesn-t-matter/ba-p/731984

- *A blog post written by Microsofts current VP of Identity Security at Microsoft, describing some common problems with passwords as a standard form of authentication.*

Wood, A. (2024, March 29). *First Human Patient to Receive a Neuralink Brain Implant Used it to Stay Up All Night Playing Civilization 6*. Retrieved from ign.com: https://www.ign.com/articles/first-human-patient-to-receive-a-neuralink-brain-implant-used-it-to-stay-up-all-night-playing-civilization-6

- *A very wholesome article describing what, so far, is the first successful human trial of the brain-computer interface, Neuralink. This is referenced in the section on BCI-based authentication.*

Yirka, B. (2023, February 10). *How Fingerprints Get Their Unqiue Whorls*. Retrieved from phys.org: https://phys.org/news/2023-02-fingerprints-unique-whorls.html#:~:text=The%20random%20placement%20of%20the,to%20those%20of%20hair%20follicles.

- *Article that discusses how fingerprints actually get their unique characteristics. I learned of the research "The Development Basis of Fingerprint Pattern Formation and Variation," which I cited earlier, from this article.*

Zhang, F., Kondoro, A., & Muftic, S. (2012). Location-based Authentication and Authorization Using Smart Phones. *11th International Conference on Trust, Security and Privacy in Computing and Communications* (pp. pp. 1285-1292). Liverpool: IEEE.

- *Discusses some security concerns of location-based authentication via smartphones.*

Research Questions

User Behavior and Adoption:
How does the trade-off between convenience and security affect the choices people make when setting up different forms of mobile authentication? What factors influence people's willingness to adopt 2FA or MFA methods on their mobile devices?

Technical Considerations in Biometrics:
What are the technical limitations of current biometric technologies on Android devices, and how can they be overcome? What are some effective fallback mechanisms if the primary form of biometric authentication fails?

Security Evaluations:
What are some common security concerns with various forms of mobile authentication? Recognizing the strengths and weaknesses of different categories of mobile authentication, how can these systems complement one another to enhance our overall personal security?

Emerging Needs:
What are some emerging needs in mobile authentication, and which upcoming technologies might require new authentication solutions?

Comparative Analysis:
How does my application prototype (SecureTouch) compare against existing authenticator applications and mobile applications' built-in authentication features? What are some improvements that can be made to my prototype to enhance its functionality?

Overall, I hope to answer these questions in my presentation of findings or through a practical demonstration of ideas I have drawn and distilled into my final Android authenticator prototype, which I call SecureTouch. I also hope that my final in-class presentation will provide an excellent opportunity to get some constructive criticism on my prototype.

Methodology

Research and Exploration

For the initial phase of my project, I undertook a comprehensive literature review and analysis of mobile authentication technologies. The primary goal was to identify vulnerabilities and challenges associated with mobile authentication, characterized by something you know, something you have, and something you are authenticating.

Something You Know (SYK): I primarily investigated the implications of traditional knowledge-based authentication, such as PINs, passwords, and patterns. This included exploring vulnerabilities like shoulder surfing, smudge inference, and brute force.

Something You Have (SYH): For this section, I investigated concerns related to the physical possession of a device, such as loss, theft, SIM swapping, malware, and social engineering.

Something You Are (SYA): For the final portion of my exploration, I examined mobile authentication methods that use biometrics, such as fingering and facial recognition.

Development of SecureTouch Prototype

Building on the insights gained from the research and exploration phase of my work, the project transitions into the practical application of my findings through the development of what I call SecureTouch. SecureTouch is a mobile authenticator application that is designed to leverage PIN and fingerprint to authenticate web application logins.

The development of SecureTouch required two main activities:

- There is a web interface that primarily acts to register an account and provide a platform for web logins. This interface will mainly serve as a platform to test the authenticator application, which was developed for Android devices. The web application uses a React frontend, tailwind for styling, and Firebase services like Firebase Authentication for handling email/password authentication and Firestore for storing simple user details and login attempt documents. Firebase makes cross-platform development much easier since I could use it as a shared backend for the web interface and the Android application.
- Creating an accompanying Android application where users can set up their accounts for further authentication via their login information and backup phrases. For additional authentication into the web application, users can provide a PIN and fingerprint to authenticate web logins. The application was developed via Android Studio, with some Material Design components, and like the web interface, it leverages Firebase Authentication and Firestore for backend functionality.

Mobile Authentication: Now and In the Future

Most of my exploration for this project involved learning more about Android and how I could leverage the sensors in Android devices to develop my authenticator app prototype. Outside of programming my Android-based authentication app – something I genuinely enjoyed – I did learn quite a bit about authentication, specifically mobile authentication. Unfortunately, not everything I read could be applied to the practical portion of my project, and I still felt it would be good to provide some insight into the rest of what I learned throughout my studies. The practical portion/prototype I developed is merely a prototype, birthed by the amalgamation of practices I would like to see in mobile authenticators.

For this write-up, I will stick with the standard categorization of "something you know," "something you have," something you are." However, I will try to stick with these forms of authentication as they relate to mobile devices like smartphones, smartwatches, or even the newer AI assistants like the Rabbit R1. If I were to discuss these categorizations in length, outside of the scope of just mobile devices, it would be general and frankly not very useful to anyone familiar with authentication. Let us start by examining "something you know."

Something You Know: PINs, Patterns, and Passwords

Lock screens are the main point of entry for mobile devices. They offer a relatively straightforward and crucial barrier to unauthorized access. In the realm of "something you know" authentication (or SYK, as I will abbreviate it going forward), Android devices offer three standard forms of authentication – PIN, pattern, or password. From what I have read in readings in the course textbook and from my exploration, the consensus in the world of info assurance is that passwords are not ideal (and that is putting it gently). It is almost like a Mandela effect; I often feel like I just imagined that I read that experts have explicitly said passwords are bad or they do not matter. However, I was not just imagining it; in a blog post on Microsoft's Entra Blog, Alex Weinert – the current VP of Identity Security at Microsoft – said just that. In the post "Your Pa$$word doesn't matter", Weinert states explicitly in the conclusion that "your password, in the case of breach, just doesn't matter – unless it is longer than 12 characters and has never been used before". Out of context, this can seem harsh, but his conclusion was based on the idea that most normal people are picking weak, common, or reused passwords. For even more context, this article is specific to server security, where you can expect a higher degree of awareness on the administrator's behalf for securing their data. He later discusses that multifactor authentication can be our saving grace, but unfortunately, that is not an option for mobile device authentication for the average person. As for the everyday user, convenience is a priority, especially regarding lock screens. I was shocked to learn that most people do not even use a PIN, pattern, password, or biometric, so how can we expect people to use authentication on their lock screens?

According to a Google Consumer Survey conducted on Android users in 2014, 52.0% percent of users had no PIN, pattern, password, or face recognition authentication in their lock screen. 25.5% of respondents used pattern-based authentication, 15.1% used PINs, 3.3% used passwords, and 2.3% used facial recognition (Bursztein, 2014). I will say that this data is a bit dated, and it did not account for fingerprint authentication fingerprint sensors, as they were only introduced to Android with the Samsung Galaxy S5 in April 2014. However, it still demonstrates the persistent underutilization of even the most basic security measures for mobile devices at the lock screen level. Apple was slightly ahead of the curve by including a fingerprint scanner in the iPhone 5S, released in September 2013. Fingerprint scanning considerably changed how consumers protect their devices, which is something I will discuss in more depth within the "something you are section."

The preference for patterns and PINs at the time, despite their vulnerability to attacks like shoulder surfing and smudge guessing, underscores the convenience vs. security trade-off. Patterns and PINS are so easily memorable and quick to enter and remain popular despite their lower security compared to more complex passwords or biometric alternatives in the early 2010s.

Although using one of these measures is better than having no-lock screen authentication, that is not to say they are perfect (they are not). PINs and patterns are more susceptible to shoulder surfing, smudge guessing, and guessing attacks, especially if the PIN is easily recognizable or guessable, like 1234 or the typical L shape many people still use in pattern-based lock screens. Although it would be more secure, imagine how annoyed you would be if your new device required a strong password with solid password checks (like a minimum number of characters, using special characters, capitalized letters, and other rules).

I never used a pattern for my lock screen, primarily because I have used an iPhone as my daily carry since my parents thought I was old enough to own a cell phone. The iPhone does not even provide pattern-based lock screen authentication, which made me curious. A YouTube channel called @AppleExplained, run by Greg Wyatt, provides an excellent and short explanation, but the research he cited was more interesting. The research in question had participants play the role of attackers, who would need to try and determine the input a victim entered on different phones and angles over video in-person and online (emulating shoulder surfing). They tested four and six-digit PINs as well as patterns. Here were their findings:

Table 1 – Online Viewing Results

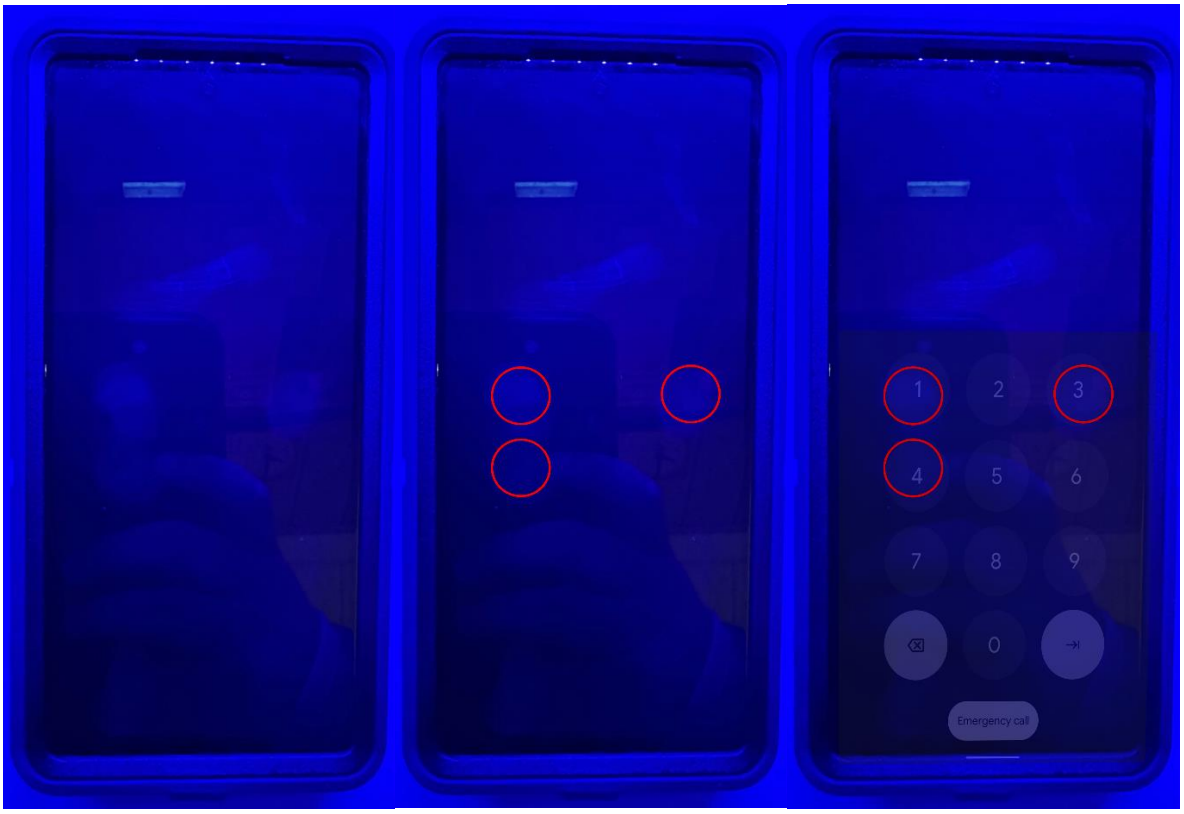
	Single View (4-Length)	Single View (6-Length)	Multiple Views (4-Length)	Multiple Views (6-Length)
PIN	34.92%	10.86%	56.72%	26.53%
Pattern	80.9%	64.2%	88.07%	79.85%

Aviv, A. J., Davin, J. T., Wolf, F., & Kuber, R. (2017). Towards Baselines for Shoulder Surfing on Mobile Authentication.

I have included the results for the study's online viewing participants in the above table. Based on their results, 6-digit PINs were the most effective at deterring the participants. However, the main takeaway here, I believe, is just how weak patterns were even when users were utilizing the max pattern lengths. After only a single view, participants could learn the correct pattern 80.9% of the time when a user had the minimum pattern length. Even more surprising, attackers could still determine the correct pattern after the first viewing 64.2% of the time, even if the users were using the maximum pattern length.

My main takeaway from that study was that Apple probably opted out of providing a pattern-based lock screen authentication option because it is ineffective at protecting users' devices, compared to PINs. However, 4-digit PINs were not solid either. So be wary of any pesky significant others or anyone around you when you have your phone out. It is just an observation, but think about how many times you have been out in public and how many times you have unlocked your device to check your phone. Someone could be watching, and if you are using a weak PIN or any pattern, you are highly vulnerable to shoulder surfing!

Another form of attack that I learned about in my exploration was smudge inference. These attacks can be as simple as examining a phone to see where a higher frequency of smudges exists or even more advanced methods by sophisticated attackers using some AI. I decided to investigate smudge inference attacks firsthand, as I was initially skeptical. Still, I found it a relatively powerful inference-based attack against mobile lock screens. To explore this attack myself, I asked to borrow a relative's Pixel 6 to see what I could learn from the streaks of their device. After opening their lock screen, I discovered they used a PIN-based lock screen mechanism. I read a few sources online that talked in depth about optimal lighting angles, lumens, and light colors. However, I just decided to wing it and go at it myself. For my simple experiment, I tried using just a generic flashlight at different angles with the lights off, and then I tried a UV flashlight and found that to be less useful. Eventually, I decided to use my work desk's color-changing LEDs and found that blue light provided the clearest smudging for the naked eye.



The leftmost image is just the device itself under a blue light. You can discern a few streaks, but we want to determine where the PIN entry prompt would be. It might be challenging to see in the images, but I could discern three smudges that would roughly be where the PIN entry is on their device in the middle image. On the far right, I overlayed the pin entry from an Android lock screen over the circled prints/smudges of interest. I could discern that their PIN may contain the numbers 1, 3, and 4. Funny enough, I confirmed that my relative's PIN contained just these three digits. They have since changed their PIN. For some perspective:

- For a 4-digit PIN: $3^4 = 81$ possible permutations
- For a 6-digit PIN: $3^6 = 729$ possible permutations

So, if I really wanted to break into my relative's device and had, let us say, at least ten opportunities a day to attempt, I could probably figure out their PIN at minimum in 8 days. Smudge guessing can also be applied to pattern-based authentication mechanisms, with possibly even more efficiency due to the restrictions of patterns. On Android devices, there is a 3x3 grid of contact points, and a pattern must meet three rules: the pattern must contact minimally four points, a contact point may only be used once, and there must be an intermediary point between two other contact points. Based on these rules and restrictions, 389,112 possible patterns can exist. (Aviv, Gibson, Mossop, Blaze, & Smith, 2010). Given the rules, though, it gives a potential smudge guess attacker a more likely pool of patterns to compare smudges against.

Unlike PINs or patterns, passwords are not as commonly used for lock screens, but it is probably the safest option if you are not using biometrics. Passwords offer a blend of character choices, which undoubtedly enhance your security compared to PINs or patterns due to the difficulty of guessing a password. Although there are experts who say we should move on from passwords in place of other mechanisms, they are also still very crucial for protecting access to sensitive applications on a device past the lock screen, like banking, email, and social media apps where unauthorized access could lead to privacy breaches and financial loss.

Moving away from talking specifically about lock screens, generally, managing strong passwords presents a unique challenge on pretty much every platform, and considering that more than 65% of users reuse the same passwords across accounts, (HYPR Team, 2022) This is evident. Especially on mobile devices, the interface and user experience can make complex password entry cumbersome and deter users from adopting multiple passwords. Apple and Google have tried to help users by introducing Google Password Manager and Apple's iCloud Keychain. iCloud keychain saves passwords and account info, then auto-fills them when required across all a user's Apple devices. It will also generate unique passwords for each account, addressing the reuse issue. Android's answer to password management works similarly, and it is integrated into the entire Google ecosystem, meaning it works on your phone and in conjunction with Chrome, Gmail, and other apps to store and autofill passwords. It, too, suggests strong passwords, and users can manage them through Google's Password Manager website.

Despite the advantages of password managers, they are not perfect. For example, if an attacker gains access to your account, they change your master password and ultimately take control of all your stored passwords, making them a prime target for attackers. Therefore, your master password should undoubtedly be a strong one. However, knowing the standard user, their master password probably is not as strong as it should be. Aside from a bad actor, users can forget the password for their password manager entirely, and recovery can be difficult. Losing your device is also very problematic.

The password managers are also subject to vulnerabilities outside of a user's knowledge and control. Bugs are a significant concern; for example, in 2021, a bug tracked as CVE-2021-40539 was discovered in Zoho's ManageEngine, which provided password management and single sign-on functionality. The bug could allow attackers to exploit REST API authentication, enabling them to launch attacks anywhere the application was present. Very recently, it was discovered that a few Android password managers, such as 1Password, LastPass, and others, shared a vulnerability called AutoSpill. The problem was when Android's WebView interacted with the password managers, which allows apps to display web content within an app. If the app that initiated a WebView was not trusted, it could lead to credential theft, as the autofill process shared credentials with the WebView, which can then mean the untrusted app could access the credentials.

The above incidents highlight the ongoing battle between evolving security measures and the persistence of vulnerabilities in mobile authentication. The cat-and-mouse game between security professionals and potential attackers emphasizes the importance of vigilance from developers and users.

To summarize, here are my takeaways from my dive into SYK mobile authentication. If you have a choice between PINs, patterns, or passwords on your lock screen and applications that provide them as entry options, I hate to say it, but passwords take the cake.

Passwords > PIN > Pattern

Do not ever use patterns. Please. Outside of lock screens, always use 2FA or MFA.

Something You Have: Your Smart Devices

Smartphones, tablets, and smartwatches are at the heart of “something you have” (SYH) authentication. These devices are often coupled with authenticator apps and authenticator functionality within specific mobile applications. They form a critical line of defense in our digital lives. However, our smart devices are still susceptible to problems like loss, theft, phishing, SIM swapping, malware, and social engineering. Throughout this section, I hope to provide some insight into these concerns.

Loss and Theft

2FA or MFA authentication that involves access to your phone means nothing if you do not have access to the device. Losing a device can be a problem for many reasons; first and foremost, any services you use that require some form of mobile authentication are inaccessible. It can be quite a pain to resolve some of these issues, as most authenticator apps like Microsoft’s or other popular apps offering one-time-passcodes (OTP) can be challenging to set up on another device. You better hope you wrote down your backup codes or have other means of asserting your identity if you need to set up your 2FA authentication again on a new device. Backup codes are often a second line of defense. However, it seems that not many people even take that basic step. I was curious and ran a quick straw poll on r/cybersecurity, thinking that the results might be skewed because most people in the community most likely understand the importance of saving their backup codes. The results were still surprising: out of 156 respondents, 39.74% said they always save their backup codes, 14.1% responded often, 14.74% said sometimes, 16.67% responded rarely, and 14.74% responded never. Although many people indicated that they always save their backup codes, a significant portion do not permanently save them, even in a community that you could assume is more aware of the importance of backup codes.

What is worse than losing a device is theft, directly or even from someone finding and taking your lost device. An attacker with physical access to a device is a problematic one. First and foremost, it leaves you open to unauthorized access, especially in cases where the user has no lock screen authentication in place (like PIN, pattern, or password). A bad actor can waltz right in. If a bad actor can access your apps, it can grant them broad flexibility for bypassing and 2FA or MFA protections for any account tied to your phone. The mere presence of your applications can provide a treasure trove of information to the possessor of your device. They can learn about all the places they can infiltrate, like what banks you use, investment accounts you have, health records, etc. They can access personal info, like contact info, photos, messages, and stored passwords. With access to your device, bad actors can take over accounts completely and update recovery information, preventing legitimate users from gaining control.

Loss and theft will also expose you to social engineering attacks against others. With access to a victim’s contact info, call history, voicemails, and messages, they can craft clever and convincing messages to people you know to deceive them and expand their reach toward future victims. However, I will discuss social engineering more later in this section.

Sorry for the gloom and doom, but the good news is that there are some ways to mitigate or even prevent a bad actor from doing some of these actions. First and foremost, setting a PIN is one of them. Device tracking is a good fallback, however. Unfortunately, though Android does not natively support location-based authentication as a direct method of securing access to your device, they do support PINs, patterns, passwords, and biometrics. However, Android does what it can to provide developers with the tools to implement location-based authentication mechanisms.

Many enterprises use Enterprise Mobility Management (EMM) and Mobile Device Management (MDM) solutions in corporate environments to enforce organizational access controls based on their geographic location. For example, a mobile device that should only work in certain organizational places might implement one of these systems, such as a specific banking or financial service app, where transactions are blocked if you are outside the company premises.

At a more personal level, location-based functionality might be included as a factor in authentication for mobile apps, such as when you cannot execute transactions on some trading platform unless you are in trusted locations. Even smart locks like Schlage Encode Smart Wi-Fi Deadbolt and Aquara Smart Locks will pair with iPhones, Androids, Apple Watches, and other devices. Merely possessing a device could allow you entry to a home that uses a Smart Lock. Possession and location-based authentication can even be spoofed at the hardware, OS, and application level (Zhang, Kondoro, & Muftic, 2012). So, the moral of the story is do not lose or leave out your device, and if you do, at the very least, use some lock screen authentication like a PIN, password, or fingerprint.

SIM Swapping

Although SIM-swapping attacks were more popular in the late 2010s and early 2020s, that is not to say that they are not a threat to mobile devices and mobile authentication. SIM swapping does require a bit of effort on the attacker's end; they will usually need to collect information about a victim that you would typically need when setting up a cell phone plan. They can do so through several means, but most popularly, they can use phishing or social engineering to get the necessary information. Hackers can even buy sensitive data of victims from others on seedy online marketplaces or forums.

Once the attacker is confident that they have enough details, they will reach out to the victim's cellular provider, often under the premise that they lost their device or want to change their service to a new phone. If the attacker can successfully convince the mobile carrier that they are their victim, they can then have the carrier port the victim’s phone number to the attacker’s SIM or device. From there, the attacker can control the victim's phone number and bypass any 2FA or MFA that involves sending an SMS or OTP containing a code and even authentication methods that require some validation from calling the number. Another problem is that many password resetting mechanisms rely on SMS-based or OTP as a fallback for a user. So, if a bad actor can control your phone number and

email apps, they can access a victim's banking, reset email passwords, and, in some cases, completely take over your accounts.

The first person in the United States to be arrested for SIM swapping was a UMass Boston student named Joel Ortiz. Ortiz was convicted of stealing over $5 million worth of cryptocurrency via SIM swapping throughout 2018 and sentenced to 10 years in prison. Not too long after, many of his accomplices and others who were exploiting the lack of awareness for SIM swapping at the time were also caught and convicted for stealing anywhere between $1 to 17 million dollars. Ortiz and his accomplices started using SIM swapping to hack into Instagram and Twitter accounts, where they would then hijack the handles and sell them. Although Ortiz was not involved, Jack Dorsey (co-founder of Twitter) fell victim to SIM swapping and had his own Twitter account broken into.

Later they got even more cocky and malicious and learned that crypto wallets were highly vulnerable to SIM swapping and begun to steal Bitcoin and other cryptocurrency. Thankfully, their egos were big, and they made their attack methodology way too transparent; they were brazenly flashing wads of cash and buying exotic cars seemingly out of the blue. They also publicized many of their actions on Instagram while masquerading as social media influencers and even went as far as casually sharing their methods with other script kiddies. Had they been less transparent and covered their tracks, who knows how much longer they could have gone on doing this?

After the golden age of SIM-swapping attackers in 2018-2019, many cellular providers and the authorities began to catch on to the issues, and providers began to change procedures to mitigate exploits. Many providers now have callback mechanisms where cellular providers will call the number registered with an account before making changes. They will also require more stringent security measures like tying a PIN to any account change actions and amping up the use of security questions. However, a lot of the onus is on the customers themselves. Kaspersky even recommends switching to authentication apps instead of OTP and SMS-based authentication and avoiding linking accounts to phone numbers altogether. “This way, in the event of a SIM swap fraud, the hackers will have less access to fewer accounts.” (Kaspersky, n.d.).

Malware

There are many threats to mobile devices and authentication, but malware is still a significant concern, as with many other devices. There are many forms of malware and ways to exploit your device. Still, one primary delivery source is malicious applications, which often function as either a form of spyware or a trojan in the context of mobile authentication. Users can be tricked or misled into downloading apps that appear to be trustworthy via email, websites, and third-party app stores. The malware could even be included in the Google Play Store if it is sophisticated enough.

Unfortunately, permissions are a central problem in distinguishing malicious apps from good ones. Applications with malicious code might cleverly offer features that require

permissions for actual functionality; however, the more malicious functionality might also use the same permissions, and there are no natural defense mechanisms for deciding which component of an application's code can use what. Permissions in Android development are often all or nothing; when permitting an app to access one piece of data, you usually permit it to data or sensor usage in all facets of the application.

Although it is not strictly specific to mobile authentication, one great example I learned about in a text called “The Android Malware Handbook” was an app called Qibla Compass Ramadan. It and many other apps were banned from the Google Play Store in 2022. The application would help Muslims schedule prayer and other Ramadan-related activities. If you are unaware, the Quibla is the direction towards the Ka’bah in the Grand Mosque in Makkah, Saudi Arabia. So, as part of the Quibla Compass app’s functionality, it would help point Muslims toward the Ka’bah when they pray. It was later learned that the application was requiring many permissions that it certainly did not need:

```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.READ_PHONE_STATE"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.GET_ACCOUNTS"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
```

Cited From Han, Q., Mandujano, S., Ports, S., Subrahmanian, V., & Tetali, S. D. (2023). *The Android Malware Handbook – Chapter 8.*

Based on the app's intended functionality, READ_PHONE_STATE, READ_CONTACTS, and GET_ACCOUNTS were unnecessary for the app's functionality. It was later discovered that a Panama-based company called Measurement Systems S. de R.L. had paid the developers to include the malicious functionality in gathering unnecessary data like email, phone numbers, contacts, and GPS data. The text also referenced a Forbes article that alleged that the company had links to US intelligence. “The Wall Street Journal found that Measurement Systems was connected via company records and an internet domain registration to a Virginia-based contractor involved in cyber intelligence operations for U.S. security agencies.” (Smith, 2022).

Aside from the clear breach in permissions necessity, it is interesting that the app requires ACCESS_COARSE_LOCATION and ACCESS_FINE_LOCATION, which could feasibly be needed to provide the prayer directions for the Quibla Compass’s prayer direction features. However, there was no way of knowing whether the location access was included in the spyware functionality. Still, based on all the permission requirements, it could be inferred that the spyware was scanning contact lists, phone numbers, and emails and even tracking the location of Muslims using the app, potentially to harvest the social network of individual Muslims to paint a larger picture for US intelligence agencies.

The problem with data brokers is not a new phenomenon by any means. In the early 2017s, companies began building software development kits (SDKs) and paying legitimate

developers to embed them into their applications. The SDKs would then gather information about a user's location history, app usage, and browsing behavior (Han, Mandujano, Ports, Subrahmanian, & Tetali, 2023). OneAudience, founded in 2016, was one of these data brokers that pushed the boundaries in data collection. In 2019, it was discovered that OneAudience was collecting Twitter and Facebook information without any user consent. Facebook later filed a lawsuit, and after a settlement, OneAudience closed. Aside from the apparent breach of privacy, they also collected authentication tokens and then used them to log into victims' Twitter and Facebook accounts to steal personal info. They even stole information about user's emails, call history, contact lists, location, etc. (Han, Mandujano, Ports, Subrahmanian, & Tetali, 2023). Even if you can say, well, an actual company is certainly more trustworthy than some hacker. Well, no, what if somehow this data got into the hands of a more malicious individual or hacker? They could use your data to launch attacks like SIM swapping, which we discussed before. It also raised a whole slew of privacy concerns, which can be a topic of its own.

Moving onto a more recent instance of some Android Malware more specific to concerns of mobile authentication, I read an article from a news outlet called Business Standard about an Android Malware called FluHorse, installed by roughly 1 million users in Taiwan and Vietnam in 2023 (IANS, 2023). The carrier apps masqueraded as ETC, a popular Taiwanese toll collection app, and a popular Vietnamese banking app called VPBankNeo, both a real toll collection app and a bank. Many of the victims received an email requesting that they take immediate action on their accounts, prompting them to download the malicious fake applications. Once downloaded on the victim's devices, the counterfeit apps would request SMS access to intercept 2FA codes and hijack the user's actual accounts. This is just one instance of many where a malicious app is used to break authentication, and it is still one of the most common ways. Keylogging, screen capture, and token theft can also be mechanisms that sophisticated malware might use to bypass two-factor authentication.

So, to summarize, Malware is still a big concern for secure mobile authentication, but as I have learned, the main problems are not always related to red hat hackers that you would imagine. Concerns are also associated with the prevalence of big data companies and government intelligence agencies. Who knows how secure the data they are harvesting is? Much of this data can be utilized as a launch pad to bypass mobile authentication by simply obtaining the data necessary to exploit the victim's two-factor authentication.

Social Engineering and Phishing

A computer security and security consultant, Roger Grimes, stated, "Social engineering is responsible for 70-90% of all malicious data breaches… greasing the skids of attacks and making them more accessible. (Grimes, 2020)." So, what is social engineering? In short, bad actors try to gain their victims' trust so they might divulge sensitive information. In an earlier section on SIM swapping, I discussed how social engineering was just one component of the process, obtaining the information necessary to have a victim's phone number ported to another device. So, while social engineering is not necessarily the prime

means of exploiting mobile devices or mobile authentication, it is undoubtedly an essential piece in an extensive strategy or plan of attack.

Many social engineering attacks involve emails, such as receiving an email from your supposed bank, a Nigerian scammer who claims to be seeking investments for a prince, or people passing themselves off as members of customer service teams. While people are certainly vulnerable to these types of deception on mobile devices, as they might browse their email using a mobile device, these email-based attacks are not particular to mobile devices. What is a concern, though, is social engineering attacks that involve SMS messages. As I mentioned in the SIM swapping section, Kaspersky recommends that people try to avoid using SMS-based authentication where possible, using authenticator apps instead.

Although SMS-based authentication is not perfect, I believe there is still an issue with push notifications. While I think push notifications are still decent alternatives to SMS since they are usually associated with a real app, some problems are still associated with them. One issue involves push-based phishing. Attackers have often compromised passwords or some other standard authentication mechanism and will request push notifications to a user. The hope is that you will approve their malicious login attempt via a push notification sent to their device. I have fallen victim to this with my Chase bank account. Someone requested authentication, and I received a push notification. Since I first thought it was an essential notification from my bank, I clicked it and thought I was logging in to check some messages from my bank. Except when I logged in, I approved their sign-in attempt. Thankfully, I immediately recognized this, changed all my passwords, and called the bank. Still, it just goes to show you that even in the face of 2FA and MFA systems, we need to remain vigilant and be wary of how social engineering plays a role in bypassing these security measures. The best defense against social engineering is awareness.

Something You Are: Biometrics

Regarding mobile authentication, I think there is a significant amount of potential for utilizing biometrics to authenticate, whether at the lock screen or as a feature of authenticator applications. Since biometrics are inherently linked to a user's characteristics, which are unique to oneself, they provide a more secure option than SYK or SYH-based authentication mechanisms. Not only are they linked to our characteristics, but some of these forms of authentication are more permanent (like fingerprint and iris). When we judge authentication methods based on the tradeoff between security and convenience, I believe that biometrics can provide the best equilibrium between security and convenience.

However, that is not to say that they do not come with concerns. Looking at biometrics as a whole, they have "a higher percentage of both false negatives and false positives compared to every other sort of non-biometric authentication. (Grimes, 2020)" . The tradeoff between security and convenience is not always so simple due to issues of high percentage of false negatives and false positives, and biometric authentication mechanisms often need to find a way to balance this. Requiring more sensitive sensor technology will mean a greater frequency of false negatives, and requiring less sensitivity will mean a greater frequency of false positives. Another primary concern is that since biometric authentication is based on one's permanent characteristics, if one's biometric traits are stolen in some way, it will create irreversible problems. If a password or PIN gets compromised, you can always reset it. If your biometric information is stolen, you cannot change it. In this section, I will share what I have learned in my exploration of biometrics and why I believe there is a significant amount of potential in these forms of authentication now and in the future.

Fingerprint

The beauty of fingerprints is that they are unique to everyone, even between identical twins. Funny enough, Alan Turing proposed what is now called Turing reaction-diffusion. He believed that molecules in a developing system promote cell growth while other cells work to prevent it. When examining fingerprints, reaction-diffusion can account for the unique formations of ridges that we see on our fingerprints and palms. In an article on *phys.org* discussing research where scientists examined fingerprints developing in fetal mice, Bob Yirka stated that researchers learned "the random placement of the fingers in the womb as the cells push against one another to form ridges is what makes the patterns so random, and that is why even identical twins have different prints. (Yirka, 2023)."

However, scientists began to examine fingerprints as early as 1686 and observe their uniqueness amongst individuals; this only had practical use when law enforcement realized that prints could be used to solve criminal cases. The first case to be solved using fingerprints by law enforcement was in 1892 in Argentina, where "police discovered a bloody fingerprint on a door frame and analyzed it to identify the murderer (CPI OpenFox, 2023)". Within a few years, law enforcement worldwide began to find similar success. By

1905, the branches of the United States military found fingerprint analysis to be highly successful in identifying individuals, and in 1924, the FBI under Edgar Hoover began to catalog fingerprints. Eventually, fingerprinting became one of the first biometric authentication methods through computing.

Apple was the first company to implement fingerprint authentication via the iPhone 5. Now, almost every decent smartphone you can buy has a fingerprint sensor, revolutionizing smartphone lock screen authentication. Referencing the article from Burztein that I discussed earlier in the SYK section, 52% of Android users did not even lock their phones before fingerprint authentication. Still, by 2021, a study of the most popular lock screen mechanisms discovered that roughly 99% of participants locked their mobile devices, an astonishing difference just eight years later. In the same study, it was learned that about 32% of people used PINs, and virtually the same amount used fingerprints. So, although we cannot confirm for sure that the advent of fingerprint biometrics is responsible for the rise in people locking their devices with some form of authentication, it is clear that there is a stark difference since fingerprint authentication was introduced by Apple in 2013, and later in Android devices starting in 2014.

At first, fingerprint scanners were simple, using a combination of lasers and light—essentially, just optical readers. However, these scanners alone were discovered to have more false negatives and false positives than the combination of sensors used today. They could also be spoofed easily using a simple image placed on the surface. Now, most fingerprint sensors on smart devices require some combination of a 3-dimensional surface which can be detected via capacitive, optical, and ultrasonic sensors, “newer reader types were created explicitly to defeat spoofed fraudulent fingerprints just printed out as images or placed on flat surfaces, like glass or pictures. (Grimes, 2020).” This is not to say that as sensors and scanner technology improve, so will attackers' strategies for trying to spoof fingerprints. Attackers have gone so far as to develop new methods of creating fake fingerprints using combinations of clay, silicon, glue, aluminum powder, and 3D printers with varying levels of success depending on the sensitivity of the sensors.

Although things have certainly improved depending on how much your specific device’s manufacturer has skimped out on hardware, some scanners can be fooled with something as simple as placing a piece of tape or silica on your finger, and more complex methods of spoofing involve stealing a print and creating a fake finger to scan. However, I imagine that when ultrasonic sensors, which are slightly better at detecting the difference between ridges in your print, and thermal sensors, which detect heat, become more cost-effective for manufacturers, many of these problems will be resolved. Ultimately, as more combinations of sensors are integrated and become cheaper in mobile devices, I believe that fingerprint authentication will become more tenable. Apple, as we discussed earlier, benefits greatly by having its dedicated hardware and can control fingerprint authentication quality to a much greater extent. Since Android is an OS, and many manufacturers and

models use the Android OS, the quality of smartphones that use the operating system can vary significantly. Although Google and Samsung put much effort into ensuring quality in their fingerprint technology, you might not be able to say the same of a cheap off-brand phone you found on Temu or Alibaba. You get what you pay for.

You cannot change your fingerprint the same way you can change knowledge-based authentication mechanisms in the case of a breach. Even organizations designed around high security have fallen victim to fingerprint data leaks. According to a text I read called *Hacking Multifactor Authentication* by Grimes, in 2015, the United States Office of Personal Management (OPM) was the target of an attack by a subsidiary of China's Ministry of State Security. Among the leaks were social security numbers, names, and birthdays, and even roughly 5.6 million people's detailed fingerprint data were stolen as part of the attack. This includes some government agents, spies, and informants, some of whom were even living under assumed identities. Eventually, a deal was brokered between President Obama and President Xi Jinping, where China "gave it back." Still, I do not think that we can necessarily trust that the Chinese government destroyed all copies of the stolen information.

Another significant concern with fingerprint-based biometrics is that although your fingerprint's features tend to remain relatively static, over time, your print might change slightly through age, wear and tear, and other factors. However, some devices now have adaptive fingerprint recognition, where machine learning algorithms are applied, and with each scan, it adapts to minute changes and gets better at recognizing your fingerprint. This will not help if your fingerprint gets completely burned or deformed, but adaptive fingerprint recognition algorithms solve the issue of your fingerprint slowly changing over time. I could not pinpoint the first device to implement this feature, but I learned that the Google Pixel 4 implemented adaptive fingerprint recognition, and the Pixel 4 was released in 2019. One can assume that other devices have adopted similar technologies in recent years, but I cannot confirm for sure.

I was interested in how Android actually stores fingerprint data and learned that it actually uses what is called a Trusted Execution Environment for storing fingerprint templates. The idea of Trusted Execution Environments is not unique to Android, but Android refers to their TEE as "Trusty." Trusty runs on the same processor as the Android OS, but it is still isolated via hardware and virtualization. Trusty is compatible with ARM and Intel Processors and works with both of their virtualization environments. In short, when a fingerprint is collected, the unique characteristics (essentially the template of your fingerprint) are processed and then stored in Trusty. In the future, when a user attempts to authenticate, the fingerprint provided will be compared against the template stored in Trusty. If the fingerprint matches the template in Trusty, then Trusty tells the OS to unlock the device and complete the authentication process. The nice part about a TEE, like Trusty storing fingerprint data and other forms of authentication, such as a PIN, password, or pattern, is

that even if your Android device is rooted, your fingerprint data is still secure on your device.

Facial Recognition

To be blunt, facial recognition is not the best in the context of mobile authentication, especially on Android devices; Apple Face ID is pretty strong comparatively. There are so many ways to fool it with facial recognition, and Android face recognition has much greater frequencies of false positives compared to other forms of biometric authentication. Another significant issue is within manufacturing. By nature, Apple has much more control over hardware since they are the ones who control the manufacturing of all their devices; they can create dedicated hardware. Apple knows and controls what hardware is going into their device, but with devices that use Android, coming from a wide range of manufacturers, the level of quality control is out of Google's hands. Overall, Apple provides much higher quality 3D recognition hardware, making it feasible to continue using facial recognition for lock screens and even authentication in banking, trading, and healthcare-related applications. "Face ID on iPhones is a whole system of sensors including a flood illuminator that projects 30,000 dots to map your face in 3D, and that cannot be gamed with a simple 2D image. It is a whole different level of security. (Hristov, 2022)". Apple's Face ID uses these combinations of 30,000 points along with infrared light to map your mouth, eyes, skin tone, and other features of the face. For a short time, it seemed like Google wanted to improve its line of Pixel smartphones, including some improvements in the launch of the Pixel 4XL, but they were pretty much scrapped in the next generation for some reason.

Outside of hardware issues, facial recognition algorithms have problems. Although things have improved slightly, a US government study in 2019 showed that "facial recognition algorithms are far less accurate at identifying African American and Asian faces than Caucasian faces. (BBC, 2019)." Although this is entirely anecdotal, I remember that, around that time, there was a video floating around on YouTube of some Asian men testing facial unlock features on each other's phones. It didn't matter whose face they were using. They were all able to unlock each other's devices. Unfortunately, I can no longer find that video, but it came to mind when researching facial recognition technology, and it still sticks with me today.

The most common method to hack facial recognition technology is a presentation attack. These can come in two forms: static 2D and static 3D (Faresse, 2020). Static 2D spoofing involves using a photo or a video (like a video of someone's face). These methods are less sophisticated and do not work too well on iPhones anymore due to the combinations of parameters that go into authenticating an individual's face. Static 3D is more sophisticated and usually involves making masks or reproductions of someone's face. Marc Faresse discussed how to combat this in the same article and mentioned that blink-eye detection could be an effective defense. This would require a face scanner to observe how an individual blinks (like an average number). Another mechanism would require individuals to make specific faces (I like to call this the funny

face defense). If included as a parameter of facial recognition algorithms, both defense mechanisms could make it quite difficult for even individuals with access to mask-making. It is one thing to make a good replica of someone's face, but it is entirely different when you need also to replicate someone's smile or squints.

One final thing I would like to mention regarding facial recognition is the usage of an iris as a parameter. Although standalone iris scanners exist in the authentication world, they are not commonly used explicitly in mobile authentication as standalone authentication features for smartphones. I mentioned that support for facial recognition in Android is limited, especially compared to Apple, and even though Apple is highly invested in supporting Face ID, I am unsure if it is a direct parameter in its facial recognition authentication. However, I did learn that Optic ID is a feature on Apple's latest device, the Apple Vision Pro.

The Apple Vision Pro is one of Apple's latest products and is their attempt (an expensive one) to break into the AR/VR space. With their previous investment in Face ID, this makes much sense, especially since, by nature, the headset is right up against your eyes. According to Apple's support page for the Vision Pro, it uses a combination of a "high-performance eye-tracking system of LEDs and infrared cameras. (Apple Support, n.d.)" It makes much sense; outside of entering a PIN, which might be inconvenient to some users, iris recognition is probably the most feasible biometric authentication they can apply. Using their Optic ID, they say you can authorize purchases, make payments with Apple Pay, and sign into apps. According to them, any app that supports Face ID will also support Optic ID.

It is not the purpose of my exploration to debate or discuss the adoption of AR/VR tech from now and into the future. Still, as Apple has done with Touch ID, it could revolutionize how people authenticate in AR/VR. As mentioned, most people had no lock screen authentication enabled until Apple introduced fingerprint authentication. In the years that followed Touch ID, the number of people with lock screen authentication rose drastically. I speculate that AR/VR authentication might evolve similarly and would be willing to bet that most people currently do not have any authentication factor for other AR/VR devices. Regardless, I am excited to hear how people will try to break Optical ID and, in the process, force Apple to improve on this technology.

Looking Forward

I hope you enjoyed reading my musings on mobile authentication as much as I did researching the topic. For the final portion of this, I want to discuss some findings and thoughts that were harder to categorize but are undoubtedly worth discussion and further examination.

Artificial Intelligence and Behavioral Biometrics

The use of artificial intelligence in mobile authentication, especially in the context of behavioral biometrics, is on the horizon. AI's role in behavioral biometrics involves using a combination of machine learning algorithms to confirm the identity of individuals based on how they typically

use or handle their devices. Unlike other forms of authentication, behavioral biometrics will passively check your behavior and make the decision that you are who you are, the person who should be using a device. Aside from the way users actually utilize their mobile devices, behavioral biometrics can even incorporate sensors on the device, detecting the way a user "holds the phone, scrolls, toggles between fields, the pressure they use when they type and how they respond to different stimuli that are presented in online applications (Turgeman, 2018)."

There is more potential for the sensors to be used as a parameter for analyzing user behaviors than by how they navigate within the device itself. For example, a situation where someone left-handed tries to use your device, even though the device has been trained to know that you are right-handed. Another use case for sensors would be knowing the user's typical walk speed. I am a more heavy-set individual who typically walks slower and probably has a more unique rhythm than others. Theoretically, through behavioral biometrics that integrates sensors, my device, through its accelerometer and magnetometer, can understand that someone else using my device walks faster or slower in comparison to myself. Typing speed on a phone's keyboard could also be used as a parameter in analyzing a user's behavior. However, as it stands, none of these individual parameters alone should be enough to lock out a user from using their device. Referencing the example of walking speed or rhythm, what would happen if I sprained my ankle or God-forbid lost my foot in some accident? Suddenly, my regular behavior in walking speed would be entirely different from the behavior that I demonstrated just prior to that incident. That is why I believe that behavioral biometrics will most likely thrive more in a dynamic environment, providing more continuous authentication rather than as an all-or-nothing mechanism. There will also need to be necessary fallbacks, and potentially different rules will need to be put in place if some behavioral biometric algorithm determines that your behavior is suspicious. For example, maybe in some instances, it will still let you send texts or browse Facebook. Still, suddenly, it might decide that you need to provide another factor of authentication if you suddenly decide to open a banking application.

In the same vein of AI and mobile authentication, I am very excited about the release of the new Rabbit R1. If you are unfamiliar with the device, it is a fantastic little compact AI personal assistant. In a quote that I have seen in articles, the founder and CEO, Jesse Lyu, says, he "does not want to replace your smartphone. At least not right away. (The Verge, 2024)." These devices currently run for $199 and are currently on backorder. I initially wanted to do my entire semester-long deep dive on this device, and I ordered one back in January after they were revealed. Still, I learned that I most likely would not even receive my device until sometime this June.

What is interesting about these devices is that they run an operating system called Rabbit OS, which utilizes a large action model to act as a universal controller for all applications. It has a learning mode where the user can train the device to learn how the user interacts with their applications, accounts, and sites so that it can eventually automate tasks that you might typically need to do on your own, like ordering groceries, paying bills, or, according to the CEO of

the company, even purchasing a car. The idea is that instead of developing a bunch of APIs that interface with all these different apps, Rabbit R1 and its LAM-powered operating system will mimic human interaction and take these actions for you. Like Amazon Alexa and other home assistant devices, it works primarily through voice commands. So, if you want to order, say, some Uber Eats to your home, theoretically, you could tell it what you want to order and from where, and it will interface with the necessary application and make it happen. Furthermore, via extensive training, the device might learn that you typically order the same thing from some restaurant, and you could instead say, "Rabbit, order my usual from _____," and it should know what you mean. Although I am doubtful that it will replace mobile devices soon, I find the technology exciting.

The first thing that I had in mind, though, was how it would handle authentication. According to their website, the device will not store any user credentials of any third-party services, and all authentication happens on the destination services login systems. If users remain signed in on applications without the need for stringent authentication at every use, this could work well. Still, I do not really see this working at all in cases where users utilize SMS verification for login authentication. If everything is cloud-based, then how can they actually see your text message and verify authentication codes if they need to access an application? I did discuss it earlier, but many companies, like cellular service providers, are recommending that people utilize authentication applications in place of SMS-based authentication, specifically in the context of making yourself less vulnerable if you were to be a victim of SIM swapping. So, if this device ends up delivering on the promises it has made in its tech demos and video demonstrations of the device at work, maybe this will be a catalyst for pushing people in that direction.

Another primary concern that I have lies within the fact that this device will now seem like an obvious target for attacks. Based on my studies on the device, it will offer passive listening features, essentially waiting for the user to give a voice command, or it will rely on push to talk. The main issue with voice commands, though, is that voices can very easily be mimicked now with AI and with relative ease. A few years ago, the ordinary person might have had a more challenging time training an AI to mimic someone's voice, but now, there are so many tools available to people to train an AI to sound like someone else. This is creating a stir in the music industry, where people can make their music using the voice of their favorite artist without their consent. That is not to say that there are no companies dedicating time to fighting voice spoofing, at least in the context of voice authentication. A somewhat recent system called Void works by checking the frequencies of voice commands to identify the "liveliness" of voices. According to an article from CSIRO, when testing this on voice replay attacks, the system managed to identify 94 to 99% of all attacks. (Donnellan, 2020).

Overall, I am excited to explore the new Rabbit R1 and how it can be exploited. While I can appreciate someone trying to innovate and try something new in the world of personal assistants, I am still skeptical, especially from an authentication standpoint. In addition to their

work on Rabbit OS and developing its LAM, I would also like to see a significant investment in ensuring that the device is protected against voice spoofing.

Authentication In Brain-Computer Interfaces (BCIs)

All my research on mobile authentication had me thinking a lot about the future. We are a smart device-engrained society, and I doubt that will change anytime in the next few decades. All this research in mobile authentication had me thinking about how we will need to adapt to newer technologies being developed, like Neuralink. Admittedly, I am a sci-fi geek and enjoy some good cyber-punk genre media, but that is why the idea of using brain-computer interfaces in my lifetime simultaneously excites and terrifies me. What will the future of authentication look like in a world where computers and brains interact? In the same vein of categorization regarding something you are, something you know, and something you have, I want to speculate on what I call "something you think" or SYT.

Thought-specific patterns elicited by personalized mental tasks and how you are conscious of the world around you can potentially be applied to thought-based authentication mechanisms. Instead of security questions like "Where was your grandma born?" or "What was your first pet's name?" you might tap into a specific memory, like scenery from a vacation or something even more personal. Depending on the true capability of BCIs, they might be able to measure your brain's response to these memories and ensure that proof of thought is as unique and secure as the thoughts themselves. Unlike passwords or biometric data, which could be replicated or stolen, the unique way an individual's brain processes thoughts would be challenging to forge. Imagine a world where something you do to fall asleep, like counting sheep, can be used as a baseline for establishing some thought-based authentication for making purchases or even remotely unlocking the front door to your house.

All of this is equally terrifying and raises significant ethical and privacy implications that have never really been considered, and frankly, should be if this technology is ever used at the level we use our current mobile devices. Tapping into an individual's thoughts to grant or deny access raises many ethical questions, such as what it means to have privacy of thought. If our internal monologues and memories can become keys to our digital lives, how do we safeguard these keys from misuse or theft? The Luddite in me says to fight this at all costs, as merely creating an interface like this opens us up to the potential for someone to want to exploit or take advantage of it. In the extreme line of thinking, it begs a philosophical question akin to people criticizing Oppenheimer for building the first atomic bomb: Now that it exists, are we really in a safer world? Would it have mattered if it was Oppenheimer? Would it have come along eventually? In the same tone, if we fight Neuralink today, what is the next catalyst?

Additionally, what would regulatory frameworks even look like for technology like this? There must be provisions for how neural data can be collected, stored, utilized, and eventually destroyed. As a former legislative staffer, I can say with some authority that I am not very

confident that the United States government is competent enough to get ahead of the curve on policy related to technology of this magnitude. Only recently have our lawmakers taken an interest in the privacy issues and data concerns behind social media applications, and it is about 15 years too late.

On the brighter side of things, this technology provides unprecedented accessibility to people who were previously severely limited or unable to use personal computers or smart devices. I found it pretty wholesome that Nolan Arbaugh, the first person to receive Neuralink's BCI implant, stated the following to Neuralink scientist Bliss Chapman, "One of the first times you all gave me complete control over this [Neuralink tech], I stayed up until [...] like 6 am playing Civilization 6." (Wood, 2024). Currently, Neuralink's main functionality is essentially just being able to control a cursor and clicks, but with your mind. For those who cannot use their hands, like people with quadriplegia or those in the late stages of ALS, this technology can be life-changing.

According to an IGN article, "The implant has also helped Noland to begin learning French and Japanese and has generally removed barriers when it comes to reading. He has also emphasized that he had suffered no "cognitive impairments" because of the procedure and that he was able to leave the hospital a day after undergoing the two-hour surgical procedure. (Wood, 2024)." Unfortunately, I was unable to find any concrete information or studies regarding the general population's acceptance of devices like these. Still, I speculate that many people with severe disabilities will be eager to get ahold of such technology, especially after Noland's so-far successful trial.

BCIs, especially in the context of authentication and in general, are equally exciting and terrifying. With the addition of new players into the industry, like Steam's Gabe Newell, who founded his BCI company Starfish, and Meta, which is exploring BCI human interfaces, the competition to develop the best products is slowly but surely beginning to ramp up. After the recent success of Neuralink's first human clinical trial, I only imagine things will continue to get more exciting in the coming months and years.

SecureTouch: A Fingerprint and PIN-Based Authenticator Application

Repository: https://bit.ly/securetouch-repo
Web Interface: https://bit.ly/securetouch-web-interface
Android APK: https://bit.ly/securetouch-apk
Presentation: https://bit.ly/mobile-auth-presentation

Summarization

SecureTouch is a prototype mobile authenticator application built for Android. It is designed to provide a secure alternative mechanism for web application sign-ins. A primary web interface created with React and Firebase allows users to register and log in, forming the basis for testing the SecureTouch application. The prototype's primary objective was to incorporate the elements of common authentication factors: something you know, have, and are in varying phases throughout account registration, the mobile application setup, and subsequent login usage.

Registering Your Authenticator Application Against Your Account:

- Something You Know: In the SecureTouch Android application, a user initiates the setup process by entering their login credentials - email and password associated with their account. This forms the first factor of authentication, relying on knowledge-based credentials.
- Something You Have: After successfully authenticating with email and password, the user is prompted to enter the 12 backup phrases. These phrases are generated and provided after registering on the web interface that I built. They become 'something you have' due to the necessity for safekeeping (even if users are just storing it somewhere on their computer).

Authentication Mechanisms for Future Use:

- Something You Are: The setup continues with users registering their fingerprints on the Android device. Stored only on the physical device, the fingerprint and PIN are intended to be used as a final layer of security in the regular authentication flow.
- Something You Know: The process concludes with the user establishing a 6-digit PIN. This knowledge-based credential offers a quick yet secure access layer, ensuring the app remains inaccessible to unauthorized users without this PIN, even if the device is compromised.

Now, each time a user attempts to log into the web application, following their email and password entry, SecureTouch prompts them to authenticate through two factors via the Android app: the PIN and their fingerprint. Successful authentication with both allows the user to access the web application. If users encounter future issues with the application or their device is lost or stolen, they can reinitiate the setup process from the beginning.

Strengths:

- The Android app SecureTouch integrates fingerprint and PIN authentication, offering two different categories of authentication for handling web logins, without even considering the fact that the user also needs to provide their web login information (email and password).
- PIN and fingerprint data is stored locally and securely on the device only via “Trusty” Android’s trusted execution environment.
- PIN and Fingerprint authentication provide quick authentication for web logins, providing an optimal trade-off between security and convenience.
- Unlike most web applications, which might typically use SMS or email-based 2FA, my application solves many problems related to authentication reliant on network and cellular services, like SIM swapping, email hacking, and interception.

Weaknesses:

- The effectiveness of fingerprint authentication is still dependent on the quality of hardware in a user’s device. Unlike Apple products, the sensors in devices that use the Android OS can vary greatly.
- Even though the biometric data is only stored locally on the user’s device, there might still be some reluctance by a user to provide their fingerprint data due to privacy concerns.
- Currently, there is no way to reinitiate the setup process for your authenticator app other than uninstalling and reinstalling the application. This means that if you forget your PIN or your fingerprint gets corrupted, you will need to uninstall and reinstall the app.
- If you lose or forget your backup phrases, there is no other way to set up the authenticator application.
- The application does not follow OAuth 2.0 protocols, but this is primarily because it is specific to a specific web application's sign-in capability.

Future Improvements:

- Explore implementing this application as a third-party authenticator application like Microsoft or Google’s authenticator applications. This would require making sure that the application follows OAuth 2.0 protocols, which is an industry-standard for third-party authentication applications.
- Explore developing this for Apple devices and, in the process, maybe provide additional biometric options like facial recognition. Apple’s Face ID provides much better facial recognition compared to devices that use Android OS.
- Provide additional recovery options for instances where a user loses their backup-phrases.
- Store the hashes and salts of a user’s security phrases in separate databases. Currently, the hashes of their backup phrases and accompanying salt are stored in the same Firestore user document.

Conclusion

Throughout my exploration of mobile authentication and, ultimately, the presentation of my findings, I have primarily scrutinized the varying forms of mobile authentication - categorized by "something you know," "something you have," and "something you are." Each offers its distinct challenges in mobile authentication. I believe it is most important to identify them by their weaknesses, though, since through that line of thinking, we can come up with the best use cases where they might find success where others do not.

In my examination of "something you know" based mobile authentication - specifically passwords, PINs, and patterns – I discussed some of their primary vulnerabilities. These authentication methods are still widely used as primary forms of authentication for lock screens and even other applications. Despite their weaknesses as single forms of authentication, I do not foresee them being phased out or replaced anytime soon, even if biometric authentication methods are improving. As much as experts express concerns over the weaknesses of password-based authentication, they remain one of the more robust options in terms of knowledge-based authentication methods in lock screens and for accessing applications. Overall, though, we should try to improve the security of our knowledge-based authentication mechanisms by combining them with other categories of authentication and using “strong passwords” instead of weak and reused ones.

In the context of "something you have" mobile authentication, our devices themselves play a critical role in our security. As learned, our devices themselves come with risks regarding mobile authentication. I discussed a wide range of concerns like loss, theft, phishing, SIM swapping, and social engineering. Through SIM swapping, an attacker can port a phone number over to the control of an attacker, enabling them to intercept SMS authentication. In a similar vein, social engineering can be used to manipulate people into revealing sensitive information, opening us up to attacks (like SIM swapping) and even our friends and family. Lost or stolen devices may render app-based 2FA moot if the device itself acts as a primary means of authentication. Especially if you have no lock screen, a bad actor who possesses our device can provide them with many details about our digital lives, making our devices an attractive target for attackers. Finally, despite efforts by app stores like Google Play and third-party app stores to remove malicious apps, the risk of downloading an app with excessive permissions beyond what is needed for the functionality of the app is a significant concern.

Using our own unique personal characteristics, like our fingerprints, face, and iris, biometrics offer a great alternative to “something you know” and “something you have” authentication methods. Our physical traits are far less susceptible to the common exploits that affect knowledge-based authentication. Since their introduction to smartphones with the iPhone 5, fingerprint-based authentication has led to a drastic

increase in people securing their devices through biometrics and even PINs, patterns, or passwords. More modern devices now combine multiple sensors to enhance security and reduce the potential for spoofing, increasingly safeguarding biometric authentication. Facial recognition technology, while less secure than fingerprint authentication due to sensor inconsistencies, has been shown to be more susceptible to spoofing. Android devices, in particular, face challenges due to inconsistent sensor quality across all the different models and manufacturers. Apple, however, has continued to work on improving its facial recognition technology through its Face ID, which creates more detailed 3D maps of a user's face, offering much greater security than less complex 2D mappings that can be fooled with something as basic as a photo. As we have discussed, privacy concerns and leaking of biometric data is another primary concern. Since we cannot change our biometric data the same way we can change knowledge-based or possession-based authentication mechanisms, we should take more significant steps to secure biometric data. Especially with the advent of AI and machine learning, where we can train models to mimic people's voices, I do not believe voice recognition has much of a future in biometrics due to the inherent public nature of our voices.

Looking forward, there are also some emerging areas in mobile authentication to be excited about. I explored, in brief, the emerging role of AI and the potential for behavioral biometrics in mobile authentication. The introduction of new devices like the Rabbit R1 underscores the need for new forms of behavioral and voice-based authentication. Even further into the future, brain-computer interfaces like Neuralink create a need for using neural activity for authentication, which also comes with its own privacy and ethics concerns.

I had much fun developing my mobile authenticator application, SecureTouch. I have grown to appreciate fingerprint authentication and felt that designing my application to leverage the use of a biometric (fingerprint) and knowledge-based layer (PIN) would provide an outstanding balance between security and convenience for a user. I was most interested in the fact that on Android devices, both of these forms of authentication can be stored safely, offline, and directly on the device. If someone manages to get your PIN, you still have to provide a print for a successful login, and if someone manages to forge your print, someone still has to get ahold of your PIN before they can provide the forged print. As someone who enjoys mobile application development, I found this project to be a great learning experience and an opportunity to learn more about how Android provides developers with the tools to develop applications like my prototype SecureTouch. Overall, I have become a much better developer while simultaneously bolstering my awareness of security as I continue to build apps in the future.

Bibliography

Apple Explained. (2022, September 5). *Why iPhones Do not Have Pattern Unlock*. Retrieved from YouTube: https://youtu.be/OPZMNtAW4MM?si=gz7sJQH9loVxn-Zi

Apple Support. (n.d.). *About Optic ID advanced technology*. Retrieved from support.apple.com: https://support.apple.com/en-us/118483

Aratek. (2023, January 1). *The Fingerprint File: 4 Fingerprint Sensor Types*. Retrieved from aratek.com: https://www.aratek.co/news/the-4-fingerprint-sensor-types

Aviv, A. J., Davin, J. T., Wolf, F., & Kuber, R. (2017). Towards Baselines for Shoulder Surfing on Mobile Authentication. *ACSAC '17: Proceedings of the 33rd Annual Computer Security Applications Conference* (pp. Pages 486–498). Association for Computing Machinery.

Aviv, A. J., Gibson, K., Mossop, E., Blaze, M., & Smith, J. M. (2010). Smudge Attacks on Smartphone Touch Screens. *4th USENIX Conference on Offensive Technologies*. USENIX Association.

BBC. (2019, December 20). *Facial recognition fails on race, government study says*. Retrieved from bbc.com: https://www.bbc.com/news/technology-50865437

Boonkrong, S. (2020). *Authentication and Access Control: Practical Cryptography Methods and Tools*. Apress.

Bursztein, E. (2014, March). *Survey: most people do not lock their android phones - but should*. Retrieved from elie.net: https://elie.net/blog/survey-most-people-dont-lock-their-android-phones-but-should

CPI OpenFox. (2023, July 17). *History of Fingerprinting*. Retrieved from openfox.com: https://www.openfox.com/history-of-fingerprinting/

Crumb. (2023, March 19). *The Honor Student Caught Stealing $7.5 Million Dollars*. Retrieved from YouTube.com: https://youtu.be/A-nIjUIFARA?si=fOKJw9fRfussZM6T

Donnellan, A. (2020, June 24). *How you can a-Void a voice spoofing attack*. Retrieved from csiro.au: https://www.csiro.au/en/news/all/articles/2020/june/how-you-can-a-void-a-voice-spoofing-attack

Fanti, M. (2023). *Implementing Multifactor Authentication*. Packt.

Faresse, M. (2020, May 21). *The Most Common Facial Recognition Spoofing Methods and How to Prevent Them*. Retrieved from dormakaba.com: https://blog.dormakaba.com/the-most-common-facial-recognition-spoofing-methods-and-how-to-prevent-them/

Franceschi-Bicchierai, L. (2019, February 1). *Hacker Who Stole $5 Million By SIM Swapping Gets 10 Years in Prison*. Retrieved from vice.com: https://www.vice.com/en/article/gyaqnb/hacker-joel-ortiz-sim-swapping-10-years-in-prison

Glover, J. D., Sudderick, Z. R., Bo-Ju Shih, B., Bath-Samblas, C., Charlton, L., Krause, A. L., . . . Headon, D. J. (2023). The Development Basis of Fingerprint Pattern Formation and Variation. *50cell*, Volume 185, ISSUE 5.

Grimes, R. A. (2020). *Hacking Multifactor Authentication*. Wiley.

Han, Q., Mandujano, S., Ports, S., Subrahmanian, V., & Tetali, S. D. (2023). *The Android Malware Handbook*.

Hristov, V. (2022, August 16). *Android has written off Face ID way too soon*. Retrieved from phonearena.com: https://www.phonearena.com/news/Android-has-written-off-Face-ID-way-too-soon_id141933

HYPR Team. (2022, April 28). *Six of the Biggest Problems with Password Managers*. Retrieved from securityboulevard.com: https://securityboulevard.com/2022/04/six-of-the-biggest-problems-with-password-managers/

IANS. (2023, May 8). *New Android malware discovered that steals your passwords, 2FA codes*. Retrieved from business-standard.com: https://www.business-

standard.com/technology/tech-news/new-android-malware-discovered-that-steals-your-passwords-2fa-codes-123050800681_1.html

Kaspersky. (n.d.). *What is SIM Swapping?* Retrieved from Kaspersky.com: https://www.kaspersky.com/resource-center/threats/sim-swapping

Schleier, S., Holguera, C., Mueller, B., & Willemsen, J. (2023). *OWASP Mobile Application Security.* Retrieved from https://mas.owasp.org/: https://mas.owasp.org/MASTG/

Smith, Z. S. (2022, April 22). *Google Reportedly Bans Dozens Of Apps Containing Spyware*. Retrieved from forbes.com: https://www.forbes.com/sites/zacharysmith/2022/04/06/google-reportedly-bans-dozens-of-apps-containing-spyware/?sh=94ed17d26578

The Verge. (2024, January 9). *The Rabbit R1 is an AI-powered gadget that can use your apps for you*. Retrieved from theverge.com: https://www.theverge.com/2024/1/9/24030667/rabbit-r1-ai-action-model-price-release-date

Turgeman, A. (2018, January 18). *Machine Learning And Behavioral Biometrics: A Match Made In Heaven*. Retrieved from forbes.com: https://www.forbes.com/sites/forbestechcouncil/2018/01/18/machine-learning-and-behavioral-biometrics-a-match-made-in-heaven/?sh=283b277e3306

Weatherbed, J. (2023, September 12). *10 years ago. Apple finally convinced us to lock our phones*. Retrieved from theverge.com: https://www.theverge.com/23868464/apple-iphone-touch-id-fingerprint-security-ten-year-anniversary

Weinert, A. (2019, July 9). *Your Pa$$word doesn't matter* . Retrieved from techcommunity.microsoft.com: https://techcommunity.microsoft.com/t5/microsoft-entra-blog/your-pa-word-doesn-t-matter/ba-p/731984

Wood, A. (2024, March 29). *First Human Patient to Receive a Neuralink Brain Implant Used it to Stay Up All Night Playing Civilization 6*. Retrieved from ign.com: https://www.ign.com/articles/first-human-patient-to-receive-a-neuralink-brain-implant-used-it-to-stay-up-all-night-playing-civilization-6

Yirka, B. (2023, February 10). *How Fingerprints Get Their Unqiue Whorls*. Retrieved from phys.org: https://phys.org/news/2023-02-fingerprints-unique-whorls.html#:~:text=The%20random%20placement%20of%20the,to%20those%20of%20hair%20follicles.

Zhang, F., Kondoro, A., & Muftic, S. (2012). Location-based Authentication and Authorization Using Smart Phones. *11th International Conference on Trust, Security and Privacy in Computing and Communications* (pp. pp. 1285-1292). Liverpool: IEEE.