Improving Security Awareness of QR Code Phishing

Zur Erlangung des akademischen Grades eines

Doktors der Ingenieurwissenschaften

(Dr.-Ing.)

von der KIT-Fakultät für
Wirtschaftswissenschaften
des Karlsruher Instituts für Technologie (KIT)

genehmigte

DISSERTATION

von

M.Sc. M.A. Mattia Mossano

Tag der mündlichen Prüfung:	12.11.2026
Hauptreferentin:	Prof. Dr. Melanie Volkamer
Korreferent:	Prof. Dr. Filipo Sharevski

Karlsruhe 12.11.2026

Abstract

Quick Response codes have become an integral part of everyday life and nowadays are routinely used for advertising, payments, authentication, ticketing, restaurant menus, and many other activities. Their widespread adoption has considerably simplified the interaction between users and digital services, while simultaneously creating new opportunities for attackers. One such opportunity is *QRishing*, i.e., phishing through QR codes. In QRishing attacks, attackers exploit QR codes to direct users toward malicious content or actions while concealing the underlying destination behind a machine-readable representation. Despite the increasing prevalence of QRishing attacks, comparatively little research has investigated how users interact with QR codes, how QRishing awareness should be communicated, and how awareness interventions addressing QRishing should be systematically developed. This thesis contributes to addressing these challenges through four distinct contributions.

The first contribution of this thesis is understanding how users interact with QR codes in real-world settings and identifying behaviors that may expose them to QRishing attacks. To this end, a naturalistic study was conducted in the metropolitan area of Chicago, United States, where phishing QR code pretexts were deployed in public locations and participants (those who scanned the QR code on their own volition) were subsequently interviewed regarding their interactions. The results show that users frequently rely on non-security-related characteristics when deciding whether to scan QR codes and often either fail to inspect the destination URL or rely on security indicators that can easily be manipulated by attackers. These findings highlight the need for dedicated QRishing awareness interventions.

The second contribution of this thesis addresses a broader limitation in the development of security awareness interventions, regardless of the domain (e.g., passwords, phishing, QRishing, etc.): the lack of systematic methodologies for their development. To address this challenge, a framework for developing security awareness interventions is proposed. The framework is developed based on an analysis of existing frameworks and their limitations, and supports developers in identifying several key aspects of a security awareness intervention (e.g., the target population, the awareness goal, the attack techniques, etc.). Unlike previous proposals, the framework is not only designed to be applicable across different security contexts and domains, but it also explicitly shows the relations between the various aspects considered and how they influence one another.

The third contribution of this thesis addresses the lack of structured knowledge required to systematically develop a QRishing awareness intervention. To this end, the thesis presents a taxonomy of QRishing attack techniques and a systematic review of misconceptions about QRishing attack techniques and mitigation strategies. Together, these contributions expand the current understanding of QRishing and provide part of the input required for the systematic development of QRishing awareness interventions.

The fourth contribution of this thesis is the development and the evaluation of an effective QRishing awareness intervention. The proposed framework is applied to develop *QRNope*, a QRishing awareness intervention aimed at lay users in the United States using the default iOS Camera application. QRNope is refined through feedback from both experts and members of the target population before being evaluated in an online between-subjects study involving 240 participants from the United States. The results show that QRNope significantly improves the participants' ability to distinguish between legitimate QR codes and phishing ones compared with both an existing QRishing awareness intervention published by the US Federal Trade Commission and a placebo text about QR codes. Furthermore, the evaluation identifies opportunities for further refinement while highlighting the importance of complementing awareness interventions with interfaces that better support users in making security-relevant decisions.

By providing the aforementioned contributions, this thesis advances the understanding of QRishing from both a security- and a human-centered perspective. It provides empirical results regarding how users interact with QR codes, proposes a framework for developing security awareness interventions, supplies the knowledge required to apply this framework within the QRishing domain, and demonstrates its successful application through the development and evaluation of an effective QRishing awareness intervention. Together, the methods, results, and insights presented throughout this thesis advance the state of the art in QRishing research while providing practical benefits for researchers, practitioners, and end users.

Contents

Abstract i

Definitions and Terminology Used in this Thesis vii

1 Introduction 1
1.1 Scope and Structure of this Work 2
1.2 Main Contributions 4
1.3 Related Publications 5

2 QRishing Awareness Among QR Code Users 7
2.1 Related Work 8
2.2 Methodology 11
2.2.1 Research Questions 12
2.2.2 Study Design 12
2.2.3 Pilot Study 18
2.2.4 Ethical Considerations 19
2.2.5 Data Analysis Process 19
2.2.6 Limitations 20
2.3 Results 21
2.3.1 Demographics 21
2.3.2 RQ1 - Pretext Persuasion Factors 21
2.3.3 RQ2 - Actions Around QR Codes 22
2.4 Discussion and Conclusion 25

3 Framework for Developing Awareness Interventions 27
3.1 Related Work 28
3.2 Version 1.0 of the Framework for Developing Awareness Interventions 33
3.2.1 Step 1 – Preparation 34
3.2.2 Step 2 – Attack techniques in general 35
3.2.3 Step 3 – Mitigation strategies in general 37
3.2.4 Step 4 – Attack techniques mitigation and misconceptions to be covered 38
3.2.5 Step 5 – Content creation based on target population 39

3.2.6 Step 6 – Design of the intervention . . . 40
3.2.7 Step 7 – Evaluation . . . 41
3.3 Comparison of Framework 1.0 with Previous Frameworks . . . 42
3.4 Version 2.0 of the Framework for Developing Awareness Interventions . . . 45
3.4.1 Step 1 – Preparation . . . 46
3.4.2 Step 2 – Attack techniques to consider . . . 48
3.4.3 Step 3 – Attack techniques and mitigation strategies to consider . . . 49
3.4.4 Step 4 – Attack techniques, mitigation strategies, and misconceptions to cover . . . 49
3.4.5 Step 5 – Content creation . . . 50
3.4.6 Step 6 – Intervention Implementation . . . 50
3.4.7 Step 7 – Evaluation . . . 51
3.5 Conclusion . . . 51

4 Inputs Required by Framework 2.0: *All Attack Techniques in the Domain*, *Misconceptions on Attack Techniques*, and *Misconceptions on Mitigation Strategies* . . . 53
4.1 QR Code Attack Techniques in the iOS Environment . . . 55
4.1.1 Related Work . . . 55
4.1.2 Methodology . . . 58
4.1.3 Results . . . 62
4.1.4 QRNope Input: *All Attack Techniques in the Domain* . . . 67
4.1.5 Limitations . . . 72
4.2 Misconceptions of Users about QRishing . . . 73
4.2.1 Related Work . . . 74
4.2.2 Methodology . . . 76
4.2.3 Limitations . . . 78
4.2.4 Results . . . 78
4.2.5 QRNope Input: *Misconceptions on Attack Techniques* and *Misconceptions on Mitigation Strategies* . . . 85
4.3 Conclusion . . . 87

5 Development of *QRNope*, a QRishing Awareness Intervention . . . 89
5.1 Related Work . . . 90
5.2 Development of *QRNope* . . . 92
5.2.1 Step 1 – Preparation . . . 92
5.2.2 Step 2 – Attack techniques to consider . . . 99
5.2.3 Step 3 – Attack techniques and mitigation strategies to consider . . . 104
5.2.4 Step 4 – Attack techniques, mitigation strategies, and misconceptions to cover . . . 113
5.2.5 Step 5 – Content creation . . . 116

5.2.6 Step 6 – Intervention implementation . . . 124
5.2.7 Step 7 – Empirical Evaluation (Feedback stages only) . . . 127
5.3 Conclusion . . . 136

6 Evaluation of *QRNope* . . . 137
6.1 Related Work . . . 138
6.2 Methodology . . . 139
6.2.1 Research Questions . . . 139
6.2.2 Study Groups . . . 145
6.2.3 Study Design . . . 145
6.2.4 Pretexts Used in the Study . . . 147
6.2.5 Recruitment and Ethical Considerations . . . 151
6.2.6 Pilot Studies . . . 152
6.2.7 Data Analysis Process . . . 152
6.2.8 Limitations . . . 154
6.3 Results . . . 155
6.3.1 Demographics . . . 156
6.3.2 Results for the Attacks (RQ1, RQ2) . . . 158
6.3.3 Results for the Misconceptions (RQ3, RQ4) . . . 163
6.4 Discussion . . . 164

7 Conclusion . . . 169
7.1 Thesis Summary . . . 169
7.1.1 QRishing Awareness Among QR Code Users . . . 169
7.1.2 Framework for Developing Awareness Interventions . . . 170
7.1.3 Inputs Required by Framework 2.0: *All Attack Techniques in the Domain*, *Misconceptions on Attack Techniques*, and *Misconceptions on Mitigation Strategies* . . . 171
7.1.4 Development of *QRNope*, a QRishing Awareness Intervention . . . 172
7.1.5 Evaluation of *QRNope* . . . 173
7.2 Future Research . . . 175
7.3 Final Remarks . . . 176

A Appendix . . . 179
A.1 Naturalistic Study – Verbal Consent Script . . . 179
A.2 Naturalistic Study – Interview Script . . . 179
A.3 Naturalistic Study – Codebook . . . 180
A.4 QRNope Evaluation - Instructions to the Participants . . . 182
A.5 QRNope Evaluation - Demographics Tables . . . 185

List of Figures . . . 187

List of Tables **189**

List of Publications **191**

Bibliography **195**

Definitions and Terminology Used in this Thesis

Informational Material, Awareness Intervention, Awareness Measure

Informational Material. A content that, when processed by an individual, informs them about a specific topic. Informational material might induce a behavioral change, but that is not its primary goal. Neither the effect occurrence, nor the direction of the behavioral change can be reliably predicted.

Awareness Intervention. An action, a tool, or a strategy that, when processed by an individual, informs them about a specific topic, and consistently induces a measurable change in that individual's behavior in a predictable direction. In this thesis, awareness interventions are always intended as *security* awareness interventions.

Awareness Measure. A tool or a method used to assess the awareness of an individual about a specific topic. Often used to determine the effectiveness of an awareness intervention.

Informational sentence, Mitigation Strategy, Addressing an attack, Security assumption

Informational Sentence. An informational sentence is a set of words that informs a reader of the existence of a certain attack technique, but does not provide a consistent way to detect and avoid said attack technique.

Mitigation Strategy. A mitigation strategy is an action, a set of actions, or a practice that, when applied, allows a user that applies it to consistently detect and avoid a specific attack technique.

Addressing an attack. An awareness intervention is only considered addressing a certain attack technique if the awareness intervention provides a mitigation strategy that allow the reader to consistently detect and avoid said attack technique.

For example, given the attack technique “Use of short URLs to obfuscate a phishing URL”, if the awareness intervention recommends to “Use expansion services to see the destination URL”, then the awareness intervention addresses the attack technique. Instead, if for the same attack technique the awareness intervention only states to “Beware that short URLs are used for phishing attacks”, without further explanations on how to avoid the attack technique, then the awareness intervention does *not* address the attack technique.

Security assumption. In this thesis, a security assumption is a statement clarifying that a certain attack technique, mitigation strategy, or security property of a software or system does not need to be addressed by the awareness intervention because it is already addressed by an external mechanism or by an external actor. For example, a security assumption can be formulated to state that the attack technique “SQL Injection” is not covered by the awareness intervention because it is addressed by system administrators.

User, Lay User, Reader, Target Population Member

User. An individual that operates a specific technology, e.g., an individual that scans QR codes.

Lay User. A user with no IT background (i.e., has not majored in, nor worked in IT) that operates a certain technology, e.g., a user that scans QR codes and that has no IT background.

Reader. A user, lay or otherwise, that reads an awareness intervention.

Target Population Member. The intended reader of an awareness intervention, selected based on criteria set by the developers of the awareness intervention. Two awareness interventions can have the same intended readers, two readers might not both be members of the target population (if one of the readers is not fulfilling the designers‘ criteria), and a member of the target population might not be a reader.

Context, Pretext

Context. Everything related to the action of using a certain technology. For example, if the action is „scanning a QR code“, the context covers everything related to said action, from the specific technology used (e.g., which device), to the situation around the scan (e.g., at work, at a bus station).

Pretext. The settings surrounding a QR code: where it is used (e.g., office, bus stop, email), and if there is a message surrounding it or if it is a standalone QR code (e.g., poster, sticker, ads).

1 Introduction

Phishing is a type of social engineering attack that dates back to the early days of the Internet (as reported in Rekouche (2011)). Its goal is to trick victims into actions beneficial to the attackers, such as revealing sensitive information (e.g., credentials, as described in NIST (2026a)), installing malware (e.g., ransomware, as NIST (2021) explains), or sending money, among others. The threat of phishing remains significant and continues to increase. Federal Bureau of Investigation (2026) reported that phishing was the most frequently reported cybercrime category in 2025, with 191,561 complaints. Furthermore, Anti-Phishing Working Group (2026) reported a 13.8% increase in phishing attacks from Q4 2025 to Q1 2026. Researchers in both academia and industry continuously advance novel and updated solutions to address phishing attacks, but attackers are also continuously improving their attack vectors (as shown in El Aassal and Verma (2019)). One such novel vector that is increasingly used for phishing purposes is employing *Quick Response (QR) codes* (as shown, e.g., in Federal Bureau of Investigation (2025)).

As explained in Shin et al. (2012), QR codes are two dimensional bar codes that offer a straightforward way to deliver several types of information without requiring the use of complex textual strings, such as product identifiers, tracking numbers, or *Universal Resource Locators* (hereafter, *URLs*). Originally invented in 1994 by Masahiko Hara (as per Denso Wave Incorporated (2025)), the first work to investigate the use of QR codes as phishing vectors was Vidas et al. (2013), who also coined the name of this practice: *QRishing*.

Despite the convenience of QR codes as information delivery vectors, they did not achieve widespread adoption for several years, as they were primarily seen as useful for potential future technologies, but with no practical application at the time (as explained in, e.g., Kan et al. (2011)). Initially aimed at the manufacturing industry, QR codes became user-approachable in 2017, when both iOS 11 (as shown in Apple (2026)) and Android 8 (as shown in Hill (2021)) integrated a QR code scanner in their camera app. However, the real push toward the normalization of QR codes in everyday life came when the COVID-19 pandemic caused an unforeseen necessity for mediums that allowed for the contactless exchange of information, and QR codes became the go-to method for information transfer (as explained in Ricson (2026)). The widespread adoption of QR codes opened up a new opportunity for attackers, and what was initially a relatively niche attack surface became a way to reach potential victims across a variety of everyday situations –

for example, phishing QR codes were discovered on parking meters throughout San Antonio in the United States (hereafter, *US*; reported in Barr (2022)), or a backdoor in an open-source QR code generator was found to embed phishing URLs into generated QR codes (as shown in Murphy (2021)).

The emergence of QRishing is problematic because this vector differs from more traditional phishing vectors (e.g., SMS in so-called *Smishing*, as named in Mishra and Soni (2019)) in ways that affect awareness interventions. Previous work found that publicly available anti-phishing webpages often focus on traditional phishing vectors, especially email phishing, while other phishing vectors, such as Smishing or QRishing, receive less attention (as shown in Mossano et al. (2020)). Furthermore, it is common for enterprise training to focus on phishing email campaigns, and disregard other vectors (as shown in Ho et al. (2025)). As such, users looking for guidance on how to recognize QRishing attacks may encounter recommendations developed primarily for traditional phishing vectors that may not always transfer directly to QR code interactions (e.g., hovering over a link to inspect its destination).

The applicability of traditional anti-phishing recommendations is further complicated by the different decision-making circumstances created by QR code interactions. For example, in email phishing there are several proposals for support tools that help users detect phishing attacks (e.g., Berens et al. (2024b)), and email clients have automated filters and defenses to detect phishing emails (e.g., Safelink, as explained in Microsoft (2026), or Google warnings, as explained in Google (2026)). In QR code scanners, however, this is not the case, and default QR code scanners do not provide filters or warnings to protect against QRishing. As a result, users may have to base their decision on fewer surrounding cues and instead rely on their ability to identify signs of deception in the content displayed after scanning.

These differences suggest that QRishing presents challenges that are not fully addressed by traditional anti-phishing approaches and therefore warrant dedicated investigation. The remainder of this introductory chapter is organized as follows: Section 1.1 describes the scope and structure of the thesis, and Section 1.2 summarizes its main contributions. Lastly, Section 1.3 presents the publications produced during the doctoral research that form the basis of the individual chapters.

1.1 Scope and Structure of this Work

The scope of this thesis is the development and evaluation of an awareness intervention about QRishing. The thesis follows a progression that begins with understanding how users interact with QR codes (Chapter 2), continues by developing a framework for developing awareness

interventions (Chapter 3), and culminates in the development of a QRishing-specific awareness intervention, called *QRNope* (Chapter 4 and Chapter 5) and its evaluation (Chapter 6).

Chapter 2 investigates how users interact with QR codes in naturalistic settings, i.e., when encountering QR codes naturally in their everyday life. Despite the widespread adoption of QR codes across numerous everyday contexts, previous studies (Vidas et al. (2013), Krombholz et al. (2015), and Bekavac et al. (2024)) mainly observed whether users scanned QR codes, but provided limited insight into the reasons behind users' actions when interacting with QR codes. The chapter addresses this gap through a naturalistic study examining how participants interact with QR codes encountered outside of a laboratory setting and by interviewing them after the interaction to understand the reasons behind their actions. The findings provide empirical evidence regarding users' behavior and establish an initial understanding of the challenges associated with QR code interactions. These results also motivate the need for QRishing-specific awareness interventions by demonstrating the lack of user awareness when it comes to QRishing attack techniques and mitigation strategies.

Chapter 3 focuses on the systematic development of awareness interventions. While Chapter 2 showed the need for awareness interventions, previous research (e.g., Mossano et al. (2020) and Redmiles et al. (2020)) has shown that awareness interventions often rely on recommendations that are outdated, inaccurate, and presented in a confusing and non-standardized manner. To address this issue, this chapter first synthesizes (and expands) several previous frameworks for developing awareness interventions (Ghazvini and Shukur (2017), Jaeger (2018), Mayer (2019), Khader et al. (2021), Khando et al. (2021), Dialog für Cyber-Sicherheit (2022), and ENISA (2024c)) into the first version of a framework for developing awareness interventions (Framework 1.0). The application of Framework 1.0 then highlights several limitations and opportunities for refinement, which are addressed through the development of Framework 2.0. Framework 2.0 serves as the methodological foundation for the remainder of the thesis and is subsequently applied to the QRishing domain.

Chapter 4 presents several systematic reviews that contribute to structuring the existing knowledge on QRishing attacks and users' misconceptions about both phishing and QRishing. At the same time, these contributions provide part of the information required for the development of QRNope according to the framework presented in Chapter 3. More specifically, they correspond to the framework's stages of *All attack techniques in the domain*, *Misconceptions on attack techniques*, and *Misconceptions on mitigation strategies*. While these systematic reviews could have been presented as part of Chapter 5, doing so would have substantially increased its length and interrupted the description of the intervention-development process. As such, they are discussed separately to improve readability and offer a coherent argument presentation.

Chapter 5 combines the insights from the preceding chapters and applies Framework 2.0 to develop QRNope, a QRishing-awareness intervention. The chapter documents the complete development process, including the identification of the target population and the awareness goal, the selection and justification of the awareness intervention's content, the implementation of the intervention itself, and the refinement of QRNope through feedback loops involving experts and members of the target population. Particular attention is given to the documentation of the security assumptions formulated during QRNope development, in order to document precisely what the awareness intervention addresses and what is instead assumed to be addressed externally (e.g., by developers or QR code scanner). The goal of the chapter is two-fold. First, the chapter demonstrates how Framework 2.0 can be applied in practice to develop and refine an awareness intervention. Second, the chapter develops an awareness intervention that addresses the needs highlighted in Chapter 2.

Chapter 6 evaluates the effectiveness of QRNope through an online between-subjects study with 240 participants from the US who use iOS devices (i.e., the target population of QRNope). The study investigates whether exposure to QRNope improves the participants' ability to correctly identify and respond to QRishing attacks and compares its effectiveness with that of both an alternative awareness intervention (i.e., the one accessible at FTC (2023)) and a control group with a placebo text (i.e., a text about QR codes with no security-relevant information). The results provide evidence regarding the effectiveness of QRNope and allow assessing the practical utility of the framework-guided development process presented in Chapter 5.

Lastly, Chapter 7 summarizes the findings of the thesis, outlines promising directions for future research on QRishing and security-awareness interventions, and presents some closing remarks.

1.2 Main Contributions

The main contributions of this thesis are:

- A naturalistic study on the behavior of users when interacting with QR codes and an assessment of whether users require additional support to detect QRishing attacks.
- A framework for developing security awareness interventions.
- A taxonomy of QRishing attack techniques from the literature, extended with attack techniques from the MITRE ATT&CK® database targeting iOS devices.
- A systematic identification of user misconceptions about QRishing attack techniques and about their corresponding mitigation strategies.

- The development of QRNope, a QRishing awareness intervention, through the application of the proposed framework.
- An empirical evaluation of QRNope through a controlled user study.

1.3 Related Publications

The research presented in this thesis is based on and extends several previous publications.

Chapter 2 is partly based on the publication:

> F. Sharevski, **M. Mossano**, M. F. Veit, G. Schiefer, and M. Volkamer, *Exploring Phishing Threats through QR Codes in Naturalistic Settings*, Symposium on Usable Security and Privacy (USEC), 2024.

Chapter 3 is partly based on the publication:

> **M. Mossano**, A. Hennig, F. L. Ballreich, B. M. Berens, F. Sharevski, A. M. Sasse, and M. Volkamer, *A Framework for Developing Information Security Awareness Measures*, Symposium on Usable Privacy and Security (SOUPS Posters), 2025.

Chapter 4 is partly based on the publications:

> **M. Mossano** & M. F. Veit, T. Länge, B. M. Berens, F. Sharevski, and M. Volkamer, *Development, Evaluation, and Implementation of SEQR – a Usable Secure QR code Scanner*, Conference on Human Factors in Computing Systems (CHI), 2026.

> **M. Mossano** and M. Volkamer, *Literature Review: Misconceptions About Phishing*, Human Aspects of Information Security and Assurance (HAISA), 2024.

Chapters 5 and 6 are planned to be submitted to the upcoming ACM Conference on Human Factors in Computing Systems (CHI) 2027.

A complete list of the author's publications produced during the doctoral research is provided in the *List of Publications* at the end of this thesis.

2 QRishing Awareness Among QR Code Users

Naturalistic inquiries (i.e., studies) seek to understand human behavior within the context in which it naturally occurs (as described in Owen et al. (2008)). Two previous studies on QRishing approached this goal by investigating QR code interactions in real-world settings (i.e., the natural approach and scanning of a QR code that people take in their everyday lives): Vidas et al. (2013) deployed QR codes throughout a university campus and observed participant interactions through CCTV and through their actions after a QR code was scanned, while Krombholz et al. (2015) placed QR-code stickers in public locations in four European countries (Austria, Finland, France, and Greece) and collected participant responses through a survey. Both of these studies provided valuable insights into real-world QR code interactions (e.g., whether they checked a URL before visiting its corresponding webpage), and both studies investigated QR code interactions in a naturalistic setting without forcing the interactions on the participants.

However, neither of them followed the expected methodology of a naturalistic study: direct observation of the interaction, and researcher-collected data. Rather, they either observed the interaction vicariously (Vidas et al. (2013)) or not at all (Krombholz et al. (2015)), and they primarily focused on access logs and surveys to collect the data. None of the studies explored the reasons behind users' actions when interacting with QR codes, i.e., none of them interviewed the participants to determine why they scanned a QR code, and what influenced this action. More importantly, both of the previous studies predate the "COVID-19" pandemic. The pandemic represented a defining moment in the history of QR code usage, as it normalized the use of QR codes in everyday quick information exchange, e.g., restaurant menus, advertisements, or embedded in emails (as pointed out in Ricson (2026)). This normalization has likely introduced considerable differences in how users interact with QR codes compared to when both Vidas et al. (2013) and Krombholz et al. (2015) conducted their studies, and their results might no longer describe the current user behavior around them. As such, there was a gap in the literature regarding a naturalistic study of how QR code users interact with QR codes in the post-COVID-19 era and the reasons behind these interactions.

This chapter covers such a naturalistic study. First, the chapter presents the related work in Section 2.1. It then continues describing the study set-up in Section 2.2. The results of the study, shown in Section 2.3, highlight that 67% of the participants' answers mentioned visiting the URL in the QR code without first inspecting it, and that 52% of the decisions to scan the QR code or not were based on the pretext alone. The results are then discussed in Section 2.4, raising two main points: (i) users are concerningly unprepared to deal with QRishing, and (ii) their awareness of this threat needs be increased.

Contributions described in this chapter:

- Conduct a naturalistic study on the behavior of users when interacting with QR codes in naturalistic settings.
- Provide insights into whether users require additional support to detect QRishing.

The results described in this chapter have been published in:

- F. Sharevski, M. Mossano, M. F. Veit, G. Schiefer, and M. Volkamer, *Exploring Phishing Threats through QR Codes in Naturalistic Settings*, USEC 2024 (NDSS workshop). Awarded the *Distinguished Paper Award*.

Note: Roughly at the same time of the publication of the naturalistic study in this chapter, another work that had a similar goal and methodology was published, i.e., Bekavac et al. (2024). Their work was not considered at the time of the naturalistic study, but it is now discussed in Section 2.1.

2.1 Related Work

Several previous studies (i.e., Vidas et al. (2013), Krombholz et al. (2015), and Bekavac et al. (2024)) investigated the interaction with QR codes in naturalistic settings from a security perspective without forcing the interaction on the participants as would happen in a laboratory study. However, two of them (Vidas et al. (2013) and Krombholz et al. (2015)) were conducted before the COVID-19 pandemic normalized the use of QR codes, and the third (Bekavac et al. (2024)) happened at roughly the same time as the study in this chapter. More importantly, none of them applied a methodology entirely fitting the definition of a naturalistic study proposed by Owen et al. (2008) (i.e., that also employs direct observation and researcher-collected data). Two of the previous studies did not observe participants interacting with the QR codes (Krombholz et al. (2015) and Bekavac et al. (2024)), while the third study relied on CCTV footage rather than direct observation (Vidas et al. (2013)). Furthermore, none of the studies included interviews conducted

immediately after the interaction. Instead, they primarily based their findings on access logs and survey responses. As such, a different methodology was required to conduct a comprehensive naturalistic study fitting the definition from Owen et al. (2008).

Vidas et al. (2013) used an unassuming surveillance CCTV camera that recorded users' interaction with QR codes on a university campus in the US. They used four pretexts (shown in Figure 2.1), three pretexts with a QR code and one pretext with a tear-off URL. Every QR code contained the same URL, i.e., `http://2dpromo.net/9m36z`, which was also the URL used in the tear-off flyer. Once visited, the URL would take the participants to a survey, where they were asked if they knew what a QR code was (83% answered "Yes"), and what was the primary reason they chose to scan the QR code in a multi-choice question. 64% of the participants selected "Out of curiosity" as their reason for scanning, 14% selected "For fun", 4% because of the related information; 20% stated not to have scanned a QR code, as they reached the survey through the tear-off flyer. The participants were also asked whether they checked the URL before accessing it and 36% answered that they did not do so. Yet, the location of the QR code was in a computer science building, hence capturing the behavior of a relatively small and technologically-sophisticated population. Furthermore, the pretext choice is also a relevant aspect of the study, as the pretext around a QR code might influence the willingness to scan it. As such, using pretexts that are not especially complex (e.g., not a poster, as done in the study in this chapter) might have influenced not only the type of people interested, but also the believability of the offer.



Figure 2.1: Pretexts used in Vidas et al. (2013). From left to right: standalone QR code, standalone QR code with instructions how to scan it, QR code with a social network study deception, tear-off flyer with URL to survey.

Krombholz et al. (2015) addressed one of the limitations of Vidas et al. (2013) (i.e., the sample) by conducting an intercultural user study between four European countries (Austria, Finland, France, and Greece). They placed 784 QR code stickers containing a URL (not provided in the paper) in highly frequented urban spaces, e.g., bus stops, public toilets and universities. The stickers were of one of three varieties: a standalone QR code, a QR code with a description, and a QR code with a cute cat image. Note that, albeit the authors stated that the standalone QR code variety and the variety with the cat image performed best, no actual data is provided, and all the data is only reported in aggregated form. As such, no insights can be inferred from the use of different

varieties. Nonetheless, irrespective of the variety encountered, once the QR code was scanned it took the participants to a survey asking them several questions about risk perception of QR codes, whether they routinely check URLs before visiting them, and which reason they had to scan the QR code. The survey results show considerable differences between participants from different countries, with Parisian users being the most active users of QR codes, and Austrian users rarely if ever scanning a QR code. These differences were also shown in the trustworthiness perception, with French users being more skeptical than the others. Albeit an interesting experiment, the authors admit that they did not monitor the behavior of the users around the QR codes, meaning that there was no observation of how many decided to scan the QR code, but stopped before visiting the landing webpage. Furthermore, just like in Vidas et al. (2013), the effect of more complex pretexts (e.g., a poster) was not considered. As such, the study described in this chapter follows Krombholz et al. (2015) by recruiting everyday users in public spaces, but it also expands on both Vidas et al. (2013) and Krombholz et al. (2015) by embedding the QR codes within a complex pretext (i.e., a poster).

Bekavac et al. (2024) also addressed the limitations from Krombholz et al. (2015) and Vidas et al. (2013). The goal of their study was to evaluate whether specific design elements in the QR codes themselves (e.g., presence of graphics) would influence their perceived trustworthiness, and decrease the effectiveness of QR code tampering attack techniques, e.g., substitution of QR codes with attackers' created ones. To this end, the authors used two sets of four posters with QR codes (shown in Figure 2.2), with one set displayed in a Christmas market and the other at a university. The posters embedded a QR code, a tampered-looking QR code (i.e., as if a sticker was placed over it), a SafeQR (i.e., a QR code with transparency and added graphics), and a tampered-looking SafeQR. Each QR code contained URLs targetting the same domain (i.e., `https://qrco.de`, but with distinct paths, each one unique to one of the posters and composed by random-like characters (e.g., `/bedwWF`, or `/behUv6`). Once scanned, each QR code led the participants to a survey asking four questions: (1) *Did you notice anything suspicious about the poster/QR code before scanning it?*; (2) *Did you pay attention to the URL/internet address before scanning?*; (3) *How familiar are you with digital media & devices?*; and (4) *How good is your knowledge about phishing attacks?*. None of the questions were open text ones: questions 1 and 2 could be answered only with "Yes", "No", "Can't recall"; questions 3 and 4, instead, were answerable through one 5-point Likert scale each (5 being "Excellent" and 1 being "Poor"); these questions were an adapted version of the Quishing Awareness Scale from Sharevski et al. (2022). The results show that the participants were not suspicious of potential tampering in the QR code condition and the SafeQR condition, but that the tampered-looking SafeQR condition raised suspicion in more participants (71.40%) than the tampered-looking QR code condition (36.40%). Regarding the second question, 48.15% of the participants did not check the URL before accessing the survey, 37.04% said they check the URL, and 14.81% selected the "Can't recall" option. For the last two questions, the only



(a) Posters placed in the Christmas market setting.



(b) Posters placed in the university setting

Figure 2.2: Posters used by Bekavac et al. (2024) in their study. For both sets, from left to right: tampered-looking QR code, QR code, SafeQR, tampered-looking SafeQR

available data was that 92% of the participants rated their degree of familiarity with digital media & devices as high, and that 37% of the participants rated their phishing knowledge as moderate. According to Bekavac et al. (2024), their results show that blending QR codes within the overall layout of a pretext (e.g., by using a transparent QR code) can improve the user detection of QR code tampering. Unfortunately, both the study presented in this chapter and the one from Bekavac et al. (2024) happened at the same time, meaning that no insight from their methodology (e.g., the use of graphical elements in the QR code) could be integrated. Their results are discussed in Section 2.4.

2.2 Methodology

Section 2.2 starts by presenting the research questions in Section 2.2.1. Then, the section presents the study design used to answer these research questions (Section 2.2.2), followed by the pilot study used to determine the viability of the study set-up (Section 2.2.3), the ethical considerations

(Section 2.2.4), and the data analysis process (Section 2.2.5). The section concludes with the study limitations (Section 2.2.6).

2.2.1 Research Questions

The research in this chapter aimed at answering the following two research questions:

- **RQ1**: What factors attract and persuade users in naturalistic settings to scan a QR code and access the URL encoded in it?
- **RQ2:** What actions are taken by users in naturalistic settings when scanning a QR code and accessing the URL encoded in it?

As shown in Section 2.1, although several studies addressed similar questions before, none did so by employing a methodology comprehensively fitting the definition of a naturalistic study proposed by Owen et al. (2008). As such, given the methodological differences between the present study and the previous related works (i.e., direct observation and interviews with participants), no prior results provided sufficient support for formulating hypotheses. Consequently, the research questions were purely exploratory.

2.2.2 Study Design

This section describes the design of the naturalistic study shown in Figure 2.3. The section starts by describing the observation phase of the design in Section 2.2.2.1, and then proceeds to cover the interview phase in Section 2.2.2.2.

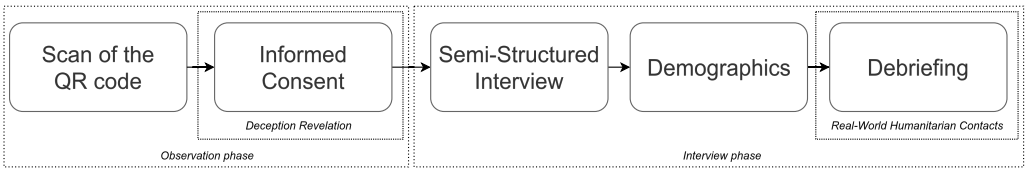


Figure 2.3: Study design followed to conduct the naturalistic study.

2.2.2.1 Observation Phase

This section covers the observation phase of the study design. It first describes the pretexts in Section 2.2.2.1.1, followed by the pretexts deployment Section 2.2.2.1.2, and the observation

protocol followed in Section 2.2.2.1.3. The section then concludes covering the informed consent step in Section 2.2.2.1.4.

2.2.2.1.1 Pretexts Description In order to answer the research questions, the study required the use of realistic and actionable pretexts that would fit in the naturalistic settings. QR codes can be delivered through both digital media (e.g., emails) and physical media (e.g., stickers, flyers, posters). The digital media were not suitable for the purposes of this study, as they would have required the targeted delivery of the pretext to specific individuals. Since the study relied on a convenience sample of users encountering the pretexts naturally in their everyday lives, it was not possible to determine in advance who would interact with the QR codes. As such, the pretexts had to be physical. The choice of which physical medium to employ was guided by RQ1, which aimed to investigate which factors attract and persuade users to scan a QR code. Consequently, the QR code needed to be embedded within a realistic pretext that participants could evaluate before deciding whether or not to interact with it. Furthermore, the interaction needed to be observable so that participants could be approached immediately afterwards for an interview. As such, smaller media such as the stickers and flyers employed by Vidas et al. (2013) and Krombholz et al. (2015) were not suitable. Thus, a poster was selected as study treatment (shown in Figure 2.4).

The poster measured 91.4×121.9 cm (or 36×48 inches, a so-called "portrait academic poster", according to Chartier (2025)). This size allowed the researchers to notice any interaction happening next to it, and made the poster sufficiently prominent to be easily noticeable without appearing intrusive. The poster's chosen topic was humanitarian (support for war-ravaged Ukraine) and the poster was designed to resemble pretexts used in real-world phishing campaigns circulating at the time of the study (as mentioned in both McCallum (2022) and Shoichet (2022)). The poster also leveraged the *social proof* persuasion principle of phishing (described in, e.g., Ferreira et al. (2015)) by hinting at solidarity through the use of a set of hands holding a globe, and using prominently the colors blue and yellow (i.e., the Ukrainian flag colors). Similar legitimate campaigns to help the people of Ukraine were happening at the time of the study (as reported by, e.g., Gibson (2023)), meaning that it was fairly reasonable to expect people to deem the pretext plausible. The poster contained a 20.32×20.32 cm (or 8×8 inches) QR code in a non-central but noticeable position, surrounded by a box stating "Scan me". All of these design decisions were guided by RQ1, and meant to determine whether the layout of the poster and its topic were sufficiently attractive to convince the participants to scan the QR code.

Regarding the QR code, even though the poster layout remained constant, the embedded QR code could have contained one of three *URL-types*:

- *Legitimate-looking* – `https://supportukrainianconflict.com/`;

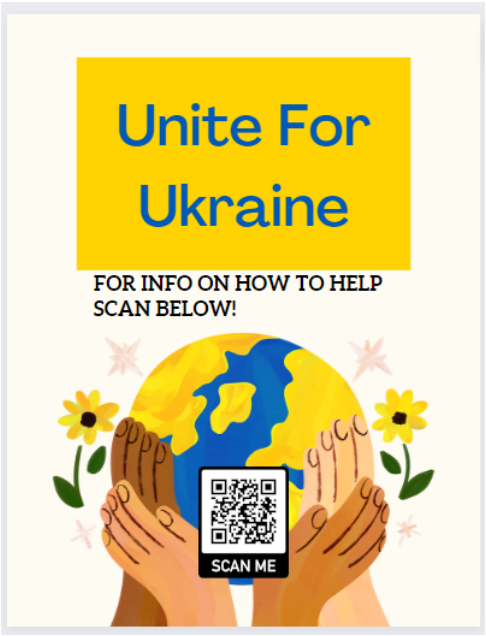


Figure 2.4: Phishing poster embedding the legitimate-looking URL type.

- *Explicitly-phishing* – `https://phishmeforukraine.com/`;
- *Short-URL* – `https://tinyurl.com/yck8w7xz`[1] redirecting to `https://unclesamsupportsukraine.com/`

The domains used were deliberately chosen to be unknown because an impersonation of a trusted/known domain might have been blocked before the end of the study, making the comparison difficult. Further ethical considerations behind the choice of the URLs are in Section 2.2.4.

The differences between the URL-types were designed to answer RQ2. In the *explicitly-phishing* URL-type, participants were expected to recognize the explicit cue "phish" included in the domain name. It is clear that no real-world phishing links would include such an explicit and obvious cue. However, the goal of the study was not to evaluate the effectiveness of URL attack techniques (i.e., attack techniques that make correctly reading a URL more difficult, e.g., arnazon.com with "r" and "n" instead of amazon.com), but whether participants would read the URL and consider such an obviously untrustworthy domain name as a factor for stopping the interaction.

[1] The short URL is no longer active.

In the *Short-URL* URL-type, instead, the goal was to investigate whether the use of a URL shortening service (i.e., "tinyurl") would raise questions about the potential untrustworthiness of the URL due to a known URL attack technique being used.

Each URL-type led to a separate webpage associated with the corresponding URL. However, the three webpages used the same layout and content, so that the URL displayed in the browser address bar differed while the webpages themselves appeared identical across conditions. The shared layout, shown in Figure 2.5, employed a minimalist visual design that emphasized the sign-up option. An image of Ukrainian soldiers holding red flares was used, with superimposed text stating that the alleged organization provided humanitarian support to Ukraine. Participants could then "sign-up" with their emails. As the ethical approval stated that, to avoid greater than minimal privacy risk, participants' emails could not be collected, the sign-up button had no functional effect, i.e., no email address was collected.

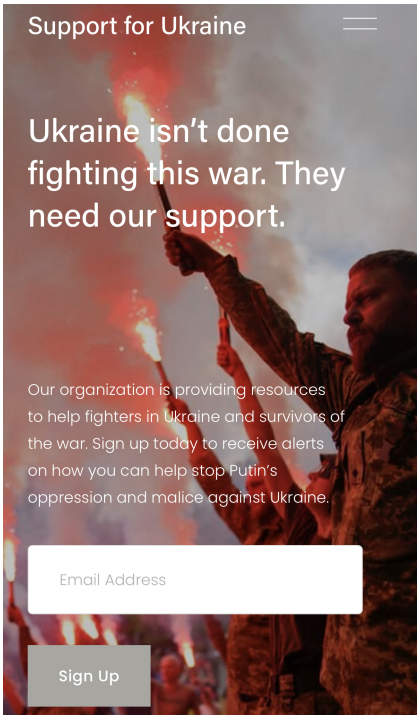


Figure 2.5: Landing webpage as seen on a mobile phone.

Unlike the studies discussed in Section 2.1, participants were not redirected to a survey page, and the interaction ended at this point with no further content being shown.

2.2.2.1.2 Pretexts Deployment The study was conducted in the metropolitan area of Chicago, US where the pretexts were placed in areas with high pedestrian traffic, i.e., a mall, a cafe, and a cultural point of interest (similarly to Krombholz et al. (2015)). As the public areas might have been part of a business, approval for the deployment of the pretexts was obtained by both the authorities and the location owners. Authorities and owners were also fully informed that an anonymous study about QRishing was being conducted, and that no actual phishing would have taken place, i.e., that both the pretexts and the URLs encoded in the QR codes were safe. The pretexts were integrated into the environment and placed next to other posters, so that they would blend in and draw passersby to them naturally. Each pretext was placed at a different location for one week (Monday to Sunday) for eight hours a day (from 10:00 to 18:00), with pretexts rotating between locations. The pretexts were displayed strictly under the researchers' supervision to avoid unobserved scans. Due to limited research personnel, only two locations were covered each week, leading to a total running time of six weeks or four weeks (224 hours) for each pretext.

2.2.2.1.3 Observation Protocol The researchers stood in the proximity of the pretext, observing and waiting for any interaction to take place, i.e., for the QR code to be scanned. When such interaction happened, one of the researchers checked the log of the webpage connected to the observed pretext and verified whether it was visited. This was done to confirm whether individuals stopped their interaction before visiting the webpage, which happened in seven cases (see Section 2.3.2). The researchers then approached the individuals, introduced themselves as researchers, and notified them that they had observed them scanning the QR code. The individuals were not considered participants until a consent to collect their data was obtained from them.

2.2.2.1.4 Informed Consent The researchers first asked the individuals whether they were at least 18 years of age to ensure their eligibility for the study. None of the individuals was younger than 18 years old, but there was nonetheless a debrief prepared regarding their ineligibility for the study (see Section 2.2.2.2.3 for more details on the debriefing process). The individuals were then informed explicitly of the research nature of the interaction, reassured that their email address was not collected (nor any other identifiable data), and asked whether they were interested in participating in a brief interview on their experience. The researchers then read a verbal script to each individual (shown in Appendix A.1), and obtained verbal consent to their participation. No individual refused to participate (in which case they would have also been debriefed). After the verbal consent was obtained, the interview phase began.

2.2.2.2 Interview Phase

This section covers the interview phase of the study design. It first describes the semi-structured interviews in Section 2.2.2.2.1, followed by the demographics questions in Section 2.2.2.2.2, and then the debriefing in Section 2.2.2.2.3.

2.2.2.2.1 Semi-Structured Interviews After verbal consent was obtained, participants took part in a brief semi-structured interview. Following the recommendations of Kallio et al. (2016), this methodology was chosen for two reasons. First, the research questions required an understanding of the participants' actions and the reasons behind these actions when interacting with QR codes. Second, the participants were expected to exhibit varying levels of QRishing awareness due to the limited coverage of QRishing in security awareness interventions. As such, the use of semi-structured interviews allowed the participants to describe their experiences and the reasons behind their actions freely while guiding the conversation through predefined themes and follow-up questions. This approach ensured that all interviews consistently addressed the research questions while allowing the researchers to probe participants' answers whenever additional clarification or detail was required (see the interview script in Appendix A.2).

The interviews were anonymous, and participants could withdraw at any point or refuse to answer any question that made them uncomfortable. Each interview was recorded using a portable voice recorder, and participants were informed both of the recording and that it would later be transcribed for analysis purposes. Furthermore, participants were asked to avoid stating any identifiable information during the interview. Although any such information would have been removed during transcription, this precaution avoided a period during which it would otherwise have remained in the audio recordings. Each interview lasted approximately 8–10 minutes, participation was voluntary, and, as a gesture of appreciation for their time, participants were offered free candy.

The interview consisted of three main questions focusing on the participants' reasons for interacting with the QR code, the actions they took after scanning it, and whether they inspected the associated URL. Each main question could be complemented by predefined and spontaneous follow-up questions to further explore the reasons behind the participants' answers whenever appropriate.

2.2.2.2.2 Demographics The final interview question collected demographic information, including participants' age, gender, race/ethnicity, educational attainment, and self-reported computer proficiency. These variables were collected to characterize the study sample and to assess

whether the participants reflected a diverse population rather than a specific demographic group, such as university students or individuals with a technical background.

2.2.2.2.3 Debriefing The study included a lengthy debriefing process at the end of each interview. First, the researchers ensured that no harm had inadvertently been caused to the participants by reiterating that the posters, the URLs, and the webpages were entirely under the researchers' control. Furthermore, the participants were reminded that the sign-up button on the webpages had no functional effect, that no email addresses or other identifiable information had been collected, and that they had therefore not been exposed to any privacy risk.

Next, the participants were shown examples of real-world phishing campaigns exploiting similar humanitarian pretexts (e.g., McCallum (2022), Shoichet (2022), and Office of Cybersecurity (2022)), explaining why such a pretext had been selected for the study.

To reassure participants that the use of such pretexts was not intended to be disrespectful to the Ukrainian people, the researchers explained that the purpose of the study was to raise awareness of the exploitation of humanitarian crises in phishing attacks. Participants wishing to support Ukraine were then provided with the contact details of legitimate humanitarian organizations (e.g., UNICEF (2026) and Amnesty International (2026)) where they could make a donation. Finally, the researchers shared their contact details and offered assistance should participants encounter suspicious URLs in the future. At the end of the debriefing process, participants were offered the opportunity to withdraw their data from the study, but none chose to do so.

2.2.3 Pilot Study

A two weeks pilot study was conducted with 18 participants (six per URL type) before the main study began at the beginning of 2023. The pilot had four objectives: (1) to verify the pretexts; (2) to test-run the poster placement and the approach; (3) to test the interview script and workflow; and (4) to identify and resolve issues in the overall data collection and analysis procedure. The same research setup detailed above was used, including the poster placement, the verification of age and eligibility, the verbal consent, and the lengthy debriefing process. Only one participant in the *explicit-phishing* URL-type inspected the URL for phishing cues, reasoning:

> "*I was suspicious when my phone read the URL as* `https://phishmefor ukraine.com/`*; Did not trust the site after that; Not about to give any information for a site with a URL that starts with "phish", unless it is for the band.*"

Most participants indicated that the humanitarian pretext attracted their attention, suggesting that the topic was relevant to the metropolitan Chicago population at the time of the study.

2.2.4 Ethical Considerations

The study received ethical approval from the Institutional Review Board (IRB) of DePaul University. The approval limited participation to individuals at least 18 years of age and stipulated that no email addresses or other identifiable information, such as IP addresses, could be collected.

Beyond these requirements, the use of a humanitarian pretext tied to real-world events required additional ethical considerations. Accordingly, the pretext was carefully designed to minimize potential ethical concerns. The pretext was carefully tailored not to impersonate real-world humanitarian aid efforts, so as to avoid impacting and unwillingly tarnishing the reputation of real-world actors. Furthermore, to support the participants' willingness to help, real-world alternatives were provided to them, so that they could actively help and contribute. Each participant who was provided with this information expressed an interest in the suggested organizations, but for privacy reasons, no record was taken of whether or not any subsequent action was taken. To address the potential discomfort caused by what could have been perceived as an unwarranted exploitation of a humanitarian crisis, an extensive debriefing was conducted to explain the necessity of the deception and to demonstrate that similar phishing campaigns were occurring at the time of the naturalistic study.

2.2.5 Data Analysis Process

Over the six-week deployment, the study attracted 42 participants, all of whom completed the interview and whose responses were included in the analysis. Due to the naturalistic format of the study, no formal invitation to participate or participant recruitment was possible.

The interview recordings were transcribed and analyzed using an inductive coding approach, as described by Saldaña (2013) and Thomas (2006), to identify frequent, dominant, or significant aspects in the participants' responses.

The author of the thesis first open coded all interview transcripts and developed an initial codebook. A second researcher then independently applied the initial codebook to all interview transcripts. After this first coding cycle, the *Inter-Rater Reliability* (hereafter, *IRR*) was calculated using Cohen's kappa (described in Cohen (1960)). The initial agreement was $\kappa = 0.58$, corresponding to *moderate agreement* according to Landis and Koch (1977), but was considered insufficient for the purposes of this study.

The coders then discussed all coding disagreements and revised the codebook to reflect the shared understanding reached during the discussion. Both coders then independently applied the revised codebook to all interview transcripts. Recalculation of the IRR yielded a Cohen's kappa of $\kappa = 0.90$, corresponding to *almost perfect agreement* according to Landis and Koch (1977), which was deemed acceptable for the purpose of this study.

After an acceptable level of agreement had been reached, the final codebook was organized hierarchically to reflect the two main themes that emerged from the analysis and were directly related to the research questions: (i) *Pretext*, comprising codes related to the humanitarian pretext and its perceived characteristics; and (ii) *Behavior*, comprising codes describing the participants' interactions with the QR code and the associated webpage. The complete hierarchical codebook is available in Appendix A.3.

2.2.6 Limitations

There are several limitations pertaining to the study.

First, the naturalistic setting restricted the study to a convenience sample of individuals encountered in high-pedestrian areas of the metropolitan area of Chicago during regular daytime hours. Consequently, the findings may not generalize to different populations, geographical locations, or settings in which QR codes are encountered under different circumstances, such as restaurants, parking facilities, or more remote locations.

Second, the humanitarian pretext was intentionally selected because it was expected to be relevant to the target population at the time of the study (year 2023). Different pretexts, countries, or periods of time may therefore lead to different participant behavior. Nevertheless, this reflects the adaptive nature of real-world phishing campaigns, which commonly tailor their themes to the targeted population and current events of public interest.

Finally, although the participants' interactions with the QR codes and the associated webpages were directly observed, ethical and privacy considerations limited the extent to which their behavior could be monitored. For example, whether the participants attempted to submit their email address was intentionally not observed to avoid collecting their information. Consequently, the reasons behind some of the participants' actions had to be reconstructed from the semi-structured interviews rather than through a direct observation alone.

2.3 Results

Section 2.3 first presents the demographic characteristics of the study sample (Section 2.3.1), followed by the qualitative findings addressing RQ1 (Section 2.3.2) and RQ2 (Section 2.3.3).

Note: Demographic information is available for all participants. However, not all participants answered every interview question, and individual responses could receive multiple codes. Consequently, the reported frequencies may differ from the total number of participants.

2.3.1 Demographics

In total, 42 individuals interacted with the posters, none of whom declined to participate. The sample was 38% female and 62% male, with a majority of participants having college education (64%), followed by post graduates (21%), high school degrees (7%), and unspecified degree (8%). Age-wise, 29% were in the [25-34] group, 24% in [18-24], 24% in [35-44], 17% in [45-54], 4% in [55-64], and 2% in [65+]. Regarding the participants' QR code proficiency, 52% reported it as "high," 43% "intermediate," and 5% "low." Because no control over the recruitment was possible (as discussed in Section 2.2.5), it was decided to collect the device model used to interact with the QR codes: 82% of the participants used Apple iPhones, 12% had a Samsung Galaxy (i.e., an Android smartphone), and 6% declined to release this information. The distribution of smartphone devices in the sample was consistent with the US smartphone market distribution at the time of the study (as shown in CounterPoint (2023)). All participants used the standard QR code scanner app for their smartphone model, while 64% used Safari and 36% used Google Chrome to open the embedded URLs.

2.3.2 RQ1 - Pretext Persuasion Factors

The results of the qualitative analysis addressing the first research question are shown in Table 2.1. As shown, the pretext's topic was the most frequently reported attraction factor. Most participants referred to its humanitarian aspect, stating that they scanned the QR code because they wanted to "*add [their] support to the Ukrainian people*". Others were attracted by the broader geopolitical context surrounding the war. For example, one participant explained: "*I'm not interested in scanning QR codes very much, [but] this one stood out, since I heard the war is actually past its one year mark*".

Participants also referred to several characteristics of the pretext itself. Some described it as "*friendly*" and '*unassuming*," reflecting its affective appeal. Others specifically mentioned the

pretext's "*vibrant colors*", while another highlighted the "*bold title 'Unite for Ukraine'*". Finally, the pretext's size was also reported as an attraction factor, with one participant stating that "*The size of the poster and QR code, could not miss it a mile away*".

Finally, general curiosity and situational factors also contributed to participants' decisions. Participants reported being "*drawn by the location of the poster in the coffee shop*" or scanning the QR code because they were "*just bored waiting for [their] girlfriend*" in a mall.

Table 2.1: Naturalistic Study – RQ1. Pretext Persuasion Factors Analysis Results

Code	Found	Percent of total
Pretext's topic	26	62%
Humanitarian aspect	22	52%
Geopolitical context	4	10%
Pretext's color	10	24%
Pretext's text	7	17%
Pretext's affective appeal	4	10%
Pretext's size	4	10%
General curiosity	3	7%

2.3.3 RQ2 - Actions Around QR Codes

The results addressing the second research question are presented in two parts. Section 2.3.3.1 reports the overall results (i.e., across all URL-types), while Section 2.3.3.2 presents the results separately for each URL-type.

2.3.3.1 Overall Results

The overall results of the qualitative analysis addressing the second research question are shown in Table 2.2.

The most frequently reported behavior was accessing the URL without first inspecting it in the QR code scanner. Rather than evaluating the displayed URL before proceeding, participants typically continued directly to the landing webpage. For example, one participant simply stated that they "*checked out the page*", while another reported that they "*just browsed through the page*".

The second most frequently reported behavior was inspecting the URL only after the landing webpage had opened by checking the webaddress bar. Several participants explicitly stated that they inspected the URL once it was displayed in the browser. For example, one participant explained that they "*looked at the URL and it said 'Ukraine' so they were OK with it*", while another noted that they inspected the URL "*once [they are] in the browser*".

Some participants did not inspect the URL but instead relied on the overall appearance and content of the landing webpage shown in Figure 2.5. Participants reported that they "*read the excerpt and thought it was safe to use, the website looked OK*", while another explained that they "*paid attention specifically to the people holding the flares*".

A smaller group of participants reported inspecting the URL directly in the QR code scanner before opening it. For example, one participant explained that they "*always look at the URL just to make sure it is not something odd*", while another stated that they "*look at the URL before [they] open it in a browser just to make sure it isn't something odd*".

Eight participants attributed their lack of URL inspection in the QR code scanner to limitations of the scanner's interface. For example, one participant explained that "*the URL disappeared so [they] tried to investigate but it was too late*", while another stated that they inspected the URL only "*once [they are] in the browser, because when the URL is highlighted yellow [they are] usually more focused on whether the QR code is working in the first place*".

Finally, seven participants reported inspecting the URL before deciding to abandon the interaction. For example, one participant stated that "*the URL was 'Phish Me,' which definitely would be a red flag*", while another explained that they "*stopped clicking because [they] saw the word 'phish*"'.

Table 2.2: Naturalistic Study – RQ2. Actions Around QR Codes: Overall Results

Code	Found	Percent of total
URL accessed without inspecting QR code scanner	28	67%
URL inspected in webaddress bar	19	45%
URL not inspected but website layout checked	13	31%
URL inspected in QR code scanner	9	21%
URL not inspected due to QR code scanner interface	8	19%
URL inspected and interaction abandoned	7	17%

2.3.3.2 Results Per URL-Type

The URL domain names were chosen to investigate how different URL characteristics influenced participants' actions after scanning a QR code. Specifically, the investigation examined whether participants behaved differently when presented with (1) an otherwise legitimate-looking URL, (2) a URL explicitly containing the term "phish" within the domain name, and (3) a short-URL service, a known URL attack technique.

The results on a per-URL type basis are shown in Figure 2.6. Each participant interacted with only one URL-type, resulting in an uneven distribution across the three conditions: 11 participants interacted with the legitimate-looking URL, 16 with the explicitly-phishing URL, and 15 with the short-URL. Although the aim was to achieve as balanced a distribution as possible, the lack of control over the recruitment process prevented this outcome. As stated previously, not all participants answered every interview question, and some responses contained conflicting statements. Consequently, the frequencies shown in Figure 2.6 do not necessarily add up to the number of participants for each URL-type.

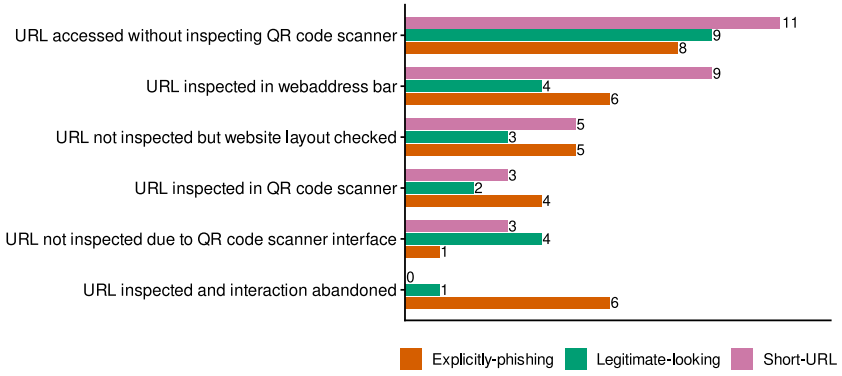


Figure 2.6: Naturalistic Study – RQ2. Actions Around QR Codes: Results Per URL-Type

Among participants exposed to the *legitimate-looking* URL-type, 82% (9/11) accessed the URL without inspecting it in the QR code scanner, whereas 18% (2/11) inspected it before proceeding. Furthermore, 36% (4/11) inspected the URL only after reaching the landing webpage by checking the webaddress bar. The 36% (4/11) of the participants exposed to the legitimate-looking URL-type attributed their lack of URL inspection in the QR code scanner to limitations of the scanner's interface. Only one participant inspected the URL before deciding to abandon the interaction.

Among participants exposed to the *explicitly-phishing* URL-type, 50% (8/16) accessed the URL without inspecting it in the QR code scanner. 12% (2/16) inspected the URL in the QR code scanner, while 25% (4/16) inspected it only after reaching the landing webpage by checking the webaddress bar. Only one participant attributed the lack of URL inspection in the QR code scanner to limitations of the scanner's interface. However, 38% (6/16) inspected the URL before deciding to abandon the interaction.

Among participants exposed to the *short-URL* URL-type, 73% (11/15) accessed the URL without inspecting it in the QR code scanner. 20% (3/15) inspected the URL in the QR code scanner, while 60% (9/15) inspected it after reaching the landing webpage by checking the webaddress bar. 20% (3/15) of the participants attributed their lack of URL inspection in the QR code scanner to limitations of the scanner's interface. No participant abandoned the interaction before reaching the landing webpage.

2.4 Discussion and Conclusion

The pretext used in this study reflected campaigns that were common at the time, both as legitimate fundraising initiatives and as phishing campaigns. This realism likely contributed to the participants' willingness to engage with the QR code, as reflected by 62% of the RQ1 codes referring to the topic. This seems consistent with the characteristics of believable phishing pretexts identified by Kersten et al. (2022), namely that effective pretexts should appear real, relevant, and persuasive.

The importance of the pretext is further supported by the variety of attraction factors reported by the participants. Beyond the topic itself, participants also referred to the poster's colors (24% of codes), text (17% of codes), affective appeal (10% of codes), and size (10% of codes). When these results are considered alongside those of Bekavac et al. (2024), they suggest that users evaluate the QR code content (e.g., the URL) in conjunction with its surrounding presentation rather than just as an isolated element. It should, however, be noted that the goals of the present study and that of Bekavac et al. (2024) differed. Whereas the present study investigated which characteristics of realistic QR code pretexts influence users' decisions to engage with QR codes, Bekavac et al. (2024) evaluated whether visual design features could help users detect the tampering of existing posters. Accordingly, the poster used in the present study represented a phishing campaign created by the attackers from scratch rather than a legitimate poster modified through the addition or replacement of a QR code. Nevertheless, both studies highlight a similar factor from two complementary points of view. Namely, that regardless of whether they are used to trick people or to support QRishing detection, the graphical elements of a pretext have a considerable impact on users' willingness to interact with QR codes.

These findings contrast with those of Vidas et al. (2013), where only 4% of the participants reported scanning the QR code because of the accompanying information. A likely explanation is the difference in pretext complexity. Whereas Vidas et al. (2013) relied primarily on standalone QR codes and relatively simple flyers, both the present study and Bekavac et al. (2024) embedded QR codes within richer and more realistic posters. Together, all of the above findings suggest that increasing the complexity and realism of the surrounding pretext influences users' willingness to engage with QR codes.

Regarding the results of the *second research question*, the data show that participants frequently attempted to assess the legitimacy of QR codes and their corresponding webpages, but often relied on non-security-relevant indicators to do so. Only 21% of the codes mentioned inspecting the URL in the QR code scanner before opening it, while 67% of the codes mentioned accessing the URL without such an inspection, despite the QR code scanner being the earliest opportunity to verify the destination. Moreover, only 45% of the codes mentioned inspecting the URL in the webaddress bar, while 31% of the codes showed reliance on the layout and content of the landing webpage to determine legitimacy. Similar observations have also been reported by previous studies. For example, Vidas et al. (2013) reported that 58% of the participants inspected the URL before accessing the QR code destination, but then still visited an unfamiliar and obscure domain. Similarly, Krombholz et al. (2015) found that more than half of the participants reported inspecting URLs before visiting the QR code destinations, despite almost the same proportion using QR code scanners that did not support URL inspection. Taken together, these findings suggest that users are often aware of the need to inspect QR code destinations, but frequently either do not understand how to interpret the information they obtain or misconceive which indicators are security-relevant (i.e., the URL) and which can be easily copied (i.e., the webpage layout).

Addressing the current situation necessitates a provision of an appropriate dsupport for users. While improvements of QR code scanner interfaces can facilitate this process (as shown in, e.g., Mossano et al. (2026b)), it is equally important to address the misconceptions surrounding QRishing and increase the users' awareness of the difference between security-relevant indicators and elements that can be easily copied by attackers. To achieve this goal, QRishing awareness interventions must provide effective and actionable guidance. However, as highlighted by both Redmiles et al. (2020) and Mossano et al. (2020), security awareness interventions often lack clear prioritization and present recommendations in ways that can confuse users. Consequently, before developing new QRishing awareness interventions, it is necessary to first establish a systematic framework that guides their development and evaluation. The next chapter addresses this challenge.

3 Framework for Developing Awareness Interventions

Governmental organizations such as the European Union Agency for Security (hereafter *ENISA*) released several guidelines to develop awareness interventions, e.g., on security crisis management (as done in ENISA (2024a)) or on education maturity (as in ENISA (2024b)). Academia also advanced several proposals on how to develop awareness interventions (as those in, e.g., Ghazvini and Shukur (2017), Jaeger (2018), Mayer (2019), Khader et al. (2021), Khando et al. (2021)). Despite these efforts, no unified methodology for developing awareness interventions currently exists. The aforementioned frameworks were developed independently, differ considerably in both scope and focus, and none encompasses all the aspects addressed by the others. Consequently, awareness intervention developers must either adopt a single framework, thereby omitting aspects covered elsewhere, or combine multiple frameworks without clear guidance on how to do so. This fragmentation of the development process is reflected in the resulting awareness interventions, which, as shown by both Mossano et al. (2020) and Redmiles et al. (2016), do not follow a standardized format, vary considerably in content, and lack a unifiying authoritative source of information.

Besides the fragmentation of awareness intervention development and the resulting awareness interventions, another challenge is that, as shown by Haney and Lutters (2018), users are generally unconcerned with information security. Consequently, awareness interventions should not only provide clear and actionable recommendations, but also appropriately incentivize the target population to follow them. This is particularly relevant because, as argued by Herley (2009), recovering from an occasional security incident is often less costly than continuously investing effort in avoiding every potential threat. Therefore, the development of effective awareness interventions requires a methodology capable of guiding developers not only in selecting and presenting appropriate awareness content, but also in designing interventions that effectively motivate the target population to follow such recommendations.

This chapter details an attempt to create such a framework. First, the chapter presents the related work that developed frameworks to design awareness interventions in Section 3.1, and then describes the initial version of the framework in Section 3.2. Afterwards, the extended

version of the framework is presented together with the rationale behind the introduced changes (Section 3.4). The chapter concludes with a discussion comparing the extended framework to the frameworks presented in the related work (Section 3.3).

Note: The figure illustrating version 1.0 of the framework retains the original terminology published in Mossano et al. (2025). However, throughout this chapter, the terminology has been aligned with the nomenclature introduced in Definitions and Terminology. Consequently, the terms "*awareness interventions*" and "*mitigation strategies*" are used in place of the terms "*awareness measures*" and "*mitigation measures*" shown in the Figure 3.8.

Contributions described in this chapter:

- Develop a framework for developing awareness interventions.

Parts of the results described in this chapter have been published in:

- M. Mossano, A. Hennig, F. L. Ballreich, B. M. Berens, F. Sharevski, A. M. Sasse, and M. Volkamer, *A Framework for Developing Information Security Awareness interventions*, SOUPS 2025 poster session.

3.1 Related Work

The reviewed frameworks approach the development of awareness interventions from different perspectives and emphasize different aspects of the development process. For instance, Jaeger (2018) focuses on factors relevant for the development of awareness interventions and how to address them, while ENISA (2024c) considers how to comply with European Legislation. Consequently, rather than competing with one another, they provide complementary insights into the design of awareness interventions. This section discusses the primary focus of each framework and briefly identifies the aspects incorporated into the extended framework. A more detailed comparison between the reviewed frameworks and the extended framework is provided in Section 3.3.

Ghazvini and Shukur (2017) proposed a framework for developing awareness interventions in the health sector, together with a case study conducted in a hospital in Malaysia. As part of the case study, the resulting awareness intervention consisted of the serious game *InfoSecure*, which was developed following the serious game design framework proposed by Yusoff (2010). The framework from Ghazvini and Shukur (2017) (shown in Figure 3.1) covers several aspects of the development process of awareness interventions, such as the organization's needs, the

target population, the awareness content, the delivery methods, as well as the implementation and evaluation of the resulting awareness intervention. Several of these aspects were incorporated into the extended framework, particularly those related to the organizational context, the target population, the awareness content, the delivery methods, and the implementation and evaluation of the awareness intervention.

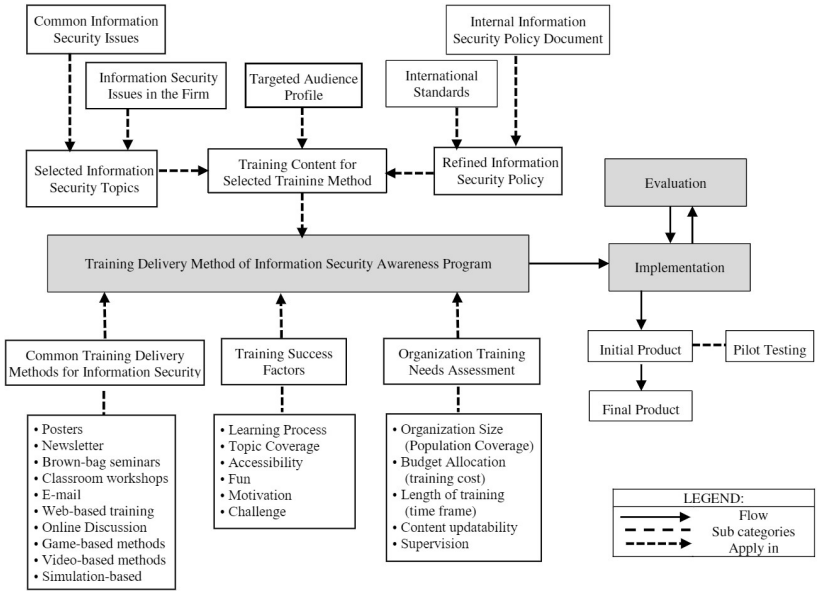


Figure 3.1: Ghazvini and Shukur (2017) framework on awareness interventions for the health sector.

Jaeger (2018) proposed a framework for developing awareness interventions based on a systematic literature review. The framework first identifies the factors relevant to the development of awareness interventions and subsequently describes how these factors should be addressed throughout the intervention (shown in Figure 3.2). In particular, it considers individual, organizational, socio-environmental, and technological factors, together with the evaluation of the resulting awareness intervention. The extended framework incorporates the individual factors through the stages addressing the target population and its awareness, while the organizational, socio-environmental, and technological factors are covered by the *Context* stage. Furthermore, the evaluation component from Jaeger (2018) informed Step 7 of the extended framework.

Mayer (2019) focused on the understandability of awareness interventions and on the feedback and evaluation phase (see Figure 3.3). The framework is quite detailed, has a clear directionality, and distributes the development process across four steps, the last of which is the evaluation of the awareness intervention. The extended framework incorporates the iterative feedback

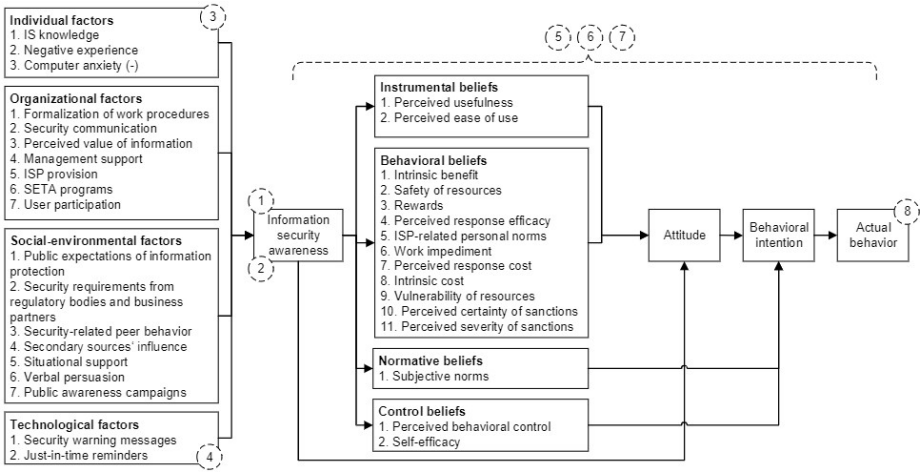


Figure 3.2: Jaeger (2018) framework on awareness interventions development).

and evaluation process proposed by Mayer (2019). It also expands on the preparatory and content development phases by explicitly covering the selection of the awareness goal, the target population, the awareness content, and the associated misconceptions and mitigation strategies.

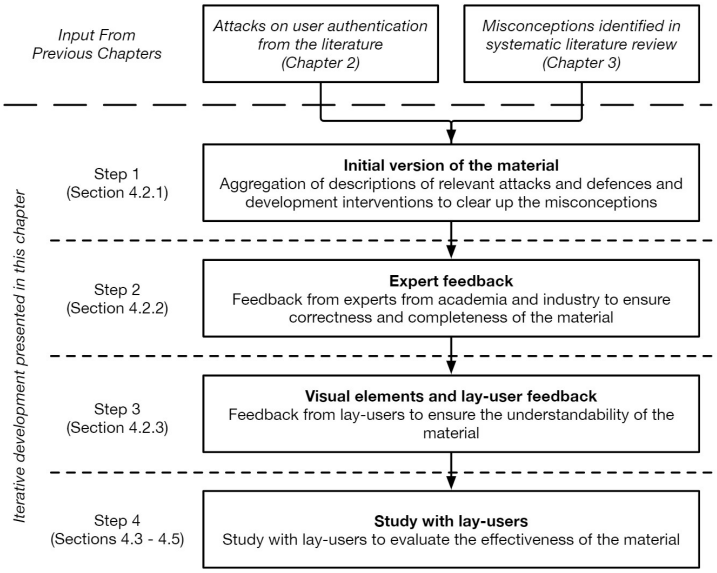


Figure 3.3: The framework presented in Mayer (2019).

Khader et al. (2021) developed a framework covering awareness interventions within the academic context (see Figure 3.4). The framework specifically addresses threats, attacks, and corresponding mitigation strategies tailored for the academic environment, together with the development, implementation, and evaluation of the resulting awareness intervention. These aspects were incorporated into the extended framework through the stages addressing the attack techniques, mitigation strategies, the format of the awareness intervention, and its evaluation. The extended framework further presents these activities through an explicit and self-contained development process, allowing developers to understand the sequence of actions directly from the framework without relying on additional textual explanations.

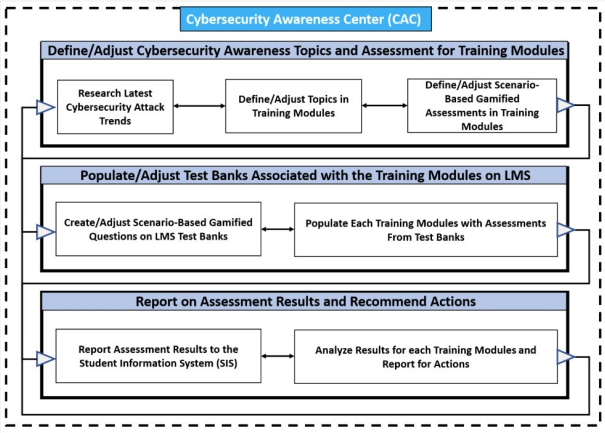


Figure 3.4: Khader et al. (2021) framework for the academic context.

Khando et al. (2021) compared the factors and methods relevant to the development of awareness interventions in both the business and public sectors (shown in Figure 3.5). The framework distinguishes between the factors influencing awareness interventions and the methods that can be employed to address them. These aspects were incorporated into the extended framework through the stages addressing the context, the target population, the awareness of the target population, the format of the awareness intervention, and its evaluation. The extended framework further provides a structured methodology describing how these aspects should be applied throughout the development process and how they relate to one another.

Dialog für Cyber-Sicherheit (2022) (hereafter *DCS*) produced a guideline document for developing and evaluating awareness interventions. The guidelines follow a five-step process covering the identification of the target population, the definition of the awareness goal, the creation of the awareness content, the implementation of the awareness intervention, and its evaluation (shown in Figure 3.6). These aspects were incorporated into the extended framework, which further

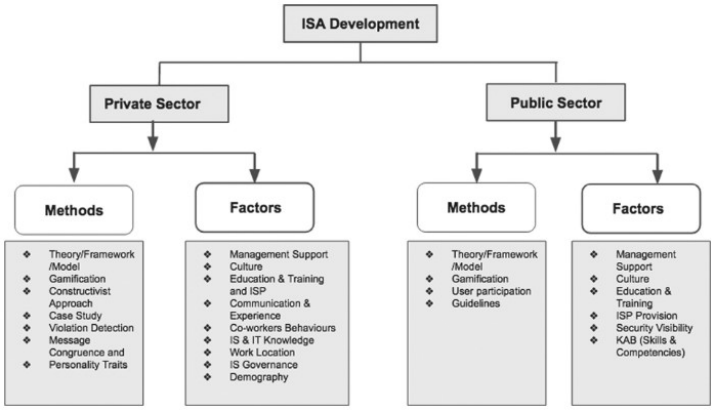


Figure 3.5: Khando et al. (2021) framework for private sector and public sector.

expands each step into a detailed development methodology and explicitly represents the relations between the individual stages.

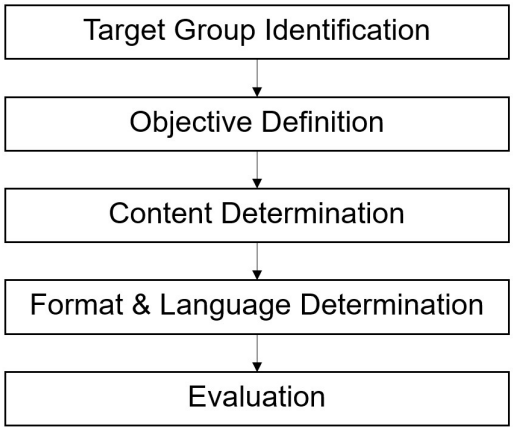


Figure 3.6: Representation of the process presented in Dialog für Cyber-Sicherheit (2022).

ENISA (2024c) presented a set of guidelines towards compliance with Articles 40 and 41 of the European Electronic Communication Code (as published in European Parliament and Council (2024)). Although the guidelines primarily focus on communication infrastructure, one of the 29 security objectives, SO6, addresses the security awareness of personnel (see Figure 3.7). SO6 provides a high-level overview of the implementation of awareness interventions within organizations, including the periodic update of the awareness material and the evaluation of the personnel's

security awareness. These aspects were incorporated into the extended framework, which further provides a structured methodology for developing, implementing, and evaluating awareness interventions, thereby supporting organizations in fulfilling the requirements established by SO6.

	Security measures	Evidence
1	a) Provide key personnel with relevant training and material on security issues.	i. Key personnel have followed security trainings and have sufficient security knowledge (interview).
2	b) Implement a program for training, making sure that key personnel have sufficient and up-to-date security knowledge. c) Organise trainings and awareness sessions for personnel on security topics important for your organisation.	ii. Personnel have participated in awareness sessions on security topics. iii. Documented program for training on security skills, including objectives for different roles and how to reach them (by e.g. training, awareness raising, etc.).
3	d) Review and update the training program periodically, taking into account changes and past incidents. e) Test the security knowledge of personnel.	iv. Updated security awareness and training program v. Results of tests of the security knowledge of personnel. vi. Review comments or change logs for the program.

Figure 3.7: The framework security interventions from ENISA (2024c).

3.2 Version 1.0 of the Framework for Developing Awareness Interventions

This section presents a step-by-step description of version 1.0 of the framework for developing awareness interventions (shown in Figure 3.8; hereafter, *Framework 1.0*). Framework 1.0 was developed by synthesizing the frameworks from the literature reviewed in Section 3.1 into a single development methodology. To this end, the frameworks from the literature were analyzed to identify the sub-processes they addressed, called *stages* in Framework 1.0. For instance, the *Domain* stage or the *Content* stage. Each stage was then related to every other stage influenced by its output; for example, the *Target population* stage influences several other stages, but is itself not influenced by any other stage. Finally, the stages were grouped according to whether they contributed to a common objective, leading to seven *Steps*. Each Step covers a specific aspect of the development process; for example, *Step 1* consists by preparatory stages, while *Step 6* contains the stages responsible for synthesizing the content. It is not possible to proceed to the next Step without addressing all the stages of the previous Step. Likewise, each Step must be completed in order, without exceptions, i.e., it is not possible to go from *Step 1* to *Step 3* without first completing *Step 2*. The fundamental idea behind the framework is to be abstract enough to

be applicable to several security awareness contexts, without being tied to a specific one (e.g., the academic security awareness context as Khader et al. (2021)).

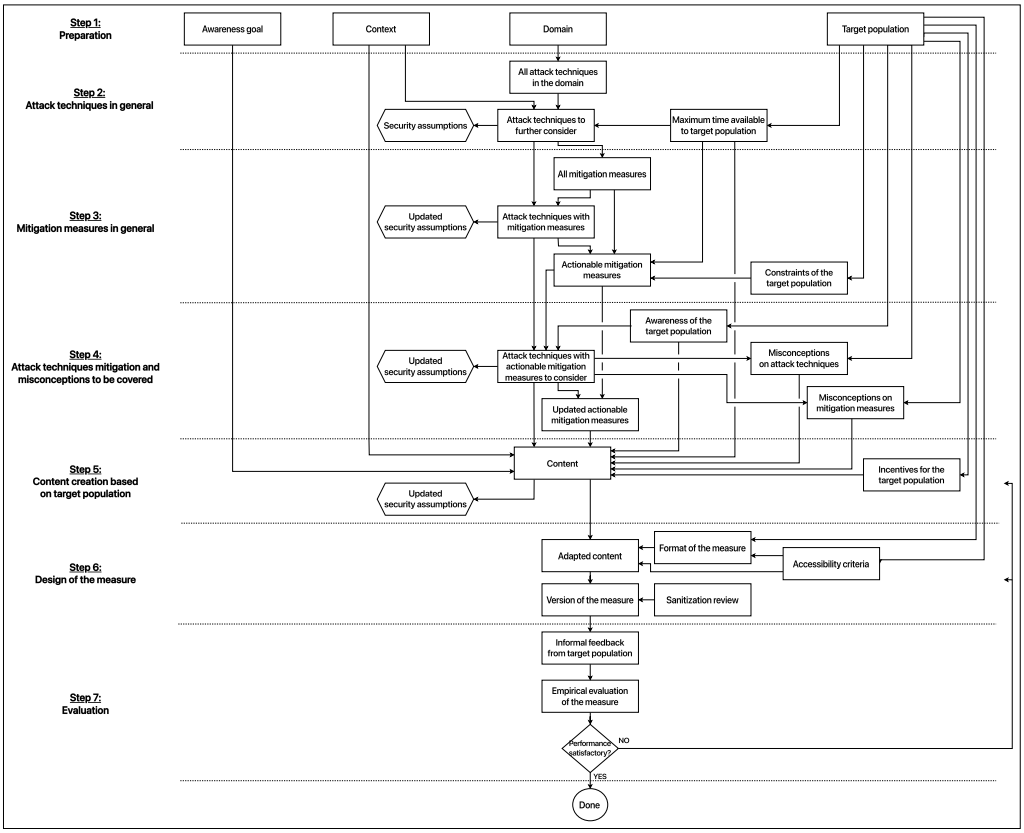


Figure 3.8: The version 1.0 of the framework for developing awareness interventions.

Note: For consistency, the titles of the following subsections correspond to the names of the Steps shown in Figure 3.8.

3.2.1 Step 1 – Preparation

In *Step 1*, the developers determine the four foundational aspects of the awareness intervention: the awareness goal, the context of use, the domain of interest, and the target population.

The *Awareness Goal* stage covers the final outcome expected after consuming the awareness intervention. This needs to be defined early on, and it needs to be precise enough to guide the

creation of the content later on (covered in Section 3.2.4). Namely, simply determining that the awareness goal is “to be able to detect phishing through QR codes" is not enough; rather, the developers needs to select an appropriate measure to evaluate whether the goal was reached or not. For example, an appropriate awareness goal could be “achieve the ’Believe: Self-Efficacy’ level from Schöni et al. (2024)”. Such a goal can be evaluated, e.g., by collecting the participants’ phishing detection rate and a self-efficacy scale measure, making it precise enough to fulfill the step requirements. The basis set to determine the awareness goal is also used to measure the awareness already possessed by the target population (as explained in Section 3.2.4).

The *Domain* stage addresses the domain of interest, representing which aspect of security the awareness intervention covers, e.g., phishing through QR codes. The domain can be as precise as it is needed, meaning that, while an awareness intervention might cover phishing through QR codes on a general level, another awareness intervention might instead focus on a more detailed “phishing through QR codes that affects Android devices”. This choice, clearly, also informs many of the other aspects of the awareness intervention.

The *Target population* stage identifies the specific segment of the user base that the awareness intervention is aimed at; this slice can be as wide or as narrow as it is needed. For example, an awareness intervention might be targeted at the whole general population, while a different awareness intervention only targets the employees of a certain company.

The *Context* stage includes every aspect of the actual, real-world use of the awareness intervention, e.g., which technology is included, where the awareness intervention would be displayed, and how the target population is expected to access it. *Note*: Differently than the *Format of the intervention* stage (covered in Section 3.2.6), the *Context* stage is mostly on a theoretical level. For example, the *Context* stage might aimed at developing “a webpage for company XYZ”, while the *Format of the intervention* stage is the practical implementation of said webpage (e.g., HTML, inclusion of JavaScript, the logos, the corporate colors, etc.).

3.2.2 Step 2 – Attack techniques in general

In *Step 2*, the *Domain* stage informs the *All attack techniques in the domain* stage, where developers collect the attack techniques pertaining to the domain. These can be based on a new investigation conducted as part of the awareness intervention developing process, e.g., through an industry database of attack techniques, or a systematic literature review of attack techniques. Another option is to base the attack techniques on an already available collection, albeit in this case the developers needs to check whether new attack techniques have surfaced since the collection was created and compiled.

Once that is done, the *Target population* stage from *Step 1* informs the *Maximum time available to the target population* stage, i.e., how much time a member of the target population can dedicate to consume the awareness intervention. This can vary greatly, depending on the target population chosen and the context: an employee cannot dedicate the same amount of time to an awareness intervention than an individual that choose to look for information themselves. Because for an employee any time spent on the awareness intervention is time not dedicated to their work, long awareness interventions might not only cause losses for the business, but also damage the employees themselves (e.g., if the loss of productivity ends up causing a reduction in their salary). Conversely, an individual has virtually however much time they desire to dedicate to an awareness intervention. This being said, however, very long awareness interventions are unlikely to keep readers engaged throughout the whole time necessary to consume them. As such, even when dealing with individuals, developers must always respect their time and create awareness interventions that are only as long as it is needed, i.e., only containing the information needed to reach the awareness goal stipulated. For example, if a target population is technically sophisticated and another one is not, reaching the same awareness goal is likely to require more time for the second, which needs to be factored in as part of the maximum time available to each population.

After the *Maximum time available to the target population* stage is complete, this, alongside the *Context* stage from Step 1, are used to inform the *Attack techniques to further consider* stage, where the relevant attack techniques based on the aforementioned two stages are selected among all of those collected in the *All attack techniques in the domain* stage. For example, some attack techniques can be part of the domain, but the associated targeting technologies not considered in the *Context* stage, meaning that they should not be covered. Conversely, other attack techniques might be considered for the awareness intervention, but deemed too high of a technological sophistication for the target population is required to understand their working, meaning that explaining them would cause the awareness intervention to exceed the maximum time available.

The *Security assumptions* stage is where developers document those attack techniques that are excluded because they are assumed to be addressed in other ways (e.g., by an email filter). In particular, developers must explicitly state the reasons behind the lack of coverage, how the attack techniques are assumed to be addressed, or why it is not possible to address them. The developers are expected to explicitly track the security assumptions (as defined in Definitions and Terminology) and the reasons behind them, so that if different developers were to employ the awareness intervention, they would be informed of what is covered and what is not covered. Furthermore, tracking the security assumptions and the reasons behind them also helps to identify whether some of the assumptions become unreasonable in the future. For example, an attack technique assumed to be stopped by an email filter might no longer be stopped in the future, leading to the re-evaluation of the awareness intervention's effectiveness and (potentially) to its update.

3.2.3 Step 3 – Mitigation strategies in general

In *Step 3* starts with the *All mitigation strategies* stage, where the developers try to collect all the mitigation strategies for the attack techniques not covered by the security assumptions.

The mitigation strategies can be collected through the same methods and channels used in the *All attack techniques in the domain* stage (described in Section 3.2.2). In case an attack technique does not have a mitigation strategy, or if a mitigation strategy only partially addresses an attack technique, the developers can either generate a new mitigation strategy, or determine that none is possible. If no mitigation strategy is available for any number of attack techniques, these attack techniques are not covered by the awareness intervention, and a corresponding security assumption(s) is formulated in stage *Updated security assumptions*.

The *Constraints of the target population* stage then determines the characteristics and abilities of the target population. These can be justified through previous literature on the target population itself, or a study can be conducted to determine them. For example, if an awareness intervention addresses phishing through emails and targets the general population, previous literature showing that users have difficulties reading URLs can be used to determine that this characteristic is a constraint of the target population.

The *Constraints of the target population* stage then informs the *Actionable mitigation strategies* stage, which determines whether or not the mitigation strategies are *actionable*, i.e., whether or not it is possible for the target population to comprehend and apply them. For example, some mitigation strategies might simply be too complex to either comprehend or apply for lay users. Obviously, different target populations have different constraints: A target population of IT experts, due to their expertise, is able to comprehend and apply a higher number of mitigation strategies than lay users. Furthermore, the maximum time available is also a factor to determine which of the mitigation strategies are actionable. Practically, this means that actionable mitigation strategies not only need to be comprehendible and applicable by the target population, but there must also be enough time to explain them. For example, some mitigation strategies might be comprehendible and applicable by the target population given that enough time is spent to explain them. However, the time required to do so might simply be too long for the maximum time available to consume the awareness intervention, meaning that those mitigation strategies become non-actionable for the specific awareness intervention being developed due to the maximum time available.

3.2.4 Step 4 – Attack techniques mitigation and misconceptions to be covered

In *Step 4*, The first stage the developers address is the *Awareness of the target population* stage, where the determine the awareness maturity of the target population, i.e., the level of awareness already possessed by the target population about the domain. How the awareness maturity of the target population is measured is up to the developer, but any measure used must be: (1) at least compatible, ideally the same, as the one used to set the awareness goal in *Step 1*; (2) replicable; and (3) clearly quantifiable. Although the last two requirements should be true for every aspect of the framework, they are highlighted here to avoid that developers simply "guess" the awareness maturity, and instead identify and follow an appropriate methodology to measure it.

Regarding the relation with the awareness goal, it is clear that to determine whether the goal is reachable or not, the only way is to compare it to the awareness already possessed by the target population. For example, if no awareness is present, for the target population to reach a high level of awareness by the end of the awareness intervention would require to spend considerable effort and time. As such, ideally the same basis used to determine the awareness goal is used to measure the awareness already possessed by the target population. Considering the example from Section 3.2.1, a way to determine the awareness of the target population might be to measure it and assign it to the appropriate level on the awareness maturity curve from Beyer et al. (2015).

The quantifiability requirement is important because it allows the comparison of an awareness intervention with other similar interventions, or against previous versions of the same awareness intervention. For example, if an awareness intervention requires updating, quantifying the target population's awareness can help the developers assess whether certain parts of the intervention may be omitted due to increased overall awareness within the target population. Admittedly, it might be that no quantification of the awareness is feasible at all, for example, because the target population is the general population. In such cases, developers should still select a measure and assume that the awareness maturity level of the target population is the lowest level. This assumption considers the least prepared members of the target population as the benchmark, so that no awareness information is left out, otherwise the least prepared members are exposed to risks unknown to them. Furthermore, having a measure in place still allows the comparison of the awareness before and after the intervention consumption, thus providing a clear measure of the effectiveness of the awareness intervention.

Once the developers measured the awareness of the target population, this is used to inform the *Attack techniques with actionable mitigation strategies to consider* stage. This is the stage where developers select those attack techniques and actionable mitigation strategies to address in the awareness intervention. As mentioned before, this entails excluding those attack techniques and

mitigation strategies that are either already known by the target population. Those that are not selected are then added to the security assumptions with a clear explanation of the reason, so that they can be reintegrated if a need arises (for example, due to a change in the awareness level in the future).

When the attack techniques and the mitigation strategies to consider are determined, the developers identify which misconceptions on both of them are present in the target population in the stages *Misconceptions on attack techniques* and *Misconceptions on mitigation strategies*. Considering misconceptions on mitigation strategies might initially seem counterintuitive, given that the awareness intervention itself is meant to teach the target population which mitigation strategies to apply. However, users can nevertheless hold misconceptions regarding how they *believe* they should protect themselves, leading them to believe that certain mitigation strategies are effective when they are not. For example, users might incorrectly believe that checking whether a website uses HTTPS is sufficient to determine its legitimacy, despite HTTPS only guaranteeing that the connection to the server is encrypted. The identification of both types of misconceptions can be carried out in several ways, from interviewing members of the target population to conducting a systematic literature review of misconceptions already identified by previous works on the same target population. The specific methodology to identify misconceptions is again up to the developers, and is influenced by the context or the intended audience. For example, the developers of an awareness intervention for a specific company might be primarily interested in the misconceptions among the employee population of their company. Conversely, for an awareness intervention aimed at the general public, a systematic literature review of previous studies might be sufficient to identify the most prevalent misconceptions to address. The important point is that the method to collect the misconceptions must be clear and replicable, so that an explicit justification is provided.

3.2.5 Step 5 – Content creation based on target population

At *Step 5,* the developers first need to identify which incentives the members of the target population have to interact with the content of the awareness intervention in the *Incentives for the target population* stage. In other words, the developers must identify suitable appeals that are personal enough to engage the interest of the target population. These appeals can vary greatly depending on the context: an employee's interest can be related to following the policies introduced by their company, while the general population might require incentives that are both more personal and, at the same time, general enough to be applicable to each member (or at least, to the majority of them). Identifying them can be achieved in a variety of ways, from interviews with the employees to consideration of relevant theories on decision making (e.g., the Protection Motivation Theory from Rogers (1975)).

Once the incentives have been identified, the developers starts creating the content, which is not only influenced by the outcome of the previous steps, but it is effectively a consolidaiton of their output. The attack techniques and the mitigation strategies, with their respective misconceptions, are explained in a way that is: (1) understandable to the target population, (2) respectful of its maximum time available, (3) appropriate for the context, and 4) capable of increasing the awareness of the target population to an awareness goal set from the onset.

3.2.6 Step 6 – Design of the intervention

In Section 3.2.1 it is described how part of the context identified in *Step 1* is the high-level description of how the awareness intervention is accessed by the target population. In *Step 6*, the developer, having prepared the content in *Step 5*, can begin the practical steps needed to implement the awareness intervention in the *Format of the intervention* stage. Clearly, this stage consists of several sub-stages that depend on the selected format. For example, a video would require the preparation of a script, the creation of the graphics, the recording of the audio track, etc. Conversely, a web-page would require the determination of how to code the content, whether, e.g., Javascript is required or not, if the layout needs to follow a specific corporate one or not, etc.

Part of the formatting is also the level of accessibility to implement, addressed in the *Accessibility criteria* stage, and that can vary greatly based on the target population. For example, a webpage meant for a target population of screen reader users would require far more effort at this stage than a webpage targeting the general public. It is important to notice that absence of any accessibility requirements is highly unlikely, given that even the general population would require a basic level of accessibility (for example, the implementation of the W3C standard presented in World Wide Web Consortium (2026)). Thus, the *Accessibility criteria* stage should always be carefully considered for any target population.

The content is then adapted to the chosen format in the *Adapted content* stage. This could require different levels of work depending on the distance between the original content and the adapted one. Admittedly, it is likely that the three stages described so far in Step 6 will happen roughly in parallel, as implementing the content fundamentally already adapts it. Yet, it was decided to keep the three stages separated for clarity.

At this point, the developers should make sure to not expose the readers to any unintended risk. To do so, a sanitization of every potential source of risk is carried out in the *Sanitization review* stage. For example, in an awareness intervention on phishing, the developers might want to make sure that the example URLs used are not actually active phishing URLs. This should be done both when the awareness intervention is being developed, and at predetermined future intervals, to avoid that previously safe URLs turned phishing could harm the readers at a later time. Once

the content is adapted, this leads to the first actual version of the awareness intervention in the *Version of the measure* stage.

3.2.7 Step 7 – Evaluation

In *Step 7*, the developers evaluate the awareness intervention in the version generated at the end of *Step 6*. The first evaluation is an informal feedback with the target population itself, covered in the *Informal feedback from target population* stage, and it is meant to include their input in the development process, as well as to address any possible issue with the awareness intervention's comprehensibility. This feedback can be collected in several ways, from cognitive walkthroughs to more structured interviews or surveys. The "informality" does not come from a relaxed methodology, but rather it is tied to the absence of statistical power required. In other words, the feedback should follow a clear methodology, but it does not need to be conducted with a statistically representative sample.

Once the feedback is received, and any changes required are implemented, the new version is then empirically evaluated in a rigorous study with a statistically representative sample of the target population in the *Empirical evaluation of the intervention* stage. Just like the informal feedback, the methodology used depends on the awareness intervention being evaluated. For example, a series of interviews can be conducted, alongside a quantitative evaluation meant to demonstrate the effectiveness of the awareness intervention against a control group, but also against other interventions with similar goals. Furthermore, whether the evaluation itself is adequate for the awareness intervention, the goal, etc. is an important aspect of this stage. However, covering how to adequately conduct evaluations is outside the scope of this thesis, and it is therefore not addressed here.

If the performance of the participants is deemed acceptable, i.e., if the awareness intervention is indeed statistically significantly effective in changing the behavior of the target population, then, the process ends. Otherwise, the developers must go back and identify the issues that impeded the significance achievement. This means that the developers trace the problems to either the content or the implementation, fix them, and then go again through all the steps necessary to reach the evaluation phase. This is repeated until the awareness intervention leads to a statistically significantly better performance than the other groups in the evaluation.

3.3 Comparison of Framework 1.0 with Previous Frameworks

Framework 1.0 included all the aspects from the various frameworks presented in Section 3.1. In this section, Framework 1.0 will be compared against the frameworks from related works to highlight how they were merged and expanded upon.

When comparing Framework 1.0 to the framework in Ghazvini and Shukur (2017), it can be seen that the *Context* stage integrates all the aspects presented in Ghazvini and Shukur (2017), often times conolidating them. Namely, the aspects of *International Standards*, *Internal Information policy*, *Refined Information Security Policy*, and all the features of *Organization Training Needs* except for the *Length of Training* are covered by the *Context* stage. The *Length of Training* aspect, instead, is part of the *Maximum time available to the target population* stage. Furthermore, the aspects of *Common Information Issues* and *Selected Information Issues* are covered with a higher degree of granularity through the several stages that address attack techniques. The target population is considered in Ghazvini and Shukur (2017) as informing only the *Training Content for Selected Training Methods* aspect. Framework 1.0 greatly expands the role of the target population, in particular by also considering its influence on the delivery methods (identified in the *Format of the intervention* stage) and on the accessibility requirements (in the *Accessibility criteria* stage). Furthermore, the *Accessibility criteria* stage combines the accessibility-related aspects, which Ghazvini and Shukur (2017) separate between the *Common Training Delivery Methods for Information Security* aspect and the *Training Success Factors* aspect. The other factors of the *Training Success Factors* aspect are instead part of the *Incentives for the target population* stage. The remainder of the aspects related to the evaluation and implementation are covered in both *Step 5* and *Step 6*.

Regarding the framework in Jaeger (2018), Framework 1.0 includes the *Individual factors* as part of the *Target population* stage and of the *Awareness of the target population* stage. The *Organizational factors*, *Socio-environmental factors*, and *Technological factors* are instead covered by the *Context* stage. Framework 1.0 further expands on these aspects by explicitly describing the development process of awareness interventions. Admittedly, the framework in Jaeger (2018) provides a more detailed description of the evaluation phase. However, what Jaeger (2018) presents is but one possible evaluation, aimed specifically at understanding the behavioral shift caused by an awareness intervention. As such, that part of Jaeger (2018)'s framework is covered by *Step 7* of Framework 1.0 as one of the potential methodologies to evaluate an awareness intervention (assuming that the evaluation goal is specifically to measure the behavioral shift it causes and not, e.g., its usability or another effect). In this sense, Framework 1.0 is less restrictive than

the framework in Jaeger (2018), as it leaves open the possibility of adopting different evaluation methodologies depending on the awareness goal.

The framework in Mayer (2019) has a higher level view than Framework 1.0. The attacks and the misconceptions are also collected, however, the entire selection process of attack techniques, mitigation strategies, and misconceptions, which constitutes the bulk of Framework 1.0, is instead consolidated in the *Initial version of the material* step in Mayer (2019). Framework 1.0 expands this step by explicitly describing the individual stages required during the content selection process. The remaining of Mayer (2019)'s framework covers the evaluation, with two important differences from Framework 1.0. The first difference is that the feedback parts of Mayer (2019) are condensed in the *Informal feedback from target population* stage. Furthermore, Framework 1.0 explicitly mentions the target population, not only lay users. The relevancy of this difference is that a lay user from a different target population might not give as meaningful feedback as somebody who is intended to consume the awareness intervention (e.g., a lay user from Germany would not give the same feedback as one from the United States). As such, Framework 1.0 not only integrates all the steps from Mayer (2019), but further specifies them.

When Framework 1.0 is compared to the framework in Khader et al. (2021), what is first noticeable is that Khader et al. (2021) also employs directional relations to define how to proceed through the development process in the form of arrows. However, the graphical representation requires the accompanying text of the paper to be fully understood. First, the graphical representation does not explicitly indicate where the development process starts. The text of the paper clarifies that *Phase 1*, i.e., where to begin, is the top-most one (see Figure 3.4), namely the *Define/Adjust Cybersecurity Awareness Topics and Assessment for Training Modules* phase. However, *Phase 1* being the starting point is not immediately obvious from the graphical representation. Second, given the double-sided arrows between the sub-elements of each Phase, the graphical representation does not explicitly indicate which of the elements should take priority. Here again, the paper's text specifies that the sub-elements of a Phase are to be followed from left to right. For instance, in the *Define/Adjust Cybersecurity Awareness Topics and Assessment for Training Modules* Phase, the process would start from the *Research Latest Cybersecurity Attack Trends* sub-element, continue in the *Define/Adjust Topics in Training Modules* sub-element, and then finish in the *Define/Adjust Scenario-Based Gamified Assessments in Training Modules* sub-element.

However, each of these sub-elements is connected with double arrows to the ones next to them, giving the impression that developers can move from the central *Define/Adjust Topics in Training Modules* sub-element to the left-most *Research Latest Cybersecurity Attack Trends* sub-element. If this is done, given that the arrow to the left of the latter sub-element points towards the sub-element itself, it is again unclear whether developers are expected to trace back to where they began or not. Framework 1.0 provides a more explicit representation of the development process,

with the directionality of the actions (i.e., top-down) and the back-tracking points (in *Step 7*) being clearly indicated. Regarding the single sub-elements of each Phase, they are either integrated in the *All attack techniques in the domain* stage (e.g., *Research Latest Cybersecurity Attack Trends*), the *Format of the intervention* stage (e.g., *Create/Adjust Scenario-Based Gamified Questions on LMS Test Banks*), or in one of the stages of *Step 7* (e.g., *Analyze Results for each Training Modules and Report for Actions*). As such, Framework 1.0 not only integrates the Phases and sub-elements of the framework in Khader et al. (2021), but also provides a more explicit representation of how developers are expected to proceed through the development process.

Regarding Khando et al. (2021), the *Factors* aspects are subdivided across several stages. The *Context* stage covers those factors related to, e.g., the company or the technology used. The *Target population* stage and the *Awareness of the target population* stage, instead, cover those aspects related to the demographics, the education, and the experience. Regarding the *Methods* aspects, they are part of the *Format of the intervention* stage (e.g., gamification) and are further subdivided across the various stages in *Step 7* (e.g., user participation). Overall, Framework 1.0 expands these aspects by explicitly describing how they are applied throughout the development process and by representing the relations between them.

The framework in Dialog für Cyber-Sicherheit (2022) provides a high-level description of the same process described by Framework 1.0: the target population is identified, the goal is set, the content is created, the format implemented, and the awareness intervention is then evaluated. As such, Framework 1.0 covers each one of these steps while providing a more detailed methodology and explicitly representing the relations between the steps and their stages.

Finally, the framework in ENISA (2024c) differs from the others. While the other frameworks described so far proposed models and processes to develop awareness interventions, ENISA (2024c) states the requirements that companies must fulfill to comply with the European Electronic Communication Code. As such, the relevant question in this case is whether applying Framework 1.0 would support companies in fulfilling these requirements. Starting from level 1, Framework 1.0 supports companies during the creation of the training content and material to be provided to key personnel. Framework 1.0 also addresses the interview requirements for gathering evidence in *Step 7*. Regarding level 2, Framework 1.0 explicitly considers the context of use, the target population, and the awareness goal, thereby supporting the creation of awareness interventions tailored to the organizational needs and the documentation of their development. In level 3, Framework 1.0 facilitates the update of awareness interventions by explicitly identifying the stages that need to be revisited whenever changes occur, whether because of new attacks or changes in the context or the target population. Furthermore, the practical implementation of awareness interventions is explicitly addressed in the *Format of the intervention* stage. Importantly, even if Framework 1.0 does not explicitly cover the requirement of performing routine evaluations of both

the personnel's awareness and the awareness interventions themselves, this is implicitly covered by the *Context* stage: Given that such requirements are part of the regulations that companies must follow, they are considered among the contextual aspects to be addressed. As such, even if Framework 1.0 is not meant to substitute the ENISA Guidelines, it would support companies in complying with them.

Given the above comparisons, Framework 1.0 integrates the elements presented in the frameworks from related work while consolidating them into a single development methodology. Furthermore, it makes explicit several stages that were previously implicit, such as the systematic process used to determine which attack techniques and mitigation strategies should be covered. As such, Framework 1.0 provides a structured methodology that can be applied independently of the context of use, the application domain, or the target population. At the same time, it allows developers the flexibility to adopt the methods they deem most appropriate to achieve each stage and each Step, provided that these choices are reasoned and documented.

3.4 Version 2.0 of the Framework for Developing Awareness Interventions

The framework described in Section 3.2 served as the basis for the development of the QRishing awareness intervention presented in Chapter 5. During its application, however, several issues were found.

First of all, the framework was not precise enough when describing the relations between the various stages. Some stages influenced several more stages than shown in the framework. This lack of clarity understated the importance of certain stages, which were in reality much more influential than appeared from the framework. The most telling example of this is the *Domain* stage, which seemed to influence only one other stage (i.e., *All attack techniques in the domain*), when in reality it influenced every other stage throughout the entire development process. For this reason, each stage relation was reconsidered and, when required, restructured to account for the actual influence exercised.

Secondly, although the idea behind the framework was always to be abstract enough to be applicable to several security contexts (as mentioned at the beginning of Section 3.2), the abstraction did not sufficiently reflect the granularity required during practical application. This led to several stages appearing as being a single stage, when in reality several occurrences of the same stage happened at different points, both within and across the steps. As such, the importance of certain steps was vastly understated. For this reason, several more stages were added when necessary to better clarify what the developers are expected to achieve before moving to the next step.

Furthermore, the role of the *Security assumptions* stage was revised. Security assumptions can be formulated or updated at different points throughout the framework, starting from *Step 1*. Once formulated, these security assumptions can then influence subsequent stages. For this reason, the *Security assumptions* stage became an overarching stage with a two-way relation with the other stages.

Finally, the terminology used in the name of both stages and steps is now in-line with the nomenclature stipulated in Definitions and Terminology.

All of these extensions lead to the development of the version 2.0 of the framework for developing awareness interventions (hereafter, *Framework 2.0*), shown in Figure 3.9. In the remainder of Section 3.4 and throughout its subsections, each step is described again, highlighting the changes from the previous framework. Only the changes are explicitly mentioned, as the descriptions from Section 3.8 (and its subsections) are still valid for Framework 2.0 whenever the stages went unmodified.

Note: As in Section 3.2, the titles of the following subsections correspond to the names of the Steps shown in Figure 3.9. As such, there are some differences in the subsections titles between Section 3.2 and this Section.

3.4.1 Step 1 – Preparation

The most evident change in *Step 1* is that it is now visually obvious which are the most influential stages. The *Domain* stage now influences everything underneath itself, meaning that every other stage throughout the process is dependent on the domain of the awareness intervention. Similarly, once formulated, the security assumptions from the *Security assumptions* stage can influence every other stage. Furthermore, any decision in any stage might require the formulation of a new security assumption (e.g., regarding an attack technique, a misconception, a mitigation strategy, etc.). This two-way interaction is now made explicit. Albeit these seem like obvious statements, this hierarchy was not explicit before. Other stages also had they influence made more obvious: for example, it is now clear that the *Context* stage is the third-most influential one, closely followed by the *Target population* stage.

Several stages originally part of later steps have now been moved to *Step 1*. Namely: stage *All attack techniques in the domain* was moved up from *Step 2*, and stages *Format of the intervention* and *Accessibility criteria* were moved up from *Step 6*.

Regarding the *All attack techniques in the domain* stage, *Step 1* is meant to be the “preparation step”, where no actual development is done but where most of the necessary information used at later steps is collected. Since knowledge of the existing attack techniques is a necessary

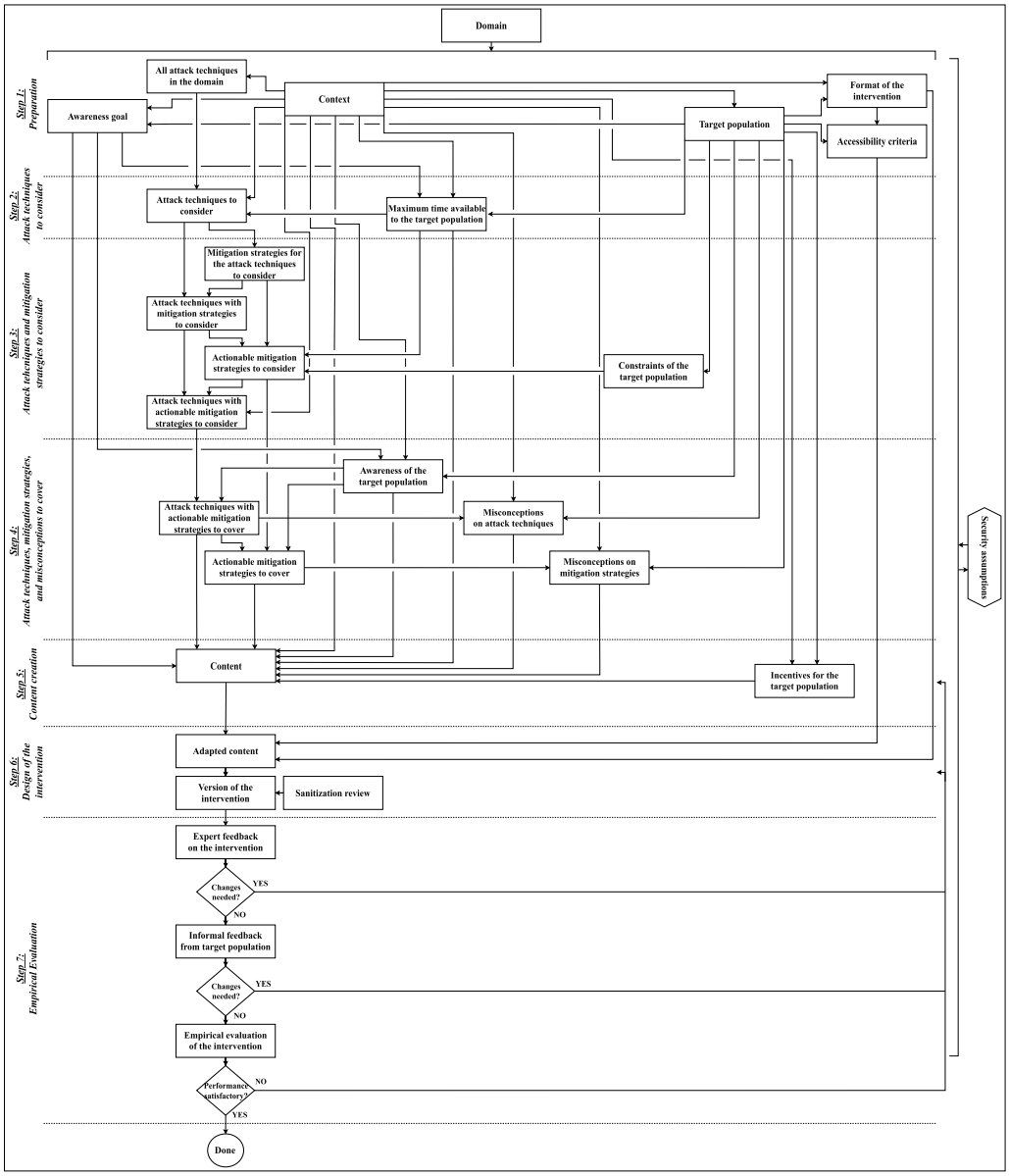


Figure 3.9: Version 2.0 of the Framework for Developing Information Security Awareness Interventions.

preparation before starting the refinement process of the later steps, the stage was moved to *Step 1*. Furthermore, in the previous framework no influence from the *Context* stage was accounted for. This meant that the developers were expected to find attack techniques also for technologies

that they were not covering. For example, if an awareness intervention was to target specifically users of Windows machines, the developers would have also had to collect attack techniques aimed at, e.g., Macintosh machines. Furthermore, without the influence of the *Context* stage, developers would have also had to collect attack techniques for mobile devices when focusing only on desktop devices. Certainly, many attack techniques work across platforms, but some are specifically tailored for certain contexts only, and this should be accounted for in order to avoid wasting time and resources. As such, Framework 2.0 now covers these considerations.

Similarly, the stages *Format of the intervention* and *Accessibility criteria* were moved from *Step 6* to *Step 1*, albeit for a different reason than the *All attack techniques in the domain* stage. It was noticed that the practical implementation decisions covered by these stages needed to be considered earlier than originally represented. Although the high-level format of the awareness intervention is already determined as part of the *Context* stage (as described in Section 3.2.1), the specific implementation details covered by the *Format of the intervention* stage can influence the development process before the implementation itself begins. For example, a webpage might require the determination and acquisition of a meaningful domain, or the set-up of a server to host it. Furthermore, even if this was not the case during the development of QRNope, these considerations elicited a more general one regarding the potential involvement of external actors during the development process. For example, if an accessibility audit is required, the accessibility auditors[1] must be selected earlier to avoid waiting for the search to complete before proceeding with the audit. Furthermore, if third parties are involved, public calls or negotiations regarding a service costs might be necessary. As such, Framework 2.0 accounts for these potential parallel processes by moving the respective stages to the beginning of the development process.

3.4.2 Step 2 – Attack techniques to consider

Besides the aforementioned moving of the *All attack techniques in the domain* stage to *Step 1*, the only change in *Step 2* was the increase of relations between stages. Namely, it was noticed that the *Maximum time available to the target population* stage was also influenced by the *Context* stage and the *Awareness goal* stage, and not only by the *Target population* stage. For example, the maximum time available is notably different between, e.g., the private and business context. Similarly, given the same awareness goal, a target population of lay users would need more time to reach it than a target population of IT experts. Both of these influences are now made explicit in Framework 2.0.

[1] An accessibility auditor is an individual or organization specialized in evaluating the accessibility of a product, service, or system based on compliance standards such as International Organization for Standardization (2025).

3.4.3 Step 3 – Attack techniques and mitigation strategies to consider

Step 3 underwent a considerable restructuring.

First of all, the *All mitigation strategies* stage was substituted with the *Mitigation strategies for the attack techniques to consider* stage. It was noticed that identifying and collecting (or creating) mitigation strategies for attack techniques that are not going to be considered would require collecting mitigation strategies that ultimately would not be used. As such, only mitigation strategies addressing the attack techniques to consider are collected or created.

Secondly, it might be that some of the mitigation strategies in the *Actionable mitigation strategies to consider* stage are deemed non-actionable due to the influence of the *Constraints of the target population* stage and the *Maximum time available to the target population* stage. If the non-actionable mitigation strategies are the only mitigation strategies for certain attack techniques, these attack techniques need to be assumed as addressed by other means (or not at all), and appropriate security assumptions must be formulated.

Thirdly, the *Attack techniques with actionable mitigation strategies to consider* stage was moved from *Step 4* to *Step 3* for two reasons. First, this allowed to differentiate *Step 3*, which covers the stages that select attack techniques and mitigation strategies to consider, from *Step 4*, which determines what will be covered by the content. Second, moving the *Attack techniques with actionable mitigation strategies to consider* stage to *Step 3* allowed to consider the influence of the *Context* stage much earlier than before. In Framework 1.0, the *Context* stage only influenced the attack techniques either when these were still without their corresponding mitigation strategies (i.e., in *Step 2*), or when the content was generated (i.e., *Step 5*). In so doing, it might have been that attack techniques that could have been excluded because their mitigation strategies were not fitting to the context, were considered until the later Steps of the development process. This caused the creation of content for attack techniques that would not actually be covered in the awareness intervention. As such, considering the influence of the *Context* stage already in *Step 3* allowed to reduce the number of attack techniques and mitigation strategies carried over to later Steps, effectively reducing the workload for the developers.

3.4.4 Step 4 – Attack techniques, mitigation strategies, and misconceptions to cover

Step 4 underwent a similar restructuring and extension to that of *Step 3*.

Firstly, *Step 4* now makes a clear terminological distinction between those attack techniques and mitigation strategies "to consider", i.e., to be processed further, and those attack techniques and mitigation strategies "to cover", i.e., for which content will be generated. The distinction was introduced to highlight that *Step 4* is where what will be part of the content is determined.

Secondly, during the application of Framework 1.0, two undocumented influences were noticed. First, it was noticed that several stages were influenced by the *Context* stage, despite this influence not being represented. As such, the influence of the *Context* stage was explicitly added to the *Awareness of the target population* stage, the *Misconceptions on attack techniques* stage, and the *Misconceptions on mitigation strategies* stage. This was done because, for example, the misconceptions (be them on attack techniques or on mitigation strategies) of a group of employees in a small business might be different from the misconceptions of a group of employees in a corporation, due to, for example, a security training received at work. Furthermore, some attack techniques and mitigation strategies are unique to certain specific technologies. For example, the same malware downloaded on a Windows machine and on a Macintosh machine could lead to very different outcomes (e.g., because bundled within an .exe file, which would be interpreted differently by the two operating systems). Still, a user might have the misconception that the same risks are present, or that the same mitigation strategies are to be followed.

Second, it was noticed that the *Awareness goal* stage also influences the *Awareness of the target population* stage, given that, depending on their starting awareness, the awareness goal might be easier or harder to reach. Therefore, without considering the awareness goal, and how well or badly the target population is situated to reach it, the description of the awareness of the target population would be incomplete. Thirdly, it was noticed that the *Misconceptions on mitigation strategies* stage should be influenced by the *Actionable mitigation strategies to cover* stage, and not (as it was before) by the *Attack techniques with actionable mitigation strategies to cover* stage. This relation is now correctly shown in Framework 2.0.

3.4.5 Step 5 – Content creation

No changes were implemented in *Step 5*, which remains unmodified from the one in Framework 1.0 and presented in Section 3.2.5.

3.4.6 Step 6 – Intervention Implementation

Besides moving the *Format of the intervention* stage and the *Accessibility criteria* stage to *Step 1* (as described in Section 3.4.1), no further changes were implemented in *Step 6*. As such,

the remaining stages of *Step 6* are unmodified from the ones in Framework 1.0 presented in Section 3.2.6.

3.4.7 Step 7 – Evaluation

When *Step 7* was reached during the application of Framework 1.0, it was noticed that this step was missing a fundamental aspect of human-centered design: the target population (and its features, constraints, etc.) was considered throughout the development process, but the feedback integration was limited. Namely, two issues were found.

The first issue was the lack of expert feedback before the empirical evaluation. Framework 1.0 only considered the feedback received from the target population, without accounting for the feedback of experts on the awareness intervention. Such feedback can potentially highlight issues with the awareness intervention (e.g., its understandability) before the target population is asked, and before the empirical evaluation is conducted. Thus, expert-indicated issues can potentially be solved earlier in the process. As such, Framework 2.0 introduces the *Expert feedback on the intervention* stage.

The second issue was related to the integration of the feedback received. In Framework 1.0, feedback was explicitly integrated only after the *Empirical evaluation of the intervention* stage. This meant that, when strictly following the framework, developers might have been required to empirically evaluate an awareness intervention for which issues had already been identified during the feedback stages. This not only introduced the risk of wasting time and resources, but also conflicted with the iterative refinement expected from a human-centered methodology. As such, Framework 2.0 introduces decision nodes after both the *Expert feedback on the intervention* stage and the *Informal feedback from the target population* stage. These decision nodes allow developers to return as far back as *Step 5* and address the identified issues before proceeding with the empirical evaluation.

Then, if the empirical evaluation identifies further issues, the existing feedback loops allow developers to repeat the necessary steps until the desired effectiveness is achieved.

3.5 Conclusion

This chapter presented the process followed to develop a framework for developing Awareness interventions. Starting from the analysis of existing frameworks, Framework 1.0 was first developed to combine and expand the existing methodologies into a single development process. The

application of Framework 1.0 then highlighted several aspects that required further refinement, leading to the development of Framework 2.0. The resulting methodology provides a systematic process to guide developers from the initial definition of the awareness intervention's context and goals to its implementation and evaluation.

Before applying Framework 2.0 to the development of a QRishing awareness intervention, the information required to complete the *All attack techniques in the domain* stage, the *Misconceptions on attack techniques* stage, and the *Misconceptions on mitigation strategies* stage must first be collected. To this end, two systematic literature reviews and one systematic industry database review were conducted. Presenting these reviews within the corresponding stages of *Step 1* and *Step 4* would substantially interrupt the description of the development process. Therefore, as an editorial choice aimed at improving the readability of this thesis, these systematic reviews and their results are presented separately in the next chapter.

4 Inputs Required by Framework 2.0: *All Attack Techniques in the Domain*, *Misconceptions on Attack Techniques*, and *Misconceptions on Mitigation Strategies*

Chapter 3 presented the development of Framework 2.0, a methodology for developing awareness interventions. Before the content of an awareness intervention can be developed, Framework 2.0 requires the collection of information regarding the domain and the target population. Specifically, this chapter addresses the inputs required for the *All attack techniques in the domain* stage, the *Misconceptions on attack techniques* stage, and the *Misconceptions on mitigation strategies* stage.

The *All attack techniques in the domain* stage requires developers to collect the attack techniques relevant for the domain of the awareness intervention. As described in Chapter 3, this collection is influenced by the *Domain* stage, the *Context* stage, and the *Security assumptions* stage, meaning that the resulting list is not expected to contain every possible attack technique, but the attack techniques relevant to the scope of the intervention being developed. The *Domain* stage, the *Context* stage, and the *Security assumptions* stage are covered in detail in Section 5.2.1, and here only the influencing factors relevant for the attack techniques collection are summarized. The domain of QRNope is defined as "Phishing attacks using QR codes that either encode the payload[1] directly or take the user to the payload within one step". For example, a QR code encoding a phishing URL, or a QR code encoding a legitimate URL leading to an app store that points to malware. Furthermore, the context is that of lay users scanning QR codes encountered throughout everyday life using iOS devices and the default QR code scanner of the iOS operating system (hereafter, *iOS Camera*). This context led to the formulation of the security assumption that only attack techniques applicable to the selected operating system and not already addressed by it are

[1] The term "payload" is used in this work to broadly identify any malicious content (be it malware, injection attacks, phishing URLs, etc.) delivered by an attack vector and used to achieve an attacker's objective.

considered. As such, the attack techniques collected in this chapter are those applicable within this domain, context, and security assumption.

The *Misconceptions on attack techniques* and *Misconceptions on mitigation strategies* stages require developers to collect the misconceptions users have regarding the attack techniques and mitigation strategies considered for the awareness intervention. As described in Chapter 3, this collection is influenced by the *Domain* stage, the *Context* stage, the *Target population* stage, and the *Security assumptions* stage. Similarly to the *Domain* stage, the *Context* stage, and the *Security assumptions* stage mentioned above, also the *Target population* stage is covered in detail in Section 5.2.1. For the purpose of the misconceptions collection, QRNope targets lay users living in the United States, and does not require specialized IT knowledge or previous phishing awareness to understand the content presented. Furthermore, the security assumptions do not influence the collection of misconceptions presented in this chapter, but only the specific misconceptions covered by QRNope (described in Section 5.2.4.3 and Section 5.2.4.4). As such, the misconceptions collected in this chapter are those applicable within this domain, context, and target population, without accounting for security assumptions.

The collection of these inputs requires dedicated methodologies and produces results that are used during the development of QRNope in Chapter 5. However, Framework 2.0 does not prescribe a specific methodology to populate these inputs. Instead, developers are free to choose the methodology they prefer, as long as the methodology chosen is justifiable and replicable. For the purpose of the development of QRNope, two systematic literature reviews and one systematic industry database review were used.

This chapter describes the methodologies and the results of these systematic reviews, and it is divided into two parts. The first part describes the process and the results of the attack techniques collection (in Section 4.1). The second part addresses the collection of users' misconceptions on attack techniques and mitigation strategies (in Section 4.2).

Note: These processes could have been explained as part of the development of QRNope. However, given the length needed to properly describe them, it was decided to move them to their own chapter to avoid bloating Chapter 5. As such, this chapter position represents an editorial choice for readability.

Contributions described in this chapter:

- Provide a categorization of QRishing attack techniques.
- Provide a list of users' misconceptions on QRishing.

The results described in this chapter have been published in:

- **M. Mossano**, M. F. Veit, T. Länge, B. M. Berens, F. Sharevski, and M. Volkamer, *Development, Evaluation, and Implementation of SEQR – a Usable Secure QR code Scanner*, Conference on Human Factors in Computing Systems (CHI), 2026.
- **M. Mossano** and M. Volkamer, *Literature Review: Misconceptions About Phishing*, Human Aspects of Information Security and Assurance (HAISA), 2024.

4.1 QR Code Attack Techniques in the iOS Environment

The first input required for the development of QRNope comes from the *All attack techniques in the domain* stage. As described in the introduction of this chapter, this input should contain the attack techniques relevant within the domain and context of the awareness intervention being developed. For QRNope, this requires identifying attack techniques that use QR codes to deliver payloads and that are applicable to the considered context.

Previous works presented surveys and taxonomies of phishing attack techniques (e.g., Franz et al. (2021), Abdillah et al. (2022), Ayeni et al. (2024), Veit et al. (2026a)), which can be used by researchers and developers to identify relevant attack techniques and corresponding mitigation strategies. However, existing works do not provide an up-to-date categorization of QRishing attack techniques applicable to the domain and context of QRNope. Therefore, for the *All attack techniques in the domain* stage, a taxonomy of attack techniques in iOS was created (hereafter, *the taxonomy*).

This section describes the process followed to create the taxonomy. Namely, this section first addresses the existing related work (Section 4.1.1), then it describes the methodology followed (Section 4.1.2), and the resulting taxonomy (Section 4.1.3). After the taxonomy, the section covers how the results were adapted for the development of QRNope (Section 4.1.4).

4.1.1 Related Work

Krombholz et al. (2014) presented the most common QR code uses at the time of writing (advertising, mobile payment, access control and augmented reality and navigation), followed by the two attack vectors they identified, i.e., (1) replacing the entire QR code, and (2) modify individual modules of a QR code. They then proceeded to present three real-world examples

of QRishing attacks: (1) the use of USSD codes to wipe a phone (now patched by Samsung; originally shared on YouTube by spamloco (2012)), (2) the use of injection attacks (the version from Sharma (2012)), and 3) the use of obfuscated URLs with typos (the version from Moore and Edelman (2010)). After these real-world examples, Krombholz et al. (2014) highlighted the open challenges for security awareness and usable security design guidelines (which are outside of the scope of the taxonomy, and therefore not discussed). Given that Krombholz et al. (2014) was one of the earliest works on QRishing, the number of attack techniques presented is limited compared to later works. Furthermore, the work did not rely on a systematic literature review, making it difficult to assess the completeness of the identified attack techniques. As such, although the attack techniques described by Krombholz et al. (2014) are also represented in the taxonomy, additional work was needed to provide a more complete overview of QRishing attack techniques.

Yong et al. (2019) is a more recent survey of attack techniques in the QRishing domains. Differently from Krombholz et al. (2014), the number of attack techniques is higher, covering both those already present in the work of Krombholz et al. (e.g., the modification discovered in Kieseberg et al. (2012)) as well also more complex techniques (e.g., *QR-in-QR*, originally presented in Dabrowski et al. (2014), and at the time still called *barcode-in-barcode*). More interestingly, Yong et al. (2019) also covers non-technical attack techniques besides typos in URLs, (e.g., the use of very long URLs that would be cut from the screen), and highlight how QR code scanners would open the URL without allowing the users to check them. They then present a series of mitigation strategies, but, as mentioned before, this is outside the scope of the taxonomy and will not be discussed. From the perspective of the methodology, similar as Krombholz et al. (2014), Yong et al. (2019) does not include a systematic literature review. However, compared to Krombholz et al. (2014), the lack of a systematic methodology has a stronger impact, as several QRishing-related works had already been published by 2019 and were not included (e.g., Guo et al. (2016), Petrova et al. (2016), Zulkefli et al. (2017) among several others). As such, while the attack techniques described by Yong et al. (2019) are represented in the taxonomy, a systematic literature review was still needed to provide a more complete and replicable collection of QRishing attack techniques.

Athanzah et al. (2026) aimed at three separate goals: (1) Identify various QRishing attack techniques, (2) identify the factors influencing QRishing susceptibility, and (3) Strategies to prevent QRishing. Of the three goals, the first one is the one related to the work presented in this chapter, especially because it was answered through a systematic literature review. The time interval was the same used in the taxonomy (2015-2025), and their methodology (shown in Figure 4.1) was also based on the PRISMA standard (described in Page et al. (2021)).

They collected 197 papers, reduced over several round of screening to 19 final studies, and eight of those 19 studies contained QRishing attack techniques. These were then used to generate

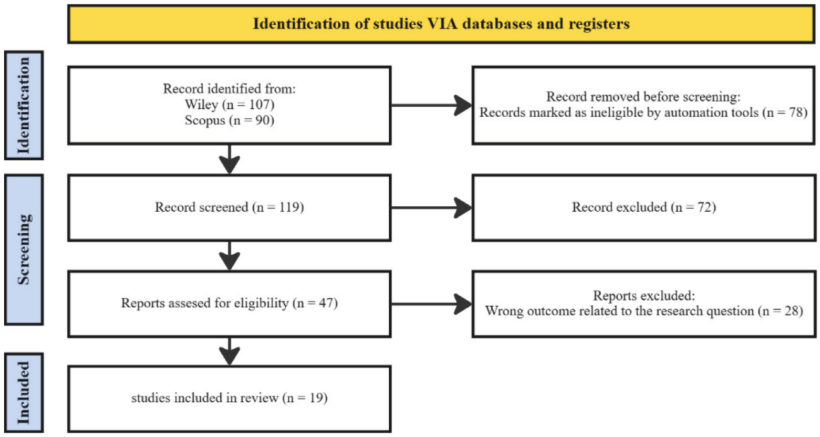


Figure 4.1: The PRISMA representation of the methodology from Athanzah et al. (2026).

their systematization, which identified seven QRishing attack techniques: *Barcode in Barcode*, *Cross Site Scripting* (XSS), *Man in the Middle*, *Social Engineering*, *SQL Injection*, *Malware Propagation*, *Embedding a Malicious Link.* Figure 4.2 shows the attack techniques identified and the number of papers that mentioned each technique. Each paper could mention more than one attack technique.

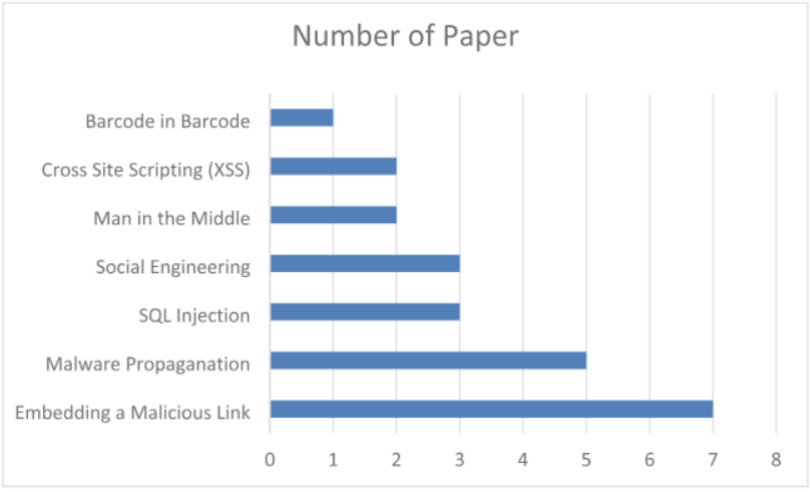


Figure 4.2: The attack techniques identified by Athanzah et al. (2026).

Regarding the methodology, both this work and Athanzah et al. (2026) relied on systematic reviews of the literature. However, the breadth of the source collection differed substantially. While

Athanzah et al. (2026) based their review on the Wiley Online Library and SCOPUS databases, the taxonomy presented in this chapter was developed using multiple academic databases (ACM Digital Library, IEEE Xplore, Elsevier Science Direct) and was further complemented by a systematic review of the MITRE ATT&CK® database. Due to the difference in the considered databases, Athanzah et al. (2026) identified two papers (Cargrill et al. (2023), Wahsheh and Al-Zahrani (2021)) that were not included in the original systematic literature review performed for this taxonomy. The analysis of these two further papers showed that they did not contain additional attack techniques, and therefore no changes to the taxonomy were required. Nonetheless, they were both added as further references.

Regarding the attack techniques themselves, the *Barcode in Barcode* attack technique corresponds to the *QR-in-QR* attack technique and is part of the *Altered* category in the taxonomy, while *Cross Site Scripting* and *SQL Injection* are both part of the *Injection attack* category. The *Malware Propagation* attack technique and the *Embedding a Malicious Link* attack technique are both covered by the *Malicious URL* category. The *Social Engineering* attack technique (e.g., creating time pressure) is not considered a QRishing attack technique in the taxonomy, as it represents the general approach underlying all social engineering attacks, independently of the specific attack vector (e.g., QR codes, SMS, voice). Finally, Athanzah et al. (2026) do not provide further details regarding the *Man in the Middle* attack technique, making it impossible to determine how it relates to the taxonomy. As such, while the attack techniques described by Athanzah et al. (2026) are represented in the taxonomy, the systematic literature review and MITRE ATT&CK® database review performed in this work was still needed to provide a comprehensive overview of QRishing attack techniques in the domain and context considered by QRNope.

4.1.2 Methodology

This section first covers the methodology applied to conduct the systematic literature review (Section 4.1.2.1), followed by the methodology to conduct the systematic industry database review (Section 4.1.2.2).

4.1.2.1 Systematic Literature Review

As mentioned in the introduction of this chapter, the *Context* stage is that of lay users scanning QR codes using iOS devices. However, academic literature is rarely tied to a specific operating system or QR code scanner, but rather describes how QR codes can be used to deliver attacks. Hence, an attack technique found in the literature might still target iOS devices even if it is not explicitly stated. Therefore, only considering sources explicitly mentioning iOS would risk

excluding relevant attack techniques. Consequently, the systematic literature review collected all QRishing attack techniques identified in the literature.

The systematic literature review followed the Preferred Reporting Items for Systematic reviews and Meta-Analyses (PRISMA) standard (described in Page et al. (2021)). Figure 4.3 contains an overview of the process followed and the outcome of each of its phases.

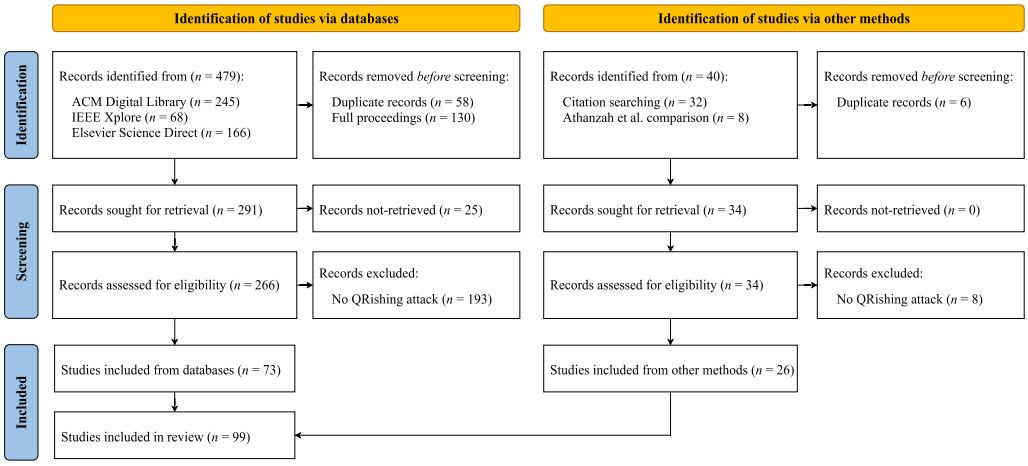


Figure 4.3: The process followed to collect the studies included in the systematic literature review.

The first step was the identification of databases known to contain works on QR codes, especially phishing: *ACM Digital Library*, *IEEE Xplore*, and *Elsevier Science Direct*. The search itself (performed through the built-in search function of each database) was constrained to works published in the time interval 2015–2025. The lower boundary of the time interval was set to include the ten-year period preceding the systematic literature review, while the upper boundary was set to when the systematic literature review was conducted, i.e., May 2025.

Each database was queried with the query *"QR" AND "phishing"*, and 479 entries were collected (ACM – 245, IEEE – 68, Elsevier – 166). After merging repeated results (58 entries), entries consisting of the full proceedings of a venue were excluded (130 entries). Before exclusion, each proceedings' website was opened, every track expanded, and Microsoft Edge's search function used with our query to find any paper containing those two words in either the title or the abstract. No further paper was found this way. Then, entries for which the full text could not be retrieved were excluded (25 entries). This left 266 entries to be checked in the second phase of the process.

The second phase of the PRISMA process is the screening phase, which commonly relies on titles and abstracts to exclude non-relevant entries. However, QRishing attack techniques might have only been a secondary product of the research presented in a paper, e.g., they were mentioned as

examples of what a machine learning algorithm detects, but they were not relevant to answer the research questions. In such cases, the QRishing attack techniques might not be found in either title or abstract, but still be discussed somewhere in the text. Thus, each of the 266 entries from Phase 1 was searched with the query keywords (*"QR" and "phishing"*) one at a time. When a passage of the text seemed to describe a QRishing attack technique, that passage was investigated further, e.g., by back-tracking to the beginning of the section and reading through it. In so doing, 73 relevant entries were identified and 193 non-relevant entries (i.e., entries not describing any QRishing attack technique) were excluded.

Finally, each entry was investigated further to identify if they referenced any previous source for the QRishing attack technique discussed. Those citations were then added to a pool of further entries (32 entries). The text of these further entries was then investigated with the same methodology described above, and 24 of them were found to be relevant (i.e., eight entries were deemed non-relevant). This left 97 entries (73 from the main search and 24 from the cited sources).

As an extra step, the sources collected by Athanzah et al. (2026) were compared with the sources collected through this methodology. Of the eight papers they collected that contained QRishing attack techniques, two (Cargrill et al. (2023) and Wahsheh and Al-Zahrani (2021)) were not among the 97 entries because they originated from databases not covered in this methodology. These two further entries were investigated, found relevant, and added to the 97 entries identified. This resulted in a final set of 99 entries.

4.1.2.2 Systematic MITRE ATT&CK® Techniques in the iOS Environment Review

MITRE ATT&CK® (described in MITRE (2026h)) is a repository of adversarial behaviors divided into two parts: *Enterprise*, covering attacks against businesses, and *Mobile*, focused on mobile devices. Differently from academic literature, MITRE ATT&CK® provides structured information regarding the platforms affected by each attack technique. As such, the *Context* stage and the *Security assumptions* stage could be directly considered during the systematic database review by focusing on attack techniques applicable to iOS devices and not already addressed by it.

The version of the MITRE ATT&CK® database used was version 19.1 from April 2026 (the latest one at May 2026, the time of the systematic review), available online at MITRE (2026z).

The focus was on the *Mobile* part of the database, which in version 19.1 contained 124 entries. Each one of the 124 entries was accessed through Mozilla Firefox, opened one by one, and stored locally as interactive snapshots in Zotero. The process followed is shown in Figure 4.4.

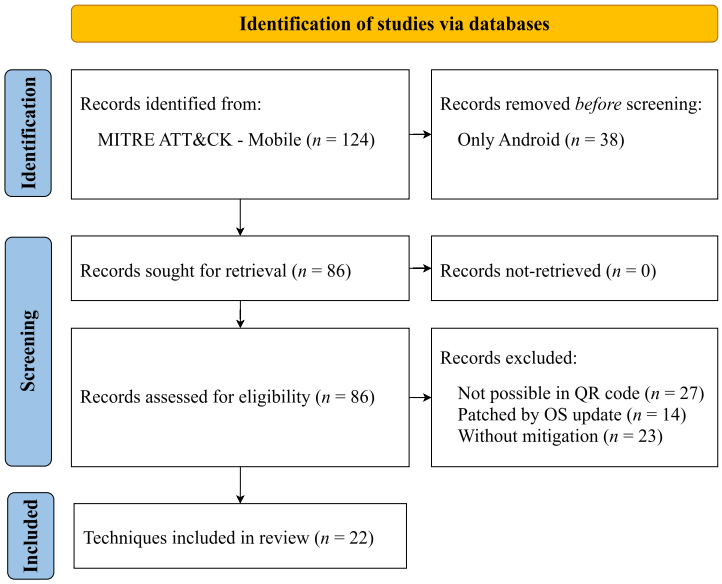


Figure 4.4: The process followed to identify the MITRE attack techniques included. Modified from Page et al. (2021).

As first round of screening, the initial 124 entries were screened into one of two groups: (1) *iOS applicable*, i.e., entries describing attack techniques that would work on iOS devices, and (2) *not iOS applicable*, i.e., entries describing attack techniques targeting solely Android devices or that would not work on iOS devices. The 38 entries labeled as "not iOS applicable" were screened-out, and the process moved to the next round of screening with the remaining 86 entries labeled as "iOS applicable".

The 86 entries labeled as "iOS applicable" were then screened into one of three groups: (1) *relevant*, i.e., entries related to QR codes or QR code technologies, (2) *not relevant*, i.e., entries unrelated to either QR codes or QR code technologies, and (3) *patched by an OS update*, i.e., entries already addressed by an operating system update. Regarding (2), ATT&CK® considers *every* mobile device attack technique, i.e., also attack techniques that are not related to QRishing. For example, attack technique T1451 "Sim Card Swap", albeit insidious, is not an attack technique delivered through or related to QR codes, and it was screened-out. After the second round of screening, 45 entries were labeled as "relevant", 28 entries as "not relevant", and 14 entries as "patched by an OS update."

In the third round of screening, the 45 entries labeled "relevant" were further screened into one of two groups: (1) *with mitigation*, i.e., entries for which a mitigation is present in the MITRE database, and (2) *without mitigation*, i.e., entries labeled as having no mitigation. After the third round of screening, 22 entries were labeled as "with mitigation", and 23 entries as "without

mitigation". The 23 entries without mitigation were excluded, resulting in a final set of 22 MITRE ATT&CK® entries included in the review.

4.1.2.3 Limitations

The systematic reviews had three potential limitations: (1) the time interval chosen, (2) the databases chosen, and 3) the query chosen.

Regarding the time interval, it might be that older works were left out. However, given that one of the related works is older than the lower-boundary of the time interval (i.e., Krombholz et al. (2014)), and that said work only contains a small number of attack techniques (all of which are part of the taxonomy), it is believed that older works would not have led to significant differences in the taxonomy.

About the academic databases chosen, it might be that other databases could have led to further sources. Nonetheless, based on (1) the number of citations for each attack technique, and (2) finding little known attack techniques (i.e., attack techniques with relatively low numbers of citations), it is believed that the taxonomy categorized, if not all, the vast majority of attack techniques. As such, further sources would have likely increased the number of references supporting the taxonomy, without significantly changing its structure. Furthermore, the MITRE database, albeit a single industry database, is the most renowned source for attacking techniques (as shown in Microsoft (2025)). As such, lesser known industry databases are unlikely to contain attack techniques not covered by the MITRE database.

On the query chosen, it was purposely kept as abstract as possible, because using a more specific query might have missed relevant works related to the use of QR codes as phishing vectors. Given the high number of entries, the query was deemed fitting to the task.

4.1.3 Results

The attack techniques identified through the systematic literature review and the MITRE ATT&CK® database review were organized into a taxonomy. This taxonomy was developed by structuring the attack techniques according to three questions: (1) *Which variety of QR code is encountered by users?*; (2) *How is the payload delivered?*; and (3) *What payload is delivered?*.

First, the attack techniques were categorized based on the variety of QR code encountered by users, i.e., whether the QR code is embedded in a pretext or encountered as a standalone QR code. Second, the attack techniques were grouped based on how the payload is delivered, i.e., whether this is done through the QR code itself, a malicious QR code generator, or a malicious QR code

scanner. Finally, the payloads delivered through these mechanisms were categorized according to what is delivered, i.e., whether they consist of a malicious URL, a malicious phone number, an injection attack, or a malicious QR code targeting a specific application.

The resulting taxonomy is shown in Figure 4.5.

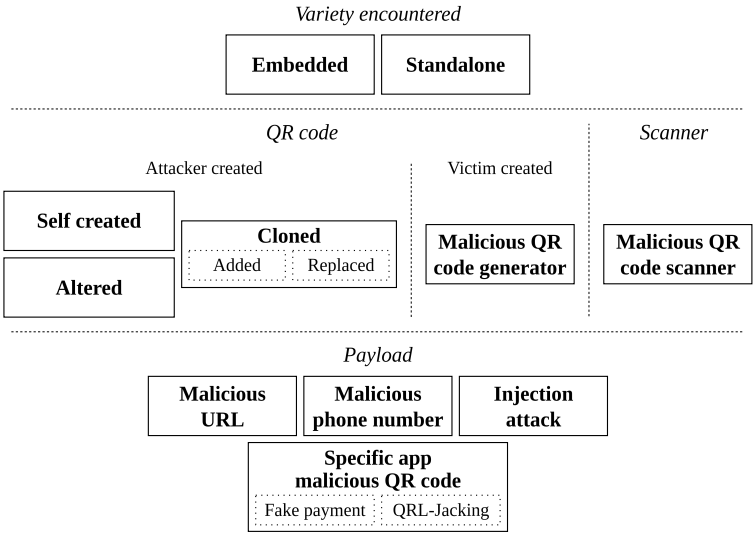


Figure 4.5: Visual representation of the taxonomy.

QR codes (both legitimate and malicious) are encountered in one of two varieties:

- ***Embedded*** in a pretext, e.g., a poster, a leaflet, an email, a social media post, etc. Found in Alabdan (2020), Bekavac et al. (2024), Bhardwaj et al. (2020), Focardi et al. (2018, 2019), Ford and Berry (2024), Guo et al. (2016), Gyimah et al. (2024), Jansen and Leukfeldt (2015), Krombholz et al. (2014, 2015), Liu et al. (2024), McPherson et al. (2015), Pawar et al. (2022), Petrova et al. (2016), Sarkar and Shukla (2023), Sharevski et al. (2022), Song et al. (2018), Thakur et al. (2024), Varshney et al. (2024b), Vidas et al. (2013), Wahsheh and Al-Zahrani (2021), MITRE (2026q)
- ***Standalone***, i.e., without any pretext, e.g., a sticker, on a sheet of paper, etc. Found in Averin and Zyulyarkina (2020), Gyimah et al. (2024), Kieseberg et al. (2012), Krombholz et al. (2014), McPherson et al. (2015), Pawar et al. (2022), Petrova et al. (2016), Sabri et al. (2023), Sravan et al. (2024), Vidas et al. (2013), Zhang et al. (2023)

These two varieties differ based on whether or not the QR code is surrounded by a pretext or not. This is an important distinction, as depending on it two distinct human factors are exploited to convince a user to scan the QR code Chapter 2 shows that a pretext attracts the attention of users by exploiting interesting layouts, or through topics that are dear to the users. This is, in general, not something that a *Standalone QR code* would do, instead, exploiting the curiosity of the victims, as shown in, e.g., Krombholz et al. (2015), Seeburger (2012). In other words, in the case of *Standalone QR codes* what attracts the users is the absence of a pretext.

Irrespective of the pretext presence, once an interaction is attracted a QRishing attack differs based on which part of the technology involved delivers the payload, i.e., either (1) the *QR code* itself, or (2) the *QR code scanner*.

Regarding the *QR code*, its origin can be traced to two sources: being *attacker created*, or being *victim created*. In case of an attacker created QR code, there are three ways this can happen:

- A ***Self-created*** QR code (and pretext, if there is one) are generated by the attackers without relying on existing legitimate versions. Found in Averin and Zyulyarkina (2020), Bekavac et al. (2024), Kieseberg et al. (2012), Krombholz et al. (2014), Mouton et al. (2016), Petrova et al. (2016), Song et al. (2018), Subairu et al. (2020).
- A preexisting QR code is found and ***Altered*** in anyone of three ways. One way is to modify the modules of a printed QR code with a pen to shown something else than what was originally intended, e.g., a different URL (likely, under a URL the attackers' control). A second way way is to add a smaller QR code inside the original one (also known as *QR-in-QR)* which would be given priority when scanned. A third way is to manipulate the lighting, e.g., by using infrared light (not visible by humans, but perceived by a camera), to alter the modules of the original, legitimate QR code. Found in: *Different URL* - Averin and Zyulyarkina (2020), Cargrill et al. (2023), Gyimah et al. (2024), Ismail et al. (2021), Kieseberg et al. (2010, 2012), Krombholz et al. (2014, 2015), Pawar et al. (2022), Rutherford and Wu (2023), Sharma (2012), Sravan et al. (2024), Varun and Rashmi (2023), Wang and Duncan (2019), Wang (2022), Yong et al. (2019); *QR-in-QR* - Bekavac et al. (2024), Cargrill et al. (2023), Chou and Wang (2020), Dabrowski et al. (2014), Gyimah et al. (2024), Krombholz et al. (2015), Rafsanjani et al. (2023), Sravan et al. (2024), Syed et al. (2022), Varun and Rashmi (2023), Wahsheh and Al-Zahrani (2021), Yong et al. (2019); Light manipulation - Zhou et al. (2019)
- A ***Cloned*** QR code is generated by modifying an existing legitimate pretext to contain a malicious QR code. If there was already a QR code, this is ***Replaced*** with the attackers' created QR code. If no QR code was present, the QR code of the attacker is ***Added***. Found in: Cloned - Aleroud and Zhou (2017), Bekavac et al. (2024), Bhardwaj et al. (2020),

Kieseberg et al. (2012); Replaced - Bekavac et al. (2024), Cargrill et al. (2023), Guan and Tick (2024), Krombholz et al. (2014), Petrova et al. (2016), Rutherford and Wu (2023), Sartayeva and Chan (2023), Sravan et al. (2024), Tu et al. (2021), Varshney et al. (2024b), Wang and Duncan (2019), Wang (2022), Yong et al. (2019), Zhang et al. (2023), MITRE (2026q).

For victims' generated QR codes, only one case was found, i.e., victims generating and disseminating phishing QR codes through ***Malicious QR code generators***. Such generators consistently substitute the victims' content with whatever payload the attackers want to disseminate. For example, a malicious generator could substitute the attackers' controlled URL to anything the victims' were trying to encode. QR codes generated this way exploit the victims' credibility to gain legitimacy, as the victims are the ones spreading them. Found in: Averin and Zyulyarkina (2020), Dudheria (2017), Froehlich et al. (2021), Pawar et al. (2022), Rafsanjani et al. (2023), Sabri et al. (2023)

Regarding the QR code scanner, the only case possible is for the attackers to develop and disseminate a ***Malicious QR code scanner***. Once installed, such malicious QR code scanners would then deliver one of three types of payload: (1) a trojan (Found in Gopinath and Sethuraman (2023), Kharraz et al. (2014), Lerner et al. (2015), Rafsanjani et al. (2023), Sharma (2012), Shen et al. (2018), MITRE (2026c,d,e,i,j,k,l)), (2) a permission escalation attack (Found in Bao et al. (2017), Focardi et al. (2018), Guan and Tick (2024), Gyimah et al. (2024), Kieseberg et al. (2010, 2012), Rafsanjani et al. (2023), Rottermanner et al. (2015), Shulzhenko and Romashkin (2020), Wahsheh and Al-Zahrani (2021), MITRE (2026i,r,s,t,u,v,w)), and (3) redirect any scanned QR code containing a URL to a different URL of the attackers' choosing (Found in Focardi et al. (2018, 2019), Froehlich et al. (2021), Kieseberg et al. (2010), Lerner et al. (2015), Rafsanjani et al. (2023)).

Regardless of whether the attackers use a QR code, a QR code generator, or a QR code scanner, one of four payloads can be delivered:

- A ***Malicious URL***, that can either lead to: (1) a webpage harvesting the credentials of the victims; or (2) disseminate malware. Such Malicious URLs often employ attack techniques to trick users into visiting them by hiding or misrepresenting the actual destination. Given that these techniques define how a malicious URL is presented to users, they are further detailed after the payload categories. Found in: Alabdan (2020), Aleroud and Zhou (2017), Barron and Sharma (2026), Bekavac et al. (2024), Bera et al. (2023), Bhardwaj et al. (2020, 2021), Cargrill et al. (2023), Dudheria (2017), Focardi et al. (2018, 2019), Ford and Berry (2024), Guo et al. (2016), Heartfield and Loukas (2018), Kharraz et al. (2014), Kieseberg et al. (2010, 2012), Krombholz et al. (2014), Lakmal et al. (2024), Lerner et al. (2015), Li

et al. (2020), Lin et al. (2019), Mouton et al. (2016), Niu et al. (2024), Petrova et al. (2016), Roy et al. (2024), Rutherford and Wu (2023), Sabri et al. (2023), Subairu et al. (2020), Syed et al. (2022), Thakur et al. (2024), Tu et al. (2021), Varun and Rashmi (2023), Vidas et al. (2013), Wahsheh and Al-Zahrani (2021), Wang and Duncan (2019), Xiao (2021), Yao and Shin (2013), Yong et al. (2019), Zulkefli et al. (2017), MITRE (2026b,f,g,o,p,q,x,y)

- A ***Malicious phone number*** that impersonates another entity, redirects all future phone calls to the attacker, or charges exorbitant tariffs when called. Similarly to *Malicious URLs*, several attack techniques can be used to hide or misrepresent the actual phone number encoded in the QR code. These attack techniques are further detailed after the payload categories. Found in: Lerner et al. (2015), Sravan et al. (2024), Yao and Shin (2013)
- An ***Injection attack*** that encodes lines of code in the QR code. Once scanned, the code disrupt the normal operation of the device. The attackers take control of the device and can steal the victims' information, e.g., contact lists, images, documents, etc. This can be done through an SQL injection or through Cross Site Scripting. Found in: Averin and Zyulyarkina (2020), Bao et al. (2017), Barron and Sharma (2026), Cargrill et al. (2023), Dabrowski et al. (2014), Focardi et al. (2018, 2019), Ford and Berry (2024), Ismail et al. (2021), Jin et al. (2014), Kieseberg et al. (2010, 2012), Krombholz et al. (2014, 2015), Niu et al. (2024), Rafsanjani et al. (2023), Rodriguez et al. (2019), Sartayeva and Chan (2023), Sharma (2012), Syed et al. (2022), Tu et al. (2021), Wahsheh and Luccio (2020), Wang and Duncan (2019), Yao and Shin (2013)
- A ***Specific app malicious QR code***, i.e., a malicious QR code aimed at specific apps, or type of apps. There are two types of such QR codes: (1) *Fake payment*, and (2) *QRL-Jacking*. The first trick the victim into sending money to the attackers (e.g., via PayPal). The second is used to steal the necessary credentials to hijack the online sessions of the app, which is then taken over by the attackers. Found in: Aleroud and Zhou (2017), Averin and Zyulyarkina (2020), Cargrill et al. (2023), Ford and Berry (2024), Jubur (2024), Krombholz et al. (2015), Lin et al. (2022), Manna (2022), Niu et al. (2024), Sharevski et al. (2022), Sravan et al. (2024), Syed et al. (2022), Tu et al. (2021), Varshney et al. (2018, 2024a), Wan and Jiang (2025), Yusop et al. (2025), Zhao (2023), MITRE (2026n)

While the previous categories describe the payload delivered through QRishing attacks, the *Malicious URL* and *Malicious Phone Number* categories require further refinement. Attackers can employ several attack techniques to hide or misrepresent the actual URL or phone number encoded in the QR code. Such techniques often leverage users' lack of understanding regarding the structure of a URL (shown in Figure 4.6), but they might also target phone numbers.

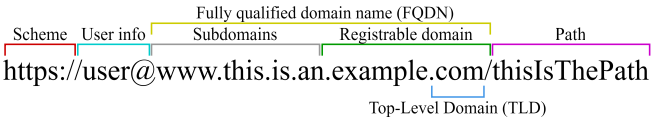


Figure 4.6: Structure of a URL and its parts.

Veit et al. (2024) presented a list of attack techniques identified in the context of phishing attacks. Since then, an updated version of this work was published in Veit et al. (2026a), which expands the previous list with several further attack techniques. The attack techniques in Veit et al. (2026a) are categorized based on three indicators available to users when evaluating phishing attempts: *Sender*, *Link*, and *Attachment*.

The attack techniques related to the *Link* indicator are of interest also for QRishing, given that QR codes can encode many of the same destinations commonly reached through links in emails. However, not all attack techniques related to the *Link* indicator are relevant in the QRishing context. For example, attack techniques based on HTML tags describe how a link is interpreted or rendered within a webpage or email, rather than a technique affecting the QR code or the payload encoded in it.

As such, the relevant attack techniques from Veit et al. (2026a) are used to further refine the *Malicious URL* category and the *Malicious Phone Number* category of the taxonomy presented in Section 4.1.3. Table 4.1 contains the resulting attack techniques, alongside a brief description and an example of each.

4.1.4 QRNope Input: *All Attack Techniques in the Domain*

This section first covers the process used to adapt the collected attack techniques to the domain and context of QRNope (Section 4.1.4.1), followed by the limitations of this process (Section 4.1.5).

4.1.4.1 Process to Adapt the Collected Attack Techniques to QRNope

The taxonomy presented in Section 4.1.3 represents the basis for the *All attack techniques in the domain* stage of QRNope. As described in the introduction of this chapter, the attack techniques considered for QRNope are limited by its domain and context. The influence of the *Context* stage was actualized through two aspects: the technology used and the context of use. Specifically, only attack techniques applicable to iOS devices and everyday QR code interactions were retained. Attack techniques outside of this scope (e.g., attack techniques requiring different operating systems or enterprise-specific contexts) are not added as security assumptions, as they are not

Table 4.1: Attack techniques from Veit et al. (2026a), with brief description and example.

Obfuscation technique	Description	Example
Malicious Phone Number		
Impersonation	Attackers use their phone number to impersonate another identity	`tel:12345678` (*Attackers' number*)
USSD Code Exploitation	Certain codes are placed before a number to obtain various effects	`tel:*21*00113371337#` (*Redirect all calls to the number after* `*21*`)
Premium Rate Service Exploitation	A phone number with high cost if used to charge victims' bills	`tel:090012345678`
Malicious URL		
Hexadecimal encoding	Domain characters are encoded in hexadecimal to hide the real domain	`https://google.demal.is` → `https://google.de%6D%61%6C%2E%69%73`
Mismatch		
Entire URL in Pretext	A legitimate URL is next to a phishing QR code to pretend it is encoded	*Pretext URL:* `https://google.com` *QR code:* `https://demal.is`
Part of the URL in Pretext	Same rationale as placing the entire URL, but only parts are used	*Pretext URL:* `google.com` *QR code:* `https://demal.is`
External Content Manipulation	Attackers controlled content is stored on reputable domains	`https://docs.google.com/document/d/` [...]
URL Path/Query/Fragment		
Domain in Path	The legitimate domain is placed in the path of the phishing URL	`https://demal.is/google.com/login`
Entire URL in Path	The entire legitimate URL is placed in the path of the phishing URL	`https://demal.is/https://google.com/login`
Luring-Label with Domain	Terms to simulate redirection to legitimate domain are used in the path	`https://demal.is/?redirect=secure-google.com`
Luring-Label with URL	Terms to simulate redirection to legitimate URL are used in the path	`https://demal.is/?redirect=https://secure-google.com`
Look-Alike Domain	The structure of known Short URLs is used to simulate redirection to legitimate URL	`https://bit.ly/4qWnYMj?redirect=https://google.com`
Omitting Slash	The slash to the path is omitted to simulate the legitimate domain	`https://customer-user.support#google.com`
UserInfo Field	The domain is placed as the UserInfo field	`https://google.com@demal.is`
URL Shortener	A Short URL is used to hide the actual destination	`https://bit.ly/4qWnYMj`
URL Redirect	A redirection is used to hide the actual destination in the path	`https://google.com/url?sa=t&source=web&rct=j&url=https://demal.is`
Homographic Spoofing	Letters from other alphabets are used to simulate a legitimate domain	`https://google.com` (*Cyrillic* e)
Unrecognizable domain		
IP Address	The URL uses the IP address	`https://142.250.74.206`
Random Characters	Random characters are used as domain	`https://aldslkskmskoewlc.at`
Use of Fitting Keywords	Terms fitting to the pretext are used to increase believability	`https://docs-share-online.com/presentation`
Domain Extension	The legitimate domain is extended with fitting words	`https://secure-google.com`
Subdomain	The legitimate domain is placed as subdomain	`https://google.com.demal.is`
Different Top-Level-Domain	The first part of the legitimate domain is used, but with a different ending	`https://google.io`
Mangle	Characters are swapped or substituted to trick inattentive readers	`https://gogle.com`
Exceedingly Long URL	A long URL is used to hide the domain outside the screen rendering area	`https://google.com.safe.account.login.user.support.email.docs.maps.security.demal.is/index.html`
Custom-Scheme and Deep-Links	A deep-link or a custom scheme is used to hide the actual destination	`msteamms://teams.microsoft.com/l/chat/` [...]

assumed to be addressed in some other way. Rather, they are outside the scope of QRNope and do not require addressing.

The technology constraint was applied by limiting the systematic MITRE ATT&CK® database review to iOS techniques from the Mobile matrix (as described in Section 4.1.2.2). Regarding the systematic literature review (covered in Section 4.1.2.1), the identified sources usually do not refer to specific operating systems. As such, attack techniques identified through academic sources are retained unless they explicitly contradict the defined context. For example, even if the *Malicious QR code scanner* attack technique is not mentioned in the MITRE ATT&CK® database, it is possible that a malicious scanner can be developed for iOS devices.

Furthermore, attack techniques related to the delivery of the QR code (e.g., *Altered*) are independent from the operating system used. Similarly, payload attack techniques are retained even if some specific implementations might only affect other operating systems. In such cases, the limitation concerns the specific implementation, not the attack technique itself.

After applying these considerations, the resulting remaining attack techniques are shown in Table 4.2.

Table 4.2: Collected attack techniques in the domain with examples or descriptions. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*.

Attack technique	**Description or Example**
Embedded	*MQRC is part of a pretext*
Standalone	*MQRC is not part of a pretext*
Self created	*MQRC is an original creation of attackers*
Altered	*Attackers modify an existing LQRC*
Cloned	
Added	*MQRC added to a pre-existing, QR code-less, legitimate pretext*
Replaced	*LQRC part of a pre-existing, legitimate pretext replaced with a MQRC*
Malicious QR code generator	*Malicious generator used to disseminate MQRCs for attackers*
Malicious QR code scanner	*Malicious app installed by users*
Malicious Phone Number	
Impersonation	`tel:12345678` (*Attackers' number*)
USSD Code Exploitation	`tel:*21*00113371337#` (*Redirect all calls to the number after* `*21*`)
Premium Rate Service Exploitation	`tel:090012345678`
Injection attack	`https://demal.is/login?user=admin' OR '1'='1`
Specific App MQRC	
Fake payment	*MQRC used to deliver a payment to the attackers*
QRL-Jacking	*MQRC used to hijack the victims' session token through QR code login*
Malicious URL	
Hexadecimal encoding	`https://google.demal.is` → `https://google.de%6D%61%6C%2E%69%73`
Mismatch	
Entire URL in Pretext	*Pretext URL:* `https://google.com` – *MQRC URL:* `https://demal.is`
Part of the URL in Pretext	*Pretext URL:* `google.com` – *MQRC URL:* `https://demal.is`
External Content Manipulation	`https://docs.google.com/document/d/` [...]
URL Path/Query/Fragment	
Domain in Path	`https://demal.is/google.com/login`
Entire URL in Path	`https://demal.is/https://google.com/login`
Luring-Label with Domain	`https://demal.is/?redirect=secure-google.com`
Luring-Label with URL	`https://demal.is/?redirect=https://secure-google.com`
Look-Alike Domain	`https://bit.ly/4qWnYMj?redirect=https://google.com/login`
Omitting Slash	`https://customer-user.support#google.com`
UserInfo Field	`https://google.com@demal.is`
URL Shortener	`https://bit.ly/4qWnYMj`
URL Redirect	`https://google.com/url?sa=t&source=web&rct=j&url=https://demal.is`
Homographic Spoofing	`https://google.com` (*Cyrillic* e)
Unrecognizable domain	
IP Address	`https://142.250.74.206`
Random Characters	`https://aldslkskmskoewlc.at`
Use of Fitting Keywords	`https://docs-share-online.com/presentation`
Domain Extension	`https://secure-google.com`
Subdomain	`https://google.com.demal.is`
Different Top-Level-Domain	`https://google.io`
Mangle	`https://gogle.com`
Exceedingly Long URL	`https://google.com.safe.account.login.user.support.email.docs.maps.security.demal.is/index.html`
Custom-Scheme and Deep-Links	`msteamms://teams.microsoft.com/l/chat/` [...]

A relevant aspect to consider at this point was that the attack techniques shown in Table 4.2 relied on the entire URL encoded in the QR code being shown to the users. However, Volkamer et al. (2016) proposed a URL visualization that omitted potentially misleading parts of URLs, while Mossano et al. (2023) further investigated the influence of such a visualization on phishing URL detection. Since the iOS Camera employed a similar URL visualization (i.e., only showing the domain and the top-level domain), it was possible that some of the attack techniques shown in Table 4.2 were already rendered ineffective by the information presented to the users.

To conduct this investigation, each of the attack techniques with an example in Table 4.2 was encoded in a QR code, exactly as presented in the "Description or Example" column. The specific examples were chosen because representative of the attack technique, and the domain `demal.is` was not used (i.e., leading to an empty webpage) at the time of the investigation (January 2026). There were two options to creating the QR codes to be scanned: (1) using an online QR code generator; and (2) create the QR codes through a Python script. Option 1 would have been easier (no script would be required), but it would have also impeded replication of the results in case the code behind the online QR code generator was changed at a later time or became unavailable. As such, option 2 was chosen. *PyCharm 2025.2*[2] was used to create a Python script in an environment that used *Python 3.12*[3]. Each QR code was created with the libraries *qrcode* 8.2[4] and *pillow* 12.1.0[5]. The QR codes were then exported as PNG files to be scanned. Once the QR code images were created, each one was printed and scanned through an iPhone SE 2020 with iOS 26.2. The results are reported in column "Shown in iOS Camera as" of Table 4.3.

As can be seen in column "Addressed by visualization" of Table 4.3, nine of the attack techniques are effectively thwarted by the URL visualization (those marked with the teal colored "Yes"). The decision regarding an attack technique being addressed or not was taken on an attack-by-attack basis and based on the omission of the confusing part of the URL. This means that, even if the way the attack technique is presented would show it as a different attack technique, one that is not addressed, the original attack technique would still be considered addressed. For example, the *Look-Alike Domain* attack technique is presented as a short URL service domain, meaning that it could be treated as a URL Shortener attack technique, which is not resolved by the URL visualization. However, because the part of the URL that makes a *Look-Alike Domain* attack technique such (the path) is not shown, the attack technique itself is addressed. Conversely, other attack techniques, e.g., the Mismatch ones, are not addressed, as the confusing part, e.g., the pretext URL, is not removed.

One of the attack techniques, i.e., Subdomain, is not entirely addressed, because a capable attacker might use very specific URLs to still deliver a Subdomain attack technique. For example, the URL `https://cnn.demal.is`, used to impersonate the CNN company (legitimate URL `https://cnn.com`), would be shown in the iOS Camera as `cnn.demal.is`. As such, a user might still be tricked by the attackers, as the `cnn` subdomain could still confuse them. Hence, although many other Subdomain attacks could be thwarted, the mere possibility of specific attacks targeting iOS to be successful means that it cannot be considered addressed.

[2] https://www.jetbrains.com/pycharm/
[3] https://www.python.org/downloads/release/python-3120/
[4] https://github.com/lincolnloop/python-qrcode
[5] https://zenodo.org/records/18128692

Table 4.3: Collected attack techniques in the domain, their examples or descriptions, how iOS Camera show them, and if the URL visualization addresses them. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*. The last two columns show how the iOS Camera would show the various attack techniques in its interface.

Attack technique	**Description or Example**	**Shown in iOS Camera as** (if Example)	**Addressed by visualization**
Embedded	*MQRC is part of a pretext*	-	No
Standalone	*MQRC is not part of a pretext*	-	No
Self created	*MQRC is an original creation of attackers*	-	No
Altered	*Attackers modify an existing LQRC*	-	No
Cloned			
Added	*MQRC added to a pre-existing, QR code-less, legitimate pretext*	-	No
Replaced	*LQRC part of a pre-existing, legitimate pretext replaced with a MQRC*	-	No
Malicious QR code generator	*Malicious generator used to disseminate MQRCs for attackers*	-	No
Malicious QR code scanner	*Malicious app installed by users*	-	No
Malicious Phone Number			
Impersonation	`tel:12345678` (*Attackers' number*)	`Call 0012345678`	No
USSD Code Exploitation	`tel:*21*00113371337#` (*Redirect all calls to the number after* `*21*`)	`Call *21*00113371337#`	No
Premium Rate Service Exploitation	`tel:090012345678`	`Call 090012345678`	No
Injection attack	`https://demal.is/login?user=admin' OR '1'='1`	`demail.is`*	No*
Specific App MQRC			
Fake payment	*MQRC used to deliver a payment to the attackers*	-	No
QRL-Jacking	*MQRC used to hijack the victims' session token through QR code login*	-	No
Malicious URL			
Hexadecimal encoding	`https://google.demal.is` → `https://google.de%6D%61%6C%2E%69%73`	`demal.is`	Yes
Mismatch			
Entire URL in Pretext	*Pretext URL:* `https://google.com` – *MQRC URL:* `https://demal.is`	`demal.is`	No
Part of the URL in Pretext	*Pretext URL:* `google.com` – *MQRC URL:* `https://demal.is`	`demal.is`	No
External Content Manipulation	`https://docs.google.com/document/d/` [...]	`docs.google.com`	No
URL Path/Query/Fragment			
Domain in Path	`https://demal.is/google.com/login`	`demal.is`	Yes
Entire URL in Path	`https://demal.is/https://google.com/login`	`demal.is`	Yes
Luring-Label with Domain	`https://demal.is/?redirect=secure-google.com`	`demal.is`	Yes
Luring-Label with URL	`https://demal.is/?redirect=https://secure-google.com`	`demal.is`	Yes
Look-Alike Domain	`https://bit.ly/4qWnYMj?redirect=https://google.com/login`	`bit.ly`	Yes
Omitting Slash	`https://customer-user.support#google.com`	`customer-user.support`	Yes
UserInfo Field	`https://google.com@demal.is`	`demal.is`	Yes
URL Shortener	`https://bit.ly/4qWnYMj`	`bit.ly`	No
URL Redirect	`https://google.com/url?sa=t&source=web&rct=j&url=https://demal.is`	`google.com`	No
Homographic Spoofing	`https://google.com` (*Cyrillic* `e`)	`xn-googl-3we.com`	Yes
Unrecognizable domain			
IP Address	`https://142.250.74.206`	`142.250.74.206`	No
Random Characters	`https://aldslkskmskoewlc.at`	`aldslkskmskoewlc.at`	No
Use of Fitting Keywords	`https://docs-share-online.com/presentation`	`docs-share-online.com`	No
Domain Extension	`https://secure-google.com`	`secure-google.com`	No
Subdomain	`https://google.com.demal.is`	`com.demal.is`	Yes
	`https://cnn.com.is`	`cnn.com.is`	No
Different Top-Level-Domain	`https://google.io`	`google.io`	No
Mangle	`https://gogle.com`	`gogle.com`	No
Exceedingly Long URL	`https://google.com.safe.account.login.user.support.email.docs.maps.security.demal.is/index.html`	`demal.is`	Yes
Custom-Scheme and Deep-Links	`msteamms://teams.microsoft.com/l/chat/` [...]	*No app* – `No usable data found` *App installed – Opens app*	No No

Another one of the attack techniques, i.e., Injection attack, was also not thwarted. The relevant part of this attack technique is that the commands given to the SQL database used in the investigation is in the path. As such, a user inspecting only the domain and top-level-domain would not notice that the path does something unexpected, and would effectively believe that the related website is safe to visit, when in reality it is not. Hence, even if the confusing part is hidden, this attack technique cannot be considered addressed by the visualization, as the confusing part is the main way to detect the attack technique in the first place.

Following this investigation, the final input for the *All attack techniques in the domain* stage of QRNope is shown in Table 4.4.

Table 4.4: The final list of attack techniques in the domain. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*.

Attack technique	Description or Example
Embedded	*MQRC is part of a pretext*
Standalone	*MQRC is not part of a pretext*
Self created	*MQRC is an original creation of attackers*
Altered	*Attackers modify an existing LQRC*
Cloned	
Added	*MQRC added to a pre-existing, QR code-less, legitimate pretext*
Replaced	*LQRC part of a pre-existing, legitimate pretext replaced with a MQRC*
Malicious QR code generator	*Malicious generator used to disseminate MQRCs for attackers*
Malicious QR code scanner	*Malicious app installed by users*
Malicious Phone Number	
Impersonation	`tel:12345678` (*Attackers' number*)
USSD Code Exploitation	`tel:*21*00113371337#` (*Redirect all calls to the number after* `*21*`)
Premium Rate Service Exploitation	`tel:090012345678`
Injection attack	`https://demal.is/login?user=admin' OR '1'='1`
Specific App MQRC	
Fake payment	*MQRC used to deliver a payment to the attackers*
QRL-Jacking	*MQRC used to hijack the victims' session token through QR code login*
Malicious URL	
Mismatch	
Entire URL in Pretext	*Pretext URL:* `https://google.com` – *MQRC URL:* `https://demal.is`
Part of the URL in Pretext	*Pretext URL:* `google.com` – *MQRC URL:* `https://demal.is`
External Content Manipulation	`https://docs.google.com/document/d/` [...]
URL Shortener	`https://bit.ly/4qWnYMj`
URL Redirect	`https://google.com/url?sa=t&source=web&rct=j&url=https://demal.is`
Unrecognizable domain	
IP Address	`https://142.250.74.206`
Random Characters	`https://aldslkskmskoewlc.at`
Use of Fitting Keywords	`https://docs-share-online.com/presentation`
Domain Extension	`https://secure-google.com`
Subdomain	`https://cnn.com.is`
Different Top-Level-Domain	`https://google.io`
Mangle	`https://gogle.com`
Custom-Scheme and Deep-Links	`msteamms://teams.microsoft.com/l/chat/` [...]

4.1.5 Limitations

The investigation of how the iOS Camera visualizes QR code destinations was conducted using the iOS Camera shipped with iOS 26.2 on an Apple iPhone SE (2020). Consequently, the results are limited to this specific hardware and software configuration. Different versions of iOS, other operating systems, or different QR code scanner applications may visualize QR code destinations differently, potentially affecting whether some attack techniques are considered addressed by the URL visualization.

Furthermore, the investigation relied on one representative example for each attack technique. Although the examples were selected to illustrate the corresponding attack techniques, different

examples may lead to different visualizations and, consequently, different conclusions regarding whether a particular attack technique is addressed by the URL visualization.

4.2 Misconceptions of Users about QRishing

The second input required for the development of QRNope is provided by the *Misconceptions on attack techniques* stage and the *Misconceptions on mitigation strategies* stage. As described in the introduction of this chapter, these stages require collecting the misconceptions held by the target population regarding the attack techniques and mitigation strategies considered for the awareness intervention.

For QRNope, this required identifying misconceptions that lay users have regarding QRishing attacks and the corresponding mitigation strategies. However, even though previous works already investigated misconceptions in security contexts (e.g., Neil et al. (2025), Herbert et al. (2023), Mayer and Volkamer (2018)), to the best of knowledge no work investigated misconceptions about QRishing. This gap is to be expected, to an extent, given the relative novelty of QRishing research. However, this novelty also causes a secondary problem: if an investigation of QRishing misconceptions was conducted, it might identify only a limited number of them, because the lack of awareness interventions on QRishing and the limited time since widespread use of QR codes might not have allowed misconceptions to emerge among users.

A solution would be to investigate previous works on misconceptions about phishing. As research on phishing is more extensive than that on QRishing, and because QRishing is a type of phishing, a larger number of misconceptions might be identified than by focusing on QRishing alone, and these misconceptions could still be relevant for QRishing. However, even if this line of research seemed promising, to the best of the present knowledge no work exists on misconceptions about phishing either. Nevertheless, given the potential of first conducting an investigation of existing misconceptions about phishing and then determining which ones are relevant for QRishing, this line of research is pursued.

Therefore, as a first step, a systematic literature review was conducted to identify and categorize misconceptions about phishing, providing a structured overview of misconceptions about phishing. These misconceptions were then further analyzed to determine which ones were also relevant for QRishing, and how they should have been categorized with regard to the *Misconceptions on attack techniques* stage and the *Misconceptions on mitigation strategies* stage.

This section describes the process followed to collect and categorize these misconceptions. First, the related work on the collection of cybersecurity misconceptions is discussed (Section 4.2.1). Then, the methodology used to conduct the systematic literature review is presented

(Section 4.2.2). Afterwards, the collected misconceptions are presented and categorized (Section 4.2.4). Finally, the misconceptions selected as input for the development of QRNope are discussed in Section 4.2.5.

4.2.1 Related Work

This section focuses on previous works that collected misconceptions in cybersecurity contexts. Given the lack of works systematically collecting misconceptions about phishing or QRishing, related work from other cybersecurity domains was considered. The goal of this section is not to provide an exhaustive overview of all misconceptions identified in cybersecurity, but to discuss previous approaches used to collect and categorize users' misconceptions.

Neil et al. (2025) aimed at investigating how non-expert users understand cybersecurity concepts, and identifying the misconceptions present in their mental models. To achieve this goal, they conducted a series of semi-structured interviews with 30 non-expert participants to investigate their understanding of cybersecurity definitions and concepts (e.g., what cybersecurity is, what an anti-virus is, what malware is, etc.). Their participants were first asked to explain the concepts in their own words, allowing the authors to identify differences between the participants' understanding and the actual definitions. Their results showed that participants often conceptualized cybersecurity as something outside of the users' role, i.e., something that users are subjects of and have no participation in. As such, Neil et al. (2025) highlighted the importance of addressing users' misconceptions, as awareness interventions should not only provide correct information, but also address the underlying mental models users rely on. However, their work focused on general cybersecurity concepts rather than phishing specifically. As such, misconceptions regarding specific attack techniques and mitigation strategies, such as those related to phishing, were outside the scope of their investigation.

Herbert et al. (2023) aimed at investigating whether users hold widespread misconceptions regarding common security and privacy concepts, and whether these misconceptions differ across countries. To achieve this goal, they conducted a study with over 12000 participants from 12 different countries and four continents. Their study focused on eight pre-determined topics, i.e., *End-to-End Encrypted Messengers*, *HTTPS*, *Wi-Fi*, *VPN*, *Passwords and Login*, *Device Security*, *Malware and Deception*, and *Private Browsing Mode*. Their results showed that users across countries were often uncertain regarding several fundamental security and privacy concepts, although the prevalence of specific misconceptions differed depending on the country considered. For example, the misconception regarding the need to frequently change passwords was present almost everywhere, with only one country (Saudi Arabia) being borderline on this belief, i.e., the agreement with the wrong statement was 3.96 on a 5-points Likert scale (5 being full agreement),

and the minimum for a misconception to be considered widespread was a median agreement of 4. Regarding the phishing-adjacent topic of HTTPS, they found that the misconception that HTTPS indicates a trustworthy website was not as widespread as the password change misconception, but the agreement regarding this misconception was still always higher than 3.5 on a 5-points Likert scale (with 5 being full agreement). Their work shows that misconceptions are not isolated misunderstandings, but can be widespread across different populations and cybersecurity topics. However, although the *Malware and Deception* category included aspects related to phishing, their work focused on broader cybersecurity concepts rather than misconceptions specific to phishing.

Both Neil et al. (2025) and Herbert et al. (2023) demonstrate the importance of identifying and addressing users' misconceptions in cybersecurity. However, both works relied on collecting new data from users to identify misconceptions in specific contexts. While this approach provides valuable insights into the target population studied, it does not address the complementary problem of collecting and systematizing misconceptions already identified across previous studies.

To the best of the present knowledge, the only other work that addressed this problem is Mayer and Volkamer (2018). Their work focused on misconceptions about password security and had two main goals: (1) systematically collecting misconceptions already reported in the academic literature, and (2) using the resulting overview as the basis for developing awareness interventions addressing them.

To achieve this goal, Mayer and Volkamer (2018) conducted a systematic literature review of publications discussing password security misconceptions. The authors searched academic databases and relevant venues using combinations of the term *password* with terms related to misconceptions, such as *misconception*, *misunderstanding*, and *misperception*. The resulting publications were filtered to identify those reporting misconceptions about password security. Furthermore, they expanded the search through forward and backward searches. This process resulted in the identification of 23 misconceptions, which were then categorized based on the aspect of password security they concerned.

Differently from the other works discussed in this section, Mayer and Volkamer (2018) did not aim at discovering new misconceptions through a user study. Instead, their goal was to consolidate existing knowledge scattered throughout previous literature. This distinction is important because misconceptions can appear as secondary findings of studies with different primary goals, meaning that individual empirical studies risk only capturing a subset of the misconceptions already known in the literature.

While the work of Mayer and Volkamer (2018) focused on passwords rather than phishing or QRishing, its underlying objective is closely aligned with the goal of this section. Namely, both aim at collecting and structuring existing knowledge on users' misconceptions in a security domain

to support the development of awareness interventions. As such, their work represents the closest methodological foundation for the systematic literature review presented in Section 4.2.2.

4.2.2 Methodology

As discussed in the introduction of this chapter, no previous work investigated misconceptions about QRishing. However, focusing only on QRishing-specific misconceptions might provide a limited overview due to the relative novelty of QRishing research. As phishing has been investigated more extensively, and misconceptions about phishing might also be relevant for QRishing, considering works on phishing misconceptions might provide a more complete overview. However, to the best of the present knowledge, no such work exists. As such, it was decided to first address this gap.

As discussed in Section 4.2.1, Mayer and Volkamer (2018) aimed at a similar goal within the password context and systematically collected misconceptions reported in the academic literature. Therefore, it was determined that a systematic literature review was also suitable for collecting misconceptions about phishing.

Like the systematic literature review in Section 4.1.2.1, also this one followed the PRISMA standard (described in Page et al. (2021)). Figure 4.7 contains an overview of the process followed and the outcome of each of its phases.

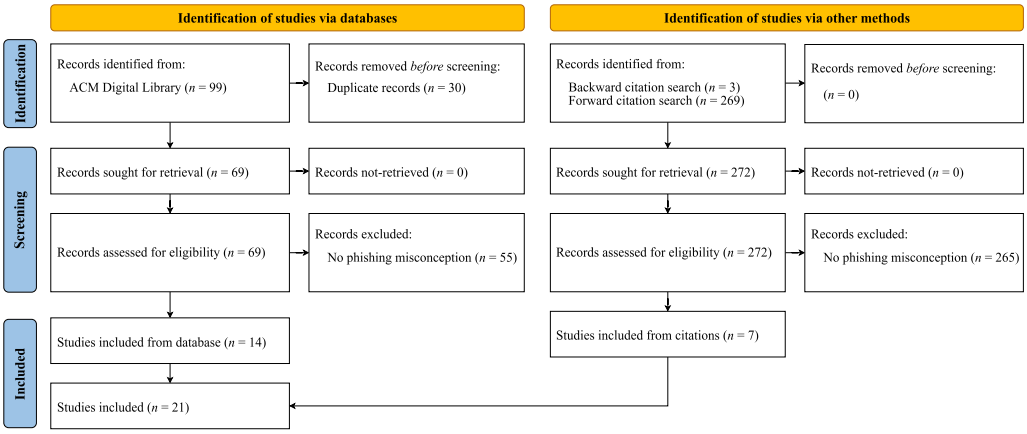


Figure 4.7: The process followed to collect the studies included in the systematic literature review.

As, to the best of the present knowledge, no previous work attempted to specifically identify and collect phishing misconceptions, it was determined that the most likely way to identify them

was to look for misconceptions that appeared as secondary products of the research presented in a paper. Furthermore, to ensure the quality of works considered, the focus was set on the proceedings of the ACM Conference on Human-Factors in Computer Systems (hereafter, *CHI*). CHI was specifically selected as it is both an A* ranked conference (as shown in CORE (2026)) and has a dedicated track on usable security. Thus, works published as part of said track are likely to be both related to phishing and of high quality. As such, the ACM Digital Library database was used, with the constraint of checking the last ten years at the time of investigation, i.e., 2013-2023.

The keywords were defined through a similar method as the one from Mayer and Volkamer (2018), itself based on the process for systematic literature reviews described in Kitchenham (2004). Namely, from two initial terms (*phishing* and *misconception*), other keywords were deduced from strong synonyms. This led to the creation of two sets of keywords, one for each starting term. The first set was formed by the terms: *phish**, *malicious message*, *fraudulent message*. The second set was formed by the terms: *misconception*, *misunderstanding*, *misperception*, *flawed perception*, *flawed understanding*. Every term from the first set was then paired with each of the terms in the second set to form the queries used. For example, the term *phish** was used for the queries: *phish** AND *misconception*, *phish** AND *misunderstanding*, *phish** AND *misperception*, *phish** AND *"flawed understanding"*, and *phish** AND *"flawed perception"*.

The ACM database was queried with each keyword pair one at a time, and the results stored in ReadCube's Paper on MacOS Sonoma 14.4. If the entire proceedings of a certain CHI edition were returned as results, the proceedings' webpage was opened in Microsoft Edge, each one of the tracks expanded, and the browser's search function used with each of the keywords one at a time. In case this search led to some papers being highlighted, those papers were added to ReadCube's Papers instead of the full proceedings, which were then removed. Across all keyword combinations, the search returned 99 results. After repeated results were merged, a total of 69 unique papers remained.

The whole text of the 69 papers selected was then investigated to determine if any misconception about phishing was reported. Namely, the same process used for the attack techniques literature review was employed, and the text of each paper was searched with each one of the keywords from both sets. When a passage seemed relevant, the section containing that passage was then investigated further. In doing so, 14 papers were deemed relevant, i.e., containing at least one misconception about phishing, and 55 not-relevant, i.e., not containing any misconception about phishing.

The search was then expanded to other venues through a backward and forward search based on the 14 relevant papers from CHI. Three papers were identified in the backward search, and these were investigated using the same methodology described above, but none were found relevant. In the forward search, the ACM Digital Library database was used to identify papers that cited the

14 relevant CHI ones, finding 269 further papers that cited at least one of the 14 relevant ones. These were collected and then investigated with the same methodology described above, leading to seven further relevant papers. As such, in total 21 papers were deemed relevant.

4.2.3 Limitations

The most straightforward limitation is that the methodology started by only considering papers published in CHI between 2013 and 2023. This choice was dictated by CHI being an A* ranked conference in the field of usable security; furthermore, the final selection contained works from other venues through a backward and forward search. Nonetheless, relevant papers might have been missed due to not being cited or not referencing the initial 14 CHI papers.

A second limitation is that the choice of keywords might have caused some papers to be missed due to the use of different terminology when discussing misconceptions. This necessity of considering several synonyms was dictated by the lack of a standard terminology for misconceptions. Furthermore, not every synonym of "phishing" and "misconception" was used. Nonetheless, as the synonyms used were the strongest ones, the review still gives a good overview of those papers that considered misconceptions about phishing.

4.2.4 Results

This section lists and describes each of the misconceptions found in the 21 papers selected through the systematic literature review described in Section 4.2.2. Namely, 15 misconceptions about phishing were identified and collected from the 21 papers included. The misconceptions are not limited to any specific delivery vector (e.g., not only emails or QR codes); rather, they cover several aspects related to phishing overall (e.g., voice phishing, emails, websites, etc.). Some papers reported more than one misconception, and some misconceptions were reported by more than one paper.

Note: In the misconceptions, terms such as *safe* and *trustworthy* are used instead of the technically more accurate term *secure* to reflect the terminology reported in the studies from which the misconceptions were extracted. While these terms are not equivalent from a security perspective, the original terminology is preserved to accurately represent the wording used by participants and reported in the original studies.

4.2.4.1 Misc. 1 – Strong passwords are enough to protect from phishing

Some participants[6] believe that strong passwords are sufficient to protect them from phishing attacks. Albeit using a strong password is an important security practice, it appears to be considered a solution against every security threat. Unfortunately, this is a misconception, as a strong password is only effective if it is a secret to be guessed by an attacker. However, a common phishing tactic is to trick users into revealing their credentials, strong passwords included. As such, even a strong password would be ineffective if an attacker could simply enter it without guessing it.

Found in: Shay et al. (2014).

4.2.4.2 Misc. 2 – If a website is known to the user or looks official it is safe to enter credentials

Several participants trust familiar-looking websites, in particular the ones with official-looking layouts, considering them to always be safe to enter their credentials. This misconception appears to originate from a misunderstanding regarding websites creation, namely, from not knowing that both layout and logos can simply be copied by an attacker. Users unaware of this possibility are more likely to be tricked into revealing their credentials. As such, considering how trivial it is for an attacker to "clone" a website (or any other pretext) and make it look official, this misconception is especially serious.

Found in: Reeder et al. (2018), Herbert et al. (2023), Sarker et al. (2024).

4.2.4.3 Misc. 3 – Clicking on a dangerous link always infects the device

Misconception 3 is different from the previous misconceptions, in the sense that it does not make a user reckless, but rather leads to excessive caution. The participants think that the mere act of clicking on a link will lead to the infection of the device used to access the website visited through the link. However, clicking a link rarely causes the automatic download and installation of malware. There have been cases where vulnerabilities were exploited to install malware without the users' input (as shown in Norton (2025)), but these were restricted to high-value targets due to the considerable development costs behind the attack. Nevertheless, suggesting to the users that only high-value targets should be concerned might generate misconceptions similar to M.23

[6] The term "participants" refers to the individuals where a misconception was found in one of the studies, while the term "users" is used as stated in Definitions and Terminology

from Mayer and Volkamer (2018), i.e., "It is possible to be too unimportant to be targeted". This would be a very dangerous outcome because users who believe themselves not important enough might start acting recklessly. Furthermore, the importance of a target depends on the goal of the attackers: a low-level employee might be a high-value target if the goal is getting access to the company network. As such, albeit generating unnecessary fear about clicking a link should be avoided, it is important to clarify that users should always evaluate links before accessing the corresponding content.

Found in: Herbert et al. (2023).

4.2.4.4 Misc. 4 – There is no difference between the text of a link and its URL

Participants confuse the anchor of a link (e.g., the text used) with the URL behind it. This misconception leads to various insecure practices, such as copy-pasting the link text in a browser engine to check its legitimacy and, once it leads the participants to the expected legitimate website, click on the original link, effectively reaching a different website. As such, by relying on this misunderstanding of how links and URLs work, and by not knowing that the attackers can use any anchor text for their link, users expose themselves to danger.

Found in: Distler (2023).

4.2.4.5 Misc. 5 – If a URL shows HTTPS it means that it is safe to click on the link

Misconception 5 derives from the participants' misunderstanding of what the "S" in HTTPS stands for. Many participants consider it as signaling that the link can be clicked without concern, as the website connected to it is secure. In reality, HTTPS merely means that the *connection* is secure, i.e., encrypted. Unfortunately, phishing websites can use encrypted connections just as well as legitimate websites. As such, users might wrongly assume that a secure connection is a sign of identity, and disclose their credentials to the attackers.

Found in: Reynolds et al. (2020), Herbert et al. (2023), Krombholz et al. (2019), Seamons (2022), Hadan et al. (2021), Okuda et al. (2021), Ruoti et al. (2017).

4.2.4.6 Misc. 6 – If a familiar name is in the URL then the URL is safe

Misconception 6 stems from the participants misunderstanding the structure of a URL (shown in Figure 4.6 in Section 4.1.3). Participants believe that if the name of the expected entity is shown, then the URL is legitimate, regardless of the position within the URL structure where

the name appears. However, this is not the case, as only the registered domain can identify a URL destination. As such, this lack of awareness regarding the importance of positioning within a URL structure exposes users to risks.

Found in: Albakry et al. (2020).

4.2.4.7 Misc. 7 – If the sender of a message is someone from the social circle of the user, the message is trustworthy

Misconception 7 comes from the belief that, as long as the sender of a message is a member of the participants' social circle, then the message is trustworthy. However, this is truly an unfounded belief, as senders can be spoofed regardless of the platform considered (as shown in, e.g., Bilge et al. (2009) for social networks, Mustafa et al. (2014) for phone numbers, Opazo et al. (2017) for email addresses). As such, the sender information is a proof of neither legitimacy nor identity, and by relying on the (wrong) notion that friends' and family addresses are safe, users expose themselves to risks.

Found in: Ahmad and Terzis (2022).

4.2.4.8 Misc. 8 – Opening attachments from seemingly legitimate emails without further inquiry is not dangerous

Misconception 8 stems from a similar basis as Misconception 2: the participants believe that, if the layout and sender of an email look legitimate, it is safe to open any attachment without checking. Just as for Misconception 2, also in this case users do not understand that an email layout can be copied. However, in addition to this, Misconception 8 is also based on the false notion that senders are a reliable indicator of legitimacy (as mentioned already in Misconception 7). As such, trusting the legitimate appearance of the layout and the sender address would cause users to base their choice of opening an attachment on no security-significant indicator, exposing them to risks.

Found in: Zou et al. (2020).

4.2.4.9 Misc. 9 – SPAM and phishing messages are the same

l

Misconception 9 is rooted in the confusing terminology used by both experts and awareness interventions. The National Institute of Standards and Technology (NIST) defines *SPAM* as unsolicited bulk commercial messages (as shown in NIST (2026b)). Nonetheless, several participants conflate SPAM with phishing by considering SPAM messages as malicious messages aimed at stealing information, spreading malware, or conducting scams. This terminological conflation becomes problematic when users assume that the mitigation strategies for SPAM and phishing are equivalent. For example, users might believe that marking a phishing message as SPAM is the same as reporting a phishing attack. However, while marking a message as SPAM can help filter unwanted commercial messages, dedicated channels exist for reporting phishing attacks (e.g., Report Fraud (2026) in the UK, or its homonymous in the US at FTC (2026)). As reporting phishing attacks is an important part of anti-phishing efforts, this misconception can negatively affect not only individual users but the broader user base, because dedicated phishing filters and automated defenses would not be trained using phishing messages reported only as SPAM.

Found in: Rader and Munasinghe (2019).

4.2.4.10 Misc. 10 – Technical solutions (e.g., filters, anti-virus) always protect from phishing

A widespread misconception (somewhat related to Misc. 9) is that technical solutions, be them anti-viruses, email filters, or full security suites (e.g., Mimecast (2026)) can block 100% of adversarial attacks. This misconception leads participants to believe that they can safely visit any website, as the technical solutions will protect them. Albeit partially true, given that older attack strategies and known phishing websites can be blocked by technical solutions (e.g., email filters or browser filters), unknown attack techniques would not be stopped until the filters properly categorize them as such. Furthermore, complex tactics[7] like *Lateral Movement*, i.e., trying to move through a compromised network to deliver further attacks (as described in, e.g., MITRE (2026m)), are not easily prevented through technical solutions. As such, misconceiving technical solutions as the "end-all", the "perfect" mitigation strategy exposes users (and companies) to severe risks.

Found in: Sharif et al. (2019), Reynolds et al. (2020).

[7] The term "tactic" identifies the reason why an attacker performs certain actions, the goal that informs their decisions, as described in MITRE (2026h).

4.2.4.11 Misc. 11 – Cybersecurity warnings always identify phishing or anyway dangerous messages

Misconception 11 originates from the same full trust in technical solutions as Misconception 10, but leads to a different outcome, i.e., extreme caution, instead of lack of it (similarly to Misconception 3). Namely, participants were shown as interpreting every warning given by an email client as meaning that an email was a phishing one. However, a warning might have several levels of severity (some of which might just be informational, as proposed by, e.g., Berens et al. (2026)) or some warnings might simply appear due to false positive triggers. For example, redistributing a digitally signed email through a mailing list breaks the digital signature due to the different sender address, causing the email to be (wrongfully) deemed an adversary-in-the-middle attack (described in MITRE (2026a)). As such, misconceiving every warning as the product of an infallible security check might lead to important and legitimate emails being ignored, potentially causing damages due to this (unwitting) inattention.

Found in: Jeong and Chiasson (2020).

4.2.4.12 Misc. 12 – If a Voice Over IP call shows a registered Caller ID, the identity of the caller is also verified

Misconception 12 is a relatively niche case related to *vishing*, i.e., phishing attacks conducted through voice calls (described in, e.g., Griffin and Rackley (2008)). The misconception originates from users misunderstanding the meaning of caller verification mechanisms, such as those based on the STIR/SHAKEN framework (described in FCC (2024)). These mechanisms aim to prevent attackers from spoofing legitimate caller information by verifying the owner of a *Voice Over IP* number. However, this verification only indicates that the number and its owner are verified; it does not indicate that the actual caller is the owner, nor that the caller's identity has been verified. In other words, the caller and the owner can be two different entities, with the caller not even being human. As such, users might misinterpret the verification as an indication of the caller identity (a misconception that is somewhat related to Misconception 11 regarding the infallibility of technical solutions), exposing themselves to vishing attacks.

Found in: Edwards et al. (2020).

4.2.4.13 Misc. 13 – Enabling anti-tracking in the smartphone options protects from phishing

Smartphones sometimes have built-in privacy-preserving technologies, e.g., impeding apps from tracking users when not in use. Participants consider the privacy focus of these technologies as meaning that they are also security enhancements capable of protecting them from phishing. This is obviously untrue, given that stopping tracking apps would not stop malicious emails from arriving, or stop a user from entering their credentials on a phishing website. As such, by considering privacy-focused technologies as having a security focus too, users rely on a false sense of protection that exposes them to risk.

Found in: Frik et al. (2022).

4.2.4.14 Misc. 14 – Blocking cookies protects from phishing

Similarly to Misconception 13, the participants interpret blocking cookies as a security-related action, instead of being primarily privacy-related. As such, as it is the case with the anti-tracking functions in smartphones, blocking the leak of data by blocking cookies does not protect users from releasing their sensitive information on a phishing website.

Found in: Frik et al. (2022).

4.2.4.15 Misc. 15 – Reporting phishing is not a user's duty

Misconception 15 was identified within the business context, but it also applies to the private context. The participants believe that reporting phishing is not important, and that it is not up-to-them to do so. This is obviously not true given that, as mentioned in Misconception 9, reporting something as SPAM is not the same as reporting it as phishing. As such, by not reporting those phishing attacks that users detect, they effectively slow down the response time to them, leaving other users exposed. Furthermore, it is important to highlight the proper channels to do so, because, as mentioned in Misconception 8, flagging something as SPAM is not the same as reporting it as phishing.

Found in: Marin et al. (2023).

4.2.5 QRNope Input: *Misconceptions on Attack Techniques* and *Misconceptions on Mitigation Strategies*

The misconceptions in Section 4.2.4 were not classified as being on attack techniques or on mitigation strategies. As such, the first step to obtain the input needed for the development of QRNope was to classify them accordingly. Namely, misconceptions were classified as being on attack techniques when they referred to an incorrect understanding of how phishing attacks are created, delivered, or presented to users. Conversely, misconceptions were classified as being on mitigation strategies when they referred to incorrect assumptions regarding how phishing attacks can be prevented, detected, or reported. This classification is shown in Table 4.5, alongside the rationale explaining why each belief is considered a misconception.

Table 4.5: Misconceptions classification between attack techniques and mitigation strategies, and rationale explaining why each belief is considered a misconception.

Misconception	Rationale
Misconceptions on attack techniques	
Misc. 2	Attackers can clone layouts and use them in attacks
Misc. 3	Not all attacks are delivered by link, not all links are malicious
Misc. 4	Attackers can use a link-text different from the actual URL
Misc. 5	Attackers can obtain TLS certificates for phishing websites
Misc. 6	Attackers can place legitimate names as subdomains or in the path
Misc. 7	Attackers can spoof senders
Misc. 8	Attackers can deliver malicious attachments from spoofed senders
Misconceptions on mitigation strategies	
Misc. 1	Passwords are only effective when not revealed
Misc. 9	Flagging something as SPAM is different than reporting it as phishing
Misc. 10	Technical solutions are not 100% effective
Misc. 11	False positives or false negatives are always possible
Misc. 12	Verified caller does not identify the actual caller
Misc. 13	Stopping apps from tracking does not stop phishing attacks
Misc. 14	Blocking cookies does not affect phishing delivery
Misc. 15	Reporting is also the duty of users

Once the misconceptions were classified as being either on attack techniques or mitigation strategies, the next step was to determine which ones were relevant for QRishing.

For the misconceptions on attack techniques, misconceptions 3, 4, and 8 were removed. Misconception 3 (“Clicking on a dangerous link always infects the device”) was removed because QR codes do not contain links that can be clicked, but encode content accessed through a QR code

scanner. Misconception 4 ("There is no difference between the text of a link and its URL") was removed because QR codes do not contain link text, but directly encode the content accessed by the QR code scanner (e.g., a URL). As such, these misconceptions are not applicable to QRishing. Misconception 8 ("Opening attachments from seemingly legitimate emails without further inquiry is not dangerous") was also removed, as QR codes do not contain attachments.

Regarding misconceptions on mitigation strategies, only misconception 12 ("If a Voice Over IP call shows a registered Caller ID, the identity of the caller is also verified.") was removed. This misconception concerns caller identification mechanisms and therefore does not apply to QR codes.

The misconceptions not relevant for the QRishing domain are summarized in Table 4.6.

Table 4.6: Misconceptions not relevant for the QRishing content, and the rationale why it is so.

Misconception	**Rationale**
Misconceptions on attack techniques	
Misc. 3	QR codes do not contain links
Misc. 4	QR codes do not contain links
Misc. 8	QR codes do not contain attachments
Misconceptions on mitigation strategies	
Misc. 12	No Caller ID for QR codes

The misconceptions remaining after this process are shown in Table 4.7.

Furthermore, the study in Chapter 2 highlighted two additional misconceptions specific to the QRishing context. Namely, the results in Section 2.3 highlighted that participants believed that A) "If a poster looks harmless it is safe to visit the QR code", and B) "Visiting a website without checking the URL in the QR code is safe".

Misconception A shows that users are not aware of the *Cloned* attack technique (described in Section 4.1.3), and that any pretext, be it a poster, an email, or a website, can be cloned. Misconception B shows that users might consider visiting the website encoded in a QR code safe without first checking the URL shown by the QR code scanner. However, several attack techniques can be identified directly from the QR code scanner without visiting the website.

As such, these two misconceptions were added to the misconceptions on attack techniques identified from the literature. The final inputs for the *Misconceptions on attack techniques* and *Misconceptions on mitigation strategies* stages are shown in Table 4.8.

Table 4.7: Misconceptions relevant for the QRishing content, and the rationale why it is so.

Misconception	Rationale
Misconceptions on attack techniques	
Misc. 2	QR codes can lead to websites
Misc. 5	QR codes can encode URLs
Misc. 6	The URL encoded in a QR code can use attack techniques
Misc. 7	QR codes can be sent from spoofed addresses
Misconceptions on mitigation strategies	
Misc. 1	Strong passwords do not protect against QRishing
Misc. 9	Flagging a message with a QR code as SPAM not the same as report it as phishing
Misc. 10	Technical solutions cannot find 100% of malicious QR codes
Misc. 11	Technical solutions might return false positives or false negatives
Misc. 13	Stopping apps from tracking does not stop QRishing attacks
Misc. 14	Blocking cookies does not affect QRishing delivery
Misc. 15	Reporting QRishing is also the duty of users

Table 4.8: Misconceptions relevant for the QRishing content, and the rationale why it is so.

Misconception	Rationale
Misconceptions on attack techniques	
Misc. A	QR codes can be embedded in cloned posters
Misc. B	The URL in a QR code should always be checked before visiting it
Misc. 2	QR codes can lead to websites
Misc. 5	QR codes can encode URLs
Misc. 6	The URL encoded in a QR code can use attack techniques
Misc. 7	QR codes can be sent from spoofed addresses
Misconceptions on mitigation strategies	
Misc. 1	Strong passwords do not protect against QRishing
Misc. 9	Flagging a message with a QR code as SPAM not the same as report it as phishing
Misc. 10	Technical solutions cannot find 100% of malicious QR codes
Misc. 11	Technical solutions might return false positives or false negatives
Misc. 13	Stopping apps from tracking does not stop QRishing attacks
Misc. 14	Blocking cookies does not affect QRishing delivery
Misc. 15	Reporting QRishing is also the duty of users

4.3 Conclusion

This chapter presented the processes followed to identify and structure the information required for three stages of Framework 2.0 during the development of QRNope. Specifically, the *All attack*

techniques in the domain stage required identifying the attack techniques applicable within the domain and context of QRNope. To this end, a systematic literature review and a systematic review of the MITRE ATT&CK® database were conducted, resulting in a taxonomy of QRishing attack techniques.

Furthermore, the *Misconceptions on attack techniques* stage and the *Misconceptions on mitigation strategies* stage required identifying users' misconceptions regarding phishing attacks and their mitigation. To this end, a systematic literature review on phishing misconceptions was conducted, resulting in the identification and categorization of 15 misconceptions about phishing. These misconceptions were then analyzed to determine which ones were relevant for QRishing and could therefore provide the required input for the development of QRNope.

The resulting attack techniques and misconceptions are used in their corresponding stages during the development of QRNope described in Chapter 5. By separating the collection of this information from the description of the development process itself, Chapter 5 can focus on the application of Framework 2.0 and the development of the final awareness intervention.

5 Development of *QRNope*, a QRishing Awareness Intervention

In Chapter 2 it was shown how users are unaware of the QRishing threat, and how they rely on uninformative cues to determine the legitimacy of QR codes (e.g., the layout of the pretext), instead of checking the information shown in their QR code scanner. However, simply being aware of the benefit of checking the information displayed by a QR code scanner might not be enough without knowing how to use that information. Within the phishing by email context, Zheng and Becker (2022) showed that even when users are given security-relevant information, they often do not know how to use it to determine legitimacy. As such, awareness interventions are necessary to prepare users and explain how to use the information they are given. Banks (e.g., Pennian Bank (2025)), private organizations (e.g., Seipel (2026)), and state organizations (e.g., FTC (2023)) already provide online awareness interventions on QRishing. However, to the best of the present knowledge, none of them was evaluated, nor does any documentation of their development process exist. This means that there is no way to determine how their content was generated nor if they are effective at supporting users distinguishing between phishing QR codes and legitimate ones. As such, there is need for an awareness intervention that is both documented in its development, and evaluated for its effectiveness.

This chapter describes the development of such a QRishing awareness intervention, called *QRNope*. Namely, the chapter first discusses related work in Section 5.1, focusing on existing online security advice and awareness interventions relevant for the design of QRNope. The bulk of this chapter is then dedicated to the detailed presentation of the development of QRNope following Framework 2.0, covering each Step and stage involved in the development process (Section 5.2). Finally, the resulting awareness intervention and the implications of its development are discussed in Section 5.3.

Note: The evaluation of QRNope is covered separately as part of Chapter 6.

Contributions described in this chapter:

- Demonstrate the application of Framework 2.0 through the development of a QRishing awareness intervention.
- Provide QRNope, a QRishing awareness intervention designed to support users in identifying and mitigating QRishing attacks.

5.1 Related Work

As described in Chapter 3, Framework 2.0 guides the developers through the stages that should be considered during the development of an awareness intervention, while leaving the intervention-specific details to the developers. A natural starting point for informing such decisions is the related work addressing similar awareness interventions, which in the case of QRNope (as described in Section 5.2.1.6), is other QRishing awareness webpages. Yet, to the best of the present knowledge, no previous work has systematically collected and analyzed existing online QRishing awareness interventions. Such an analysis is outside the scope of this thesis; instead, there is previous work on online phishing awareness interventions (i.e., Mossano et al. (2020)) and security advice from online sources more broadly (i.e., Redmiles et al. (2020)). Although neither work focused specifically on QRishing, both identified aspects that provide useful guidance for the development of QRNope. In particular, Mossano et al. (2020) analyzed 94 anti-phishing webpages from four types of organizations and nine countries, while Redmiles et al. (2020) identified 374 unique security recommendations across 614 online sources.

One issue identified by Mossano et al. (2020) is the limited use of visual material. Among the analyzed webpages, 57.45% lacked visual examples entirely. Furthermore, Redmiles et al. (2020) highlighted that the effectiveness of security advice depends not only on the content itself, but also on how the information is communicated to users. Together, these findings suggest that QRNope should not rely exclusively on textual explanations, but should instead incorporate concrete examples illustrating both how attacks may appear in practice and how users can obtain the necessary information to detect them.

A second aspect concerns victim support. Mossano et al. (2020) found that 56.38% of the analyzed webpages provided no explanation regarding how users should report phishing attacks or what victims should do after falling for one. These findings suggest that awareness interventions should not only focus on attack prevention, but should also consider how users are supported after an attack has occurred. While QRNope was originally intended to include such information directly, it was noticed that the FTC (the relevant reporting authority for the US, i.e., QRNope's target

population) already provides dedicated resources for reporting fraud and assisting victims. As such, it was determined to only direct the readers of QRNope to the relevant portal and avoid adding information directly in the awareness intervention, thereby maintaining a single point of truth.

Both Mossano et al. (2020) and Redmiles et al. (2020) also highlight problems related to the practical usefulness of security recommendations. For example, Mossano et al. (2020) found that only 15.96% of the analyzed anti-phishing webpages highlighted the specific features users should focus on when evaluating an example, suggesting that recommendations are often presented without sufficient guidance regarding how they should be applied in practice. Similarly, Redmiles et al. (2020) found substantial disagreement among both users and experts regarding which recommendations should be prioritized. In particular, users identified 146 different recommendations as belonging among the three most important security behaviors, while experts identified 118 different recommendations as belonging among the five most important. These findings suggest that merely providing recommendations is insufficient; recommendations should also be concrete, actionable, and clearly connected to observable cues. Consequently, it was determined that the mitigation strategies presented by QRNope should focus not only on what readers should do, but also on the specific characteristics they should examine and on how those mitigation strategies should be applied in practice.

5.2 Development of *QRNope*

The section contains the development process of *QRNope*. The structure of the section follows the steps of Framework 2.0 from Chapter 3.

5.2.1 Step 1 – Preparation

This section covers *Step 1* of the development process (shown in Figure 5.1). Specifically, it clarifies the domain of QRNope (Section 5.2.1.1), its target population (Section 5.2.1.2), the context of use (Section 5.2.1.3), and the attack techniques in the domain (Section 5.2.1.4). Furthermore, it defines the awareness goal (Section 5.2.1.5), the format of the intervention (Section 5.2.1.6), the accessibility criteria followed (Section 5.2.1.7), and the security assumptions formulated throughout the step (Section 5.2.1.8).

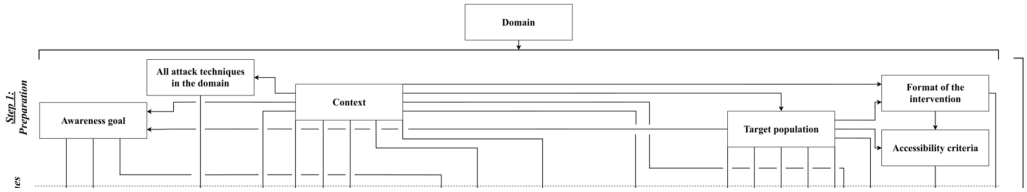


Figure 5.1: *Step 1* of the framework for developing awareness interventions. The *Security assumptions* stage is missing due to space.

5.2.1.1 Domain

The domain of QRNope is that of phishing attacks using QR codes that either encode the payload directly or take the user to the payload within one step. For example, a standalone QR code with a URL pointing at a malware app on the legitimate Apple "App Store" is part of the domain even if the URL itself is legitimate, because scanning the QR code directs to the payload (the malware app) within one step (after clicking the link). Conversely, those cases where a QR code is part of the pretext of an attack, but does not contribute to the delivery of the payload, are outside the domain. For example, a phishing email containing a legitimate QR code but delivering the payload through a link embedded in its body is not part of the domain.

5.2.1.2 Target Population

QRNope targets users that: i) live in the United States of America (*hereafter*, US), ii) have no IT background, i.e., *lay users* (as defined in Definitions and Terminology), and iii) have no previous phishing awareness. The latter two characteristics should not be interpreted as claiming that every member of the target population lacks technical expertise or has neither previously encountered nor heard of phishing. Rather, they were intentionally selected to guide the development of QRNope by considering users with the least prior knowledge. Specifically, considering readers with no IT background ensured that the intervention avoided unnecessary technical terminology and did not rely on specialist knowledge, while considering readers with no previous phishing awareness ensured that phishing-related concepts and recommendations were introduced without assuming any prior familiarity with the topic. Consequently, while readers possessing technical expertise or previous phishing awareness can still benefit from QRNope, the intervention was intentionally designed so that neither is required to understand its content. Furthermore, no special accessibility requirements are considered, e.g., braille screen translation. However, due to QRNope's format described in Section 5.2.1.6, accessibility guidelines as delineated in Nurthen et al. (2025) are still followed (as explained in Section 5.2.1.7).

5.2.1.3 Context

The intended use case of QRNope is that of a lay user searching online for recommendations against QR code phishing. The intended context of QRNope, and to which the recommendations refer to, is the scanning of QR codes encountered throughout everyday life in the US. For example, a QR code on a parking meter, or a sticker with only a QR code attached to a bus stop. Consequently, QRNope focuses on everyday QR code interactions rather than organizational or institutional environments. As such, governmental policies, mandatory training, dedicated IT assistance, and other business practices are outside the scope of the intervention.

Regarding the technology used to scan QR codes, QRNope only considers relatively common technologies with QR code scanning capabilities (e.g., smartphones and tablets). In other words, QRNope ignores niche devices with such capabilities, e.g., the Microsoft Hololens (see the lack of adoption of Hololens in Stewart (2024)). Specifically, given that the context of use is everyday life, the most likely technologies considered are widespread mobile devices.

However, not all mobile devices are the same. Different operating systems and devices provide different features and user interface elements, meaning that not all recommendations or attack techniques apply equally. Given that the target population of QRNope is from the US, Apple devices were selected because they constitute the largest market share in the country (according

to Elagina (2025)), thereby potentially covering the highest number of users in the target population. Consequently, QRNope explicitly focuses on iOS devices, specifically iPhones running at least iOS 26.2[1].

More precisely, QRNope targets lay users of iOS devices that use the default QR code scanner (hereafter, *iOS Camera*) to scan QR codes. This further limitation is due to the variability among third-party QR code scanner applications. For example, different apps employ different URL visualizations, exposing users to different attack techniques. A subdomain attack technique might be effective when the full URL is shown, but ineffective when a pruned visualization is employed (as shown in Volkamer et al. (2016)). Similarly, some QR code scanner applications provide additional security-related features, such as warnings or automated phishing URL detection, while others do not. A market investigation could potentially reveal more detailed information on which third-party apps are the most common ones, but such an endeavor is outside the scope of this thesis. Instead, relying on the iOS Camera allows to virtually cover every iOS user, given that, even if they normally use third-party apps, they would still have access to the iOS Camera.

5.2.1.4 All attack techniques in the domain

The input for the *All attack techniques in the domain* stage was generated in Section 4.1.4. Specifically, the taxonomy of QR code attack techniques created through the systematic literature review and the systematic review of the MITRE ATT&CK® database was restricted according to the domain and context of QRNope. The resulting attack techniques were then investigated to determine whether any of them were already addressed by the selected operating system or by the URL visualization employed by the iOS Camera. The complete process and its results are described in Section 4.1.4.

During the review of the MITRE ATT&CK® database, it was noticed that several iOS attack techniques no longer required consideration because they had already been addressed by the latest version of iOS available at the time of the investigation. This led to the formulation of the following security assumption: **A_1.** *If the readers of QRNope scan a QR code exploiting an attack technique already addressed by an updated of iOS upt to iOS 26.2, no harm comes to them because the attack technique is already addressed by the operating system.*

Furthermore, the investigation described in Section 4.1.4 showed that the main URL visualization employed by the iOS Camera rendered several URL attack techniques ineffective. This led to the formulation of the following security assumption: **A_2.** *If the readers of QRNope scan a QR code exploiting an attack technique rendered ineffective by the main URL visualization employed by*

[1] This version was selected because it was the latest one available at the time of the investigation, i.e., January 2026.

the iOS Camera shipped with iOS 26.2, no harm comes to them because the attack technique is already addressed by the URL visualization.

However, the iOS Camera also allows users to inspect the complete URL encoded in a QR code through an additional view. If readers used this view, the parts of the URL omitted by the main URL visualization would become visible, and the attack techniques covered by A_2 would no longer be considered addressed. Moreover, several previous works (e.g., Alsharnouby et al. (2015), Albakry et al. (2020), Dhamija et al. (2006), Zheng and Becker (2022)) showed that users are often unaware of how to correctly read a complete URL, while truncated URL visualizations led to a higher rate of phishing URL detection (as found in, e.g., Volkamer et al. (2016)). This led to the formulation of the following security assumption: **A_3.** *If the readers of QRNope encounter an attack technique rendered ineffective by the main URL visualization employed by the iOS Camera and base their decision solely on that visualization, no harm comes to them.*

As such, QRNope explicitly points readers to the main URL visualization presented by the iOS Camera, and it is assumed that they follow this indication. After accounting for the domain, context, and security assumptions A_1–A_3, the final input for the *All attack techniques in the domain* stage is the set of attack techniques shown in Table 5.1[2]. These attack techniques are used as input for the following stages of Framework 2.0.

5.2.1.5 Awareness goal

The awareness goal of QRNope is based on the awareness maturity curve from Hielscher et al. (2021), following the interpretation of it by Schöni et al. (2024) (shown in Figure 5.2). Namely, the awareness goal of QRNope is to reach the full 'Educational Element", i.e., after reading QRNope the readers obtain the necessary knowledge to detect the attacks covered, but the intervention does not contain exercises to practice this knowledge.

5.2.1.6 Format of the intervention

QRNope is intended as a web page, a choice based on four reasons: The first reason is the ease of correction and update should changes become necessary. Such changes are expected over time, for example due to changes in the iOS Camera URL visualization, the emergence of new attack techniques, or the identification of improved mitigation strategies. Furthermore, examples included in QRNope might eventually become unsuitable (e.g., if they become part of a phishing campaign). While modifying a printed version of QRNope (e.g., a flyer) would require redistributing the

[2] The table is the same as Table 4.4. It is repeated here for the reader's convenience.

Table 5.1: All attack techniques in the domain with descriptions or examples. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*.

Attack technique	Description or Example
Embedded	*MQRC is part of a pretext*
Standalone	*MQRC is not part of a pretext*
Self created	*MQRC is an original creation of attackers*
Altered	*Attackers modify an existing LQRC*
Cloned	
Added	*MQRC added to a pre-existing, QR code-less, legitimate pretext*
Replaced	*LQRC part of a pre-existing, legitimate pretext replaced with a MQRC*
Malicious QR code generator	*Malicious generator used to disseminate MQRCs for attackers*
Malicious QR code scanner	*Malicious app installed by users*
Malicious Phone Number	
Impersonation	`tel:12345678` (*Attackers' number*)
USSD Code Exploitation	`tel:*21*00113371337#` (*Redirect all calls to the number after* `*21*`)
Premium Rate Service Exploitation	`tel:090012345678`
Injection attack	`https://demal.is/login?user=admin' OR '1'='1`
Specific App MQRC	
Fake payment	*MQRC used to deliver a payment to the attackers*
QRL-Jacking	*MQRC used to hijack the victims' session token through QR code login*
Malicious URL	
Mismatch	
Entire URL in Pretext	*Pretext URL:* `https://google.com` – *MQRC URL:* `https://demal.is`
Part of the URL in Pretext	*Pretext URL:* `google.com` – *MQRC URL:* `https://demal.is`
External Content Manipulation	`https://docs.google.com/document/d/` [...]
URL Shortener	`https://bit.ly/4qWnYMj`
URL Redirect	`https://google.com/url?sa=t&source=web&rct=j&url=https://demal.is`
Unrecognizable domain	
IP Address	`https://142.250.74.206`
Random Characters	`https://aldslkskmskoewlc.at`
Use of Fitting Keywords	`https://docs-share-online.com/presentation`
Domain Extension	`https://secure-google.com`
Subdomain	`https://cnn.com.is`
Different Top-Level-Domain	`https://google.io`
Mangle	`https://gogle.com`
Custom-Scheme and Deep-Links	`msteamms://teams.microsoft.com/l/chat/` [...]

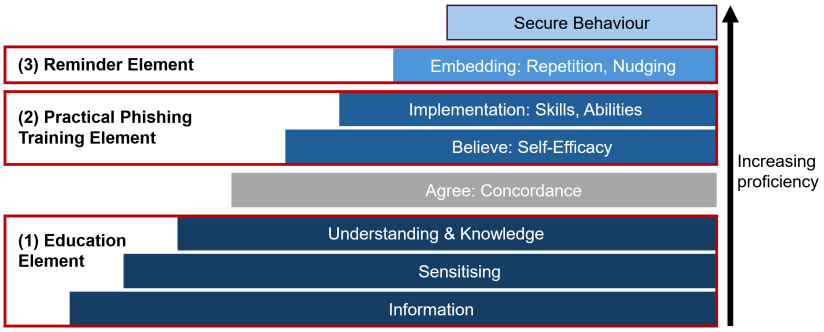


Figure 5.2: Awareness maturity curve from Schöni et al. (2024). Highlighting in the original. QRNope's goal is to cover the full "Education Element".

intervention, a webpage can be updated without replacing previously distributed copies. The second reason is the ease of expansion. If, e.g., new attacks become relevant within the domain, using a web page allow for a relatively quick extension of the intervention. The third reason is that a webpage can be easily reached by the target population. Furthermore, should QRNope

eventually be adopted by organizations already providing cybersecurity recommendations (e.g., the FTC or the FBI), it could be integrated into communication channels already familiar to the target population. The fourth and last reason is that using a webpage allows for the implementation of the accessibility criteria identified in 5.2.1.7, making QRNope more accessible to visually impaired users (through screen readers) than a text without a braille translation.

5.2.1.7 Accessibility criteria

As mentioned already in Section 5.2.1.2, no special accessibility need is assumed besides those for online content described in Nurthen et al. (2025). To achieve this, the ARIA HTML notation (documented in Diggs et al. (2023)) is used throughout the QRNope implementation. These attributes allow for the description of the images contained in the measure, and for the ease of presentation of the links contained. Furthermore, even though the intervention is mean to use red and green to indicate, respectively, danger and safety, color blind friendliness is achieved through the use of icons, and font shapes and dimension that allow for easy distinction of both danger and safety, and of other user interface elements, e.g., titles, labels, and parts that are meant to draw the readers' attention.

5.2.1.8 Security assumptions (End of *Step 1*)

The security assumptions formulated throughout Step 1 are summarized in this section. Each assumption is followed by a short description of the rationale behind it, as recommended by Van Landuyt and Joosen (2022) for documenting the origin of assumptions. This decision is intended to facilitate future reassessment of the assumptions and determine whether any of them are no longer valid, as suggested by both Van Landuyt and Joosen (2022) and Scandariato et al. (2015). For example, if a system initially assumed to be secure is later found to be insecure, the documented rationale behind each assumption would enable the identification of assumptions that relied on that system and therefore need to be revisited.

A_1. *If the readers of QRNope encounter an attack technique already addressed by one of the iOS updates up to and including iOS 26.2, no harm comes to them.*

A_1 originated from the review of the MITRE ATT&CK® database, during which several attack techniques were found to be already addressed by iOS 26.2. Consequently, these attack techniques were excluded from further consideration during the development of QRNope.

A_2. *If the readers of QRNope encounter an attack technique rendered ineffective by the main URL visualization employed by the iOS Camera shipped with iOS 26.2 and base their decision solely on that visualization, no harm comes to them.*

A_2 originated from the investigation described in Section 4.1.4, which showed that the main URL visualization employed by the iOS Camera rendered several URL-based attack techniques ineffective by omitting the part of the URL carrying the attack. Consequently, these attack techniques were excluded from further consideration during the development of QRNope.

A_3. *If the readers of QRNope encounter an attack technique rendered ineffective by the main URL visualization employed by the iOS Camera and base their decision solely on that visualization, no harm comes to them.*

A_3 originated from the possibility offered by the iOS Camera to inspect the complete URL encoded in a QR code (called *expanded URL view* in this thesis). Several previous works (e.g., Alsharnouby et al. (2015), Albakry et al. (2020), Dhamija et al. (2006), Zheng and Becker (2022)) showed that users are generally unable to correctly interpret complete URLs, while truncated URL visualizations lead to a higher phishing URL detection rate (as shown in, e.g., Volkamer et al. (2016), Mossano et al. (2023)). As such, QRNope explicitly instructs its readers to base their decision on the main URL visualization employed by the iOS Camera, and it is assumed that they follow this recommendation.

Table 5.2 summarizes the attack techniques assumed to be addressed at the end of Step 1 and the corresponding security assumption.

Table 5.2: Attack techniques assumed to be addressed at the end of Step 1.

Attack technique	External mechanism	Assumption
Malicious URL		
Hexadecimal encoding	URL visualization	A_2, A_3
URL Path/Query/Fragment		
Domain in Path	URL visualization	A_2, A_3
Entire URL in Path	URL visualization	A_2, A_3
Luring-Label with Domain	URL visualization	A_2, A_3
Luring-Label with URL	URL visualization	A_2, A_3
Look-Alike Domain	URL visualization	A_2, A_3
Omitting Slash	URL visualization	A_2, A_3
UserInfo Field	URL visualization	A_2, A_3
Homographic Spoofing	URL visualization	A_2, A_3
Exceedingly Long URL	URL visualization	A_2, A_3

The remaining attack techniques at the end of *Step 1* are shown in Table 5.3.

Table 5.3: Attack techniques remaining at the end of *Step 1*. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*.

Attack technique	Description or Example
Embedded	*MQRC is part of a pretext*
Standalone	*MQRC is not part of a pretext*
Self created	*MQRC is an original creation of attackers*
Altered	*Attackers modify an existing LQRC*
Cloned	
Added	*MQRC added to a pre-existing, QR code-less, legitimate pretext*
Replaced	*LQRC part of a pre-existing, legitimate pretext replaced with a MQRC*
Malicious QR code generator	*Malicious generator used to disseminate MQRCs for attackers*
Malicious QR code scanner	*Malicious app installed by users*
Malicious Phone Number	
Impersonation	`tel:12345678` (*Attackers' number*)
USSD Code Exploitation	`tel:*21*00113371337#` (*Redirect all calls to the number after* `*21*`)
Premium Rate Service Exploitation	`tel:090012345678`
Injection attack	`https://demal.is/login?user=admin' OR '1'='1`
Specific App MQRC	
Fake payment	*MQRC used to deliver a payment to the attackers*
QRL-Jacking	*MQRC used to hijack the victims' session token through QR code login*
Malicious URL	
Mismatch	
Entire URL in Pretext	*Pretext URL:* `https://google.com` – *MQRC URL:* `https://demal.is`
Part of the URL in Pretext	*Pretext URL:* `google.com` – *MQRC URL:* `https://demal.is`
External Content Manipulation	`https://docs.google.com/document/d/` [...]
URL Shortener	`https://bit.ly/4qWnYMj`
URL Redirect	`https://google.com/url?sa=t&source=web&rct=j&url=https://demal.is`
Unrecognizable domain	
IP Address	`https://142.250.74.206`
Random Characters	`https://aldslkskmskoewlc.at`
Use of Fitting Keywords	`https://docs-share-online.com/presentation`
Domain Extension	`https://secure-google.com`
Subdomain	`https://cnn.com.is`
Different Top-Level-Domain	`https://google.io`
Mangle	`https://gogle.com`
Custom-Scheme and Deep-Links	`msteamms://teams.microsoft.com/l/chat/` [...]

5.2.2 Step 2 – Attack techniques to consider

This section covers *Step 2* of the development process (shown in Figure 5.3). Specifically, it covers the maximum time available (Section 5.2.2.1), the attacks to consider (Section 5.2.2.2), and the security assumptions (Section 5.2.2.3).



Figure 5.3: *Step 2* of the framework for developing awareness interventions. The *Security assumptions* stage is missing due to space.

5.2.2.1 Maximum time available

As described in Section 5.2.1.3, the chosen context is the everyday life of the target population. Expecting users to spend long periods of their own free time on awareness is unrealistic at best, and disrespectful at worst, because it implies that personal time has no value, which is not the case. For this reason, QRNope should not require more than roughly 5 minutes to be read and understood in all of its parts.

Framework 2.0 requires the developers to determine the maximum time available for the awareness intervention, but does not prescribe how this value should be estimated. Therefore, the following estimation procedure is specific to QRNope.

The 5-minutes reading-time is based on the "long text" intervention from Reinheimer et al. (2020), which was aimed at supporting the detection of phishing through emails. Reinheimer et al. (2020) contains several distinct interventions, all showed to have various levels of effectiveness: a video, an interactive email, and two texts. The interactive email is not relevant to QRNope's domain, which leaves video and the two texts as potential maximum time bases. Because the chosen format of QRNope is that of a webpage (as stated in Section 5.2.1.6), the focus is on the two texts, as these are closer to the webpage format.

The two textual interventions differ in length, with the "Long text" intervention being more complete than the "Short text" intervention, and achieving a higher sensitivity score (i.e., a value indicating the correct distinction between phishing emails and legitimate emails; 1.61 for the Long text vs. 1.56 for the Short text). Given the higher effectiveness, the Long text intervention is used to determine the reading-time.

The Long text intervention is analyzed through the *koRpus* R package described in Michalke et al. (2021) to calculate the number of words contained, i.e., 1097 words. This number is then divided by the average number of words per minute (*hereafter*, WPM) that a reader can read to obtain the average reading-time needed to go over the whole Long text intervention. The WPM depends on both the language of the text and the native language of the reader, meaning that the WPM differs across languages and readers. Regarding the readers, for the calculation it is assumed that all readers are native speaker of the language of an intervention. Therefore, because the Long text intervention is in German, the WPM to consider is that of a native German speaker, identified by Trauzettel-Klosinski et al. (2012) to be 249 WPM. After the total number of words of the Long text intervention is divided by the average WPM, the average reading-time needed is estimated at 4.4 minutes.

Yet, QRNope is aimed to a target population living in the US, meaning that the language used is English, not German. As such, and following the assumption that all readers are native speakers,

the estimated reading-time needs to be based on the WPM of a native English speaker, determined by Brysbaert (2019) to be 238 WPM, i.e., slightly slower than the WPM of a German speaker. Hence, assuming that QRNope will have the same length as the Long text intervention (i.e., 1097 words), a maximum reading-time of 5 minutes is estimated, i.e., roughly 20 seconds longer than the Long Text intervention.

Admittedly, the estimated maximum reading-time of 5 minutes is an approximation, as the density and number of words per concept (i.e., how many words are required to explain a concept; *hereafter*, WPC) are not 1-to-1 transposable between different languages. This means that the same concept in one language might require more (or less) words to be conveyed in another language. As such, the assumption that QRNope will have the same length as the Long text intervention is not entirely warranted, given that the two interventions use different languages. Nonetheless, a maximum reading-time of 5 minutes is considered an acceptable approximation, because Kunz and Lapshinova-Koltunski (2018) show that, albeit the WPC are significantly different between the English language and the German language when considering fictional texts, this WPC difference is practically not present in scientific texts, such as papers. Admittedly, neither the Long text intervention nor QRNope are papers, meaning that the registry and complexity of the language used are likely different than they would be in a scientific text, i.e., they both use so called "simple language" instead of the "technical language" of a scientific publication. As such, simple English might require a higher WPC than simple German, leading the former to have a longer reading-time than the latter. Given all the above, the estimated maximum reading-time of 5 minutes should be interpreted as a design target rather than the measured reading-time of QRNope. A more precise reading-time estimate is obtained in Step 5 (covered in Section 5.2.5), once the content of QRNope is finalized. Furthermore, the reading-time is estimated again in Step 6 (Section 5.2.7.2) after the feedback cycles with experts and the target population.

5.2.2.2 Attack techniques to consider

At this stage, the attack techniques identified in Step 1 are further reduced according to the maximum time available determined in Section 5.2.2.1.

Among the remaining attack techniques, only the *Injection attack* was considered unsuitable for QRNope. To understand how an injection attack works, a considerable amount of technical knowledge would first have to be introduced. For example, readers would first need to understand the role of SQL databases, how websites interact with them, and how malicious code can manipulate database queries to extract or modify information. Furthermore, unlike the remaining attack techniques considered by QRNope, injection attacks do not correspond to a single, recognizable

pattern that users can identify during everyday QR code interactions. Rather, they encompass a broad family of attacks exploiting different vulnerabilities and requiring different mitigation strategies. Consequently, explaining the *Injection attack* in a sufficiently accurate yet understandable way to a target population without an IT background is likely to require a disproportionate part of the maximum reading-time available, possibly pushing the awareness intervention past its maximum reading-time.

This led to the formulation of the following security assumption: **A_4.** *If the readers of QRNope encounter an injection attack, no harm comes to them because the targeted web application correctly validates and sanitizes user input.*

5.2.2.3 Security assumptions (End of *Step 2*)

The security assumption formulated in Step 2 is described in this section. The assumption is followed by a short description of the rationale behind it, as recommended by Van Landuyt and Joosen (2022) for documenting the origin of assumptions. This decision is intended to facilitate future reassessment of the assumption and determine whether it is no longer valid, as suggested by both Van Landuyt and Joosen (2022) and Scandariato et al. (2015). For example, if a system initially assumed to be secure is later found to be insecure, the documented rationale behind the assumption would enable the identification of the assumption that relied on that system and therefore needs to be revisited.

A_4. *If the readers of QRNope encounter an injection attack, no harm comes to them because the targeted web application correctly validates and sanitizes user input.*

A_4 is a so-called *trust assumption*, as described in Haley et al. (2004, 2006). Trust assumptions rely on other mechanisms or actors to implement the necessary security practices required to mitigate specific security threats. In the case of A_4, this trust assumption originated from the exclusion of the *Injection attack* due to the maximum reading-time available. Injection attacks are prevented through the correct validation and sanitization of user input by web applications. Consequently, the *Injection attack* was excluded from further consideration during the development of QRNope.

Table 5.4 shows the updated list of attack techniques assumed to be addressed at the end of *Step 2*.

The remaining attack techniques at the end of *Step 2* are shown in Table 5.5.

Table 5.4: Attack techniques assumed to be addressed at the end of Step 2.

Attack technique	**External mechanism**	**Assumption**
Injection attack	Input validation and sanitization	A_4
Malicious URL		
Hexadecimal encoding	URL visualization	A_2, A_3
URL Path/Query/Fragment		
Domain in Path	URL visualization	A_2, A_3
Entire URL in Path	URL visualization	A_2, A_3
Luring-Label with Domain	URL visualization	A_2, A_3
Luring-Label with URL	URL visualization	A_2, A_3
Look-Alike Domain	URL visualization	A_2, A_3
Omitting Slash	URL visualization	A_2, A_3
UserInfo Field	URL visualization	A_2, A_3
Homographic Spoofing	URL visualization	A_2, A_3
Exceedingly Long URL	URL visualization	A_2, A_3

Table 5.5: Attack techniques remaining at the end of *Step 2*. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*.

Attack technique	**Description or Example**
Embedded	*MQRC is part of a pretext*
Standalone	*MQRC is not part of a pretext*
Self created	*MQRC is an original creation of attackers*
Altered	*Attackers modify an existing LQRC*
Cloned	
Added	*MQRC added to a pre-existing, QR code-less, legitimate pretext*
Replaced	*LQRC part of a pre-existing, legitimate pretext replaced with a MQRC*
Malicious QR code generator	*Malicious generator used to disseminate MQRCs for attackers*
Malicious QR code scanner	*Malicious app installed by users*
Malicious Phone Number	
Impersonation	`tel:12345678` (*Attackers' number*)
USSD Code Exploitation	`tel:*21*00113371337#` (*Redirect all calls to the number after* `*21*`)
Premium Rate Service Exploitation	`tel:090012345678`
Specific App MQRC	
Fake payment	*MQRC used to deliver a payment to the attackers*
QRL-Jacking	*MQRC used to hijack the victims' session token through QR code login*
Malicious URL	
Mismatch	
Entire URL in Pretext	*Pretext URL:* `https://google.com` – *MQRC URL:* `https://demal.is`
Part of the URL in Pretext	*Pretext URL:* `google.com` – *MQRC URL:* `https://demal.is`
External Content Manipulation	`https://docs.google.com/document/d/` [...]
URL Shortener	`https://bit.ly/4qWnYMj`
URL Redirect	`https://google.com/url?sa=t&source=web&rct=j&url=https://demal.is`
Unrecognizable domain	
IP Address	`https://142.250.74.206`
Random Characters	`https://aldslkskmskoewlc.at`
Use of Fitting Keywords	`https://docs-share-online.com/presentation`
Domain Extension	`https://secure-google.com`
Subdomain	`https://cnn.com.is`
Different Top-Level-Domain	`https://google.io`
Mangle	`https://gogle.com`
Custom-Scheme and Deep-Links	`msteamms://teams.microsoft.com/l/chat/` [...]

5.2.3 Step 3 – Attack techniques and mitigation strategies to consider

This section covers *Step 3* of the development process (shown in Figure 5.4). Specifically, it covers the identification of mitigation strategies (Section 5.2.3.1), the determination of the attack techniques with mitigation strategies to consider (Section 5.2.3.2), and the identification of the constraints of the target population (Section 5.2.3.3). It then determines the mitigation strategies actionable by the target population (Section 5.2.3.4), the attack techniques with actionable mitigation strategies to consider (Section 5.2.3.5), and the corresponding security assumptions (Section 5.2.3.6).

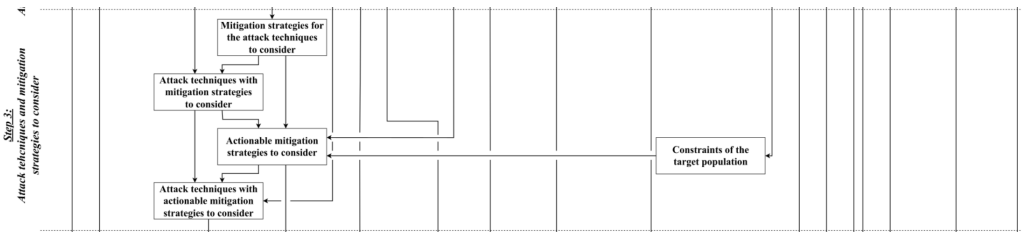


Figure 5.4: *Step 3* of the framework for developing awareness interventions. The *Security assumptions* stage is missing due to space.

5.2.3.1 Mitigation strategies for all attack techniques to consider

The mitigation strategies were identified by reviewing the literature associated with each of the attack techniques remaining at the end of *Step 2*. Specifically, the references associated with each attack technique in Chapter 4 were investigated to identify mitigation strategies applicable to the corresponding attack.

The mitigation strategies identified through this process were classified into one of three types.

Type 1 ($n = 24$) denotes mitigation strategies directly identified in the literature. When the reviewed literature explicitly proposed a mitigation strategy, that recommendation was adopted. Sometimes a small level of terminological normalization was required, e.g., the use of the "Standalone QR code" instead of "QR code sticker".

Type 2 ($n = 3$) denotes mitigation strategies synthesized from the mitigation strategies of distinct attack techniques. The resulting mitigation strategy did not introduce new security advice, but unified semantically equivalent or complementary mitigation strategies into a single formulation. For example, several publications described attackers adding phishing QR codes to a pretext,

replacing existing QR codes with phishing ones, or embedding phishing QR codes in copied pretexts. Taken together, these findings supported the broader principle that the pretext surrounding a QR code cannot establish legitimacy. They were therefore synthesized into the mitigation strategy *Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext.*

Type 3 ($n = 6$) denotes mitigation strategies derived from the corresponding attack technique. When neither direct identification nor a synthesis yielded an adequate mitigation strategy for an attack technique, a mitigation strategy was systematically derived from the mechanics of the corresponding attack. Specifically, the user behavior required for the attack to succeed was identified, and the mitigation strategy was formulated as the inverse of that behavior. For example, the *URL Redirect* attack technique hides the destination URL within the path of another URL. The attack mechanism was therefore inverted by encouraging users to inspect the URL path. This resulted in the mitigation strategy *Check the path of the URL for the destination.*

Table 5.6 shows the resulting mitigation strategies, together with their classification and the literature supporting each mitigation strategy.

5.2.3.2 Attack techniques with mitigation strategies to consider

All attack techniques were associated with a mitigation strategy. Consequently, the attack techniques to consider in the *Actionable mitigation strategies to consider* stage are all those shown in Table 5.6.

5.2.3.3 Constraints of the general population

Framework 2.0 requires the developers to determine the constraints imposed by the target population on the awareness intervention. For QRNope, these constraints were derived from the target population identified in Section 5.2.1.2. Specifically, because QRNope targets lay users, technical terminology is to be avoided. Furthermore, US health-literacy recommendations (as shown in, e.g., Hutchinson et al. (2016)) and recommendations for public-facing information (as specified in, e.g., US Environmental Protection Agency (2020)) suggest that material intended for the general public should be understandable at approximately a 7th-grade reading level. Consequently, QRNope was designed to be understandable at approximately a 7th-grade reading level.

Besides these two constraints, no further constraints are considered. Specifically, the target population is assumed to possess the abilities to use QR codes and smartphones expected of an average US citizen (e.g., no need for visually impaired support technology).

Table 5.6: Mitigation strategies found in the attack techniques sources. Type denotes whether the mitigation strategy was (1) directly identified in the literature, (2) synthesized from similar mitigation strategies, or (3) derived from the corresponding attack technique.

Attack technique	Mitigation strategy	Type	Citation(s)
Embedded	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext	2	Bekavac et al. (2024); Krombholz et al. (2015); Vidas et al. (2013)
Standalone	If possible, avoid scanning standalone QR codes	1	Gyimah et al. (2024); Krombholz et al. (2014); Sravan et al. (2024)
Self created	Anyone can create and disseminate QR codes	1	Kieseberg et al. (2012); Krombholz et al. (2014); Petrova et al. (2016)
Altered	Always check QR codes for tampering	1	Bekavac et al. (2024); Krombholz et al. (2015); Yong et al. (2019)
Cloned			
Added	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext	2	Bekavac et al. (2024); Bhardwaj et al. (2020); Kieseberg et al. (2012)
Replaced	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext	2	Bekavac et al. (2024); Krombholz et al. (2014); Petrova et al. (2016)
Malicious QR code generator	Inform yourself of a developer trustworthiness	3	Dudheria (2017); Froehlich et al. (2021); Rafsanjani et al. (2023)
	Only use the official app store	3	Pawar et al. (2022); Rafsanjani et al. (2023); Sabri et al. (2023)
Malicious QR code scanner	Inform yourself of a developer trustworthiness	1	Focardi et al. (2018); Focardi et al. (2019); Rafsanjani et al. (2023)
	Only use the official app store	1	Focardi et al. (2018); Gyimah et al. (2024); Rafsanjani et al. (2023)
	Deny unnecessary permissions	1	Focardi et al. (2018); Focardi et al. (2019); Rafsanjani et al. (2023); Rottermanner et al. (2015)
Malicious Phone Number			
Impersonation	Always look for contacts yourself	1	Yao and Shin (2013)
	Never use the contacts provided in the QR code or pretext	1	Yao and Shin (2013)
USSD Code Exploitation	Always look for contacts yourself	1	Sravan et al. (2024); Yao and Shin (2013)
	Never use the contacts provided in the QR code or pretext	1	Sravan et al. (2024); Yao and Shin (2013)
Premium Rate Service Exploitation	Always look for contacts yourself	1	Yao and Shin (2013)
	Never use the contacts provided in the QR code or pretext	1	Yao and Shin (2013)
Specific App MQRC			
Fake payment	Always check the receiver of a payment	1	Jubur (2024); Sravan et al. (2024); Varshney et al. (2024a)
QRL-Jacking	If possible, avoid authenticating to services through QR code login.	3	Blancaflor et al. (2024);Subash et al. (2023)
Malicious URL			
Mismatch			
Entire URL in Pretext	Never trust URLs in pretexts, always check what the app shows	1	Dudheria (2017); Krombholz et al. (2014); Yao and Shin (2013); Yong et al. (2019)
Part of the URL in Pretext	Never trust URLs in pretexts, always check what the app shows	1	Dudheria (2017); Krombholz et al. (2014); Yao and Shin (2013); Yong et al. (2019)
External Content Manipulation	If possible, avoid opening user-generated online documents	3	Veit et al. (2026a)
URL Shortener	Use an expansion service	1	Dudheria (2017); Krombholz et al. (2014); Rafsanjani et al. (2023); Yong et al. (2019)
URL Redirect	Check the path of the URL for the destination	3	Dudheria (2017); Krombholz et al. (2014); Rafsanjani et al. (2023); Yong et al. (2019)
Unrecognizable domain			
IP Address	Stop when IP address	1	Wang and Duncan (2019)
Random Characters	Stop when registered domain not plausible	1	Wang and Duncan (2019)
Use of Fitting Keywords	Distrust URLs with buzzwords	1	Bhardwaj et al. (2020); Wang and Duncan (2019)
Domain Extension	Distrust URLs with buzzwords	1	Wang and Duncan (2019)
Subdomain	Always focus on the registered domain	1	Krombholz et al. (2014); Wang and Duncan (2019)
Different Top-Level-Domain	Carefully read the URL and make sure that its last part corresponds to the legitimate one	1	Wang and Duncan (2019); Zulkefli et al. (2017)
	Check for further information in a search engine	1	Wang and Duncan (2019)
Mangle	Carefully read the registered domain	1	Wang and Duncan (2019); Zulkefli et al. (2017)
Custom-Scheme and Deep-Links	Distrust QR code content with unknown schemes	3	Veit et al. (2026a)

5.2.3.4 Actionable mitigation strategies to consider

Table 5.7 shows whether a mitigation strategy is actionable based on the constraints of the target population.

First of all, two attack techniques (highlighted in orange in Table 5.7) have mitigation strategies in conflict with assumption A_3, i.e., to base the decision solely on the main URL visualization employed by the iOS Camera. Specifically, the mitigation strategy *Check the path of the URL for the destination* requires users to inspect the URL path, which is only visible through the expanded

Table 5.7: Actionable mitigation strategies. Highlighted in orange are those mitigation strategies conflicting with previous security assumptions, in magenta are those too technically complex for the target population, in cyan are the so-called "unreasonable mitigation strategies," and in blue the so called "interruption mitigation strategies". Interruption mitigation strategies marked with (∗) are treated differently due to the outcome being the same.

Attack technique	Mitigation strategy	Actionable by target population
Embedded	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext	Yes
Standalone	If possible, avoid scanning standalone QR codes	No, unreasonable
Self created	Anyone can create and disseminate QR codes	Yes
Altered	Always check QR codes for tampering	Yes
Cloned		
Added	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext	Yes
Replaced	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext	Yes
Malicious QR code generator	Inform yourself of a developer trustworthiness	No, interruption
	Only use the official app store	Yes
Malicious QR code scanner	Inform yourself of a developer trustworthiness	No, interruption
	Only use the official app store	Yes
	Deny unnecessary permissions	No, too complex
Malicious Phone Number		
Impersonation	Always look for contacts yourself (∗)	No, interruption (∗)
	Never use the contacts provided in the QR code or pretext	Yes
USSD Code Exploitation	Always look for contacts yourself (∗)	No, interruption (∗)
	Never use the contacts provided in the QR code or pretext	Yes
Premium Rate Service Exploitation	Always look for contacts yourself (∗)	No, interruption (∗)
	Never use the contacts provided in the QR code or pretext	Yes
Specific App MQRC		
Fake payment	Always check the receiver of a payment	Yes
QRL-Jacking	If possible, avoid authenticating to services through QR code login	No, unreasonable
Malicious URL		
Mismatch		
Entire URL in Pretext	Never trust URLs in pretexts, always check what the app shows	Yes
Part of the URL in Pretext	Never trust URLs in pretexts, always check what the app shows	Yes
External Content Manipulation	If possible, avoid opening user-generated online documents	No, unreasonable
URL Shortener	Use an expansion service	No, interruption
URL Redirect	Check the path of the URL for the destination	No, conflict A_3
Unrecognizable domain		
IP Address	Stop when IP address	Yes
Random Characters	Stop when registered domain not plausible	Yes
Use of Fitting Keywords	Distrust URLs with buzzwords	Yes
Domain Extension	Distrust URLs with buzzwords	Yes
Subdomain	Always focus on the registered domain	Yes
Different Top-Level-Domain	Carefully read the URL and make sure that its last part corresponds to the legitimate one	Yes
	Check for further information in a search engine	No, interruption
Mangle	Carefully read the registered domain	Yes
Custom-Scheme and Deep-Links	Distrust QR code content with unknown schemes	No, conflict A_3

URL view. Similarly, the mitigation strategy *Distrust QR code content with unknown schemes* requires users to inspect the URL scheme, which is likewise only visible through the expanded URL view. Consequently, both mitigation strategies are incompatible with assumption A_3. This led to the formulation of the following security assumptions: $\mathbf{A_5}$. *If the readers of QRNope scan a QR code containing a URL Redirect attack, no harm comes to them, because developers are capable of dealing with such attack techniques at the technology level.* $\mathbf{A_6}$. *If the readers of QRNope scan a QR code containing a Custom Scheme and Deep-Link attack, no harm comes to them, because developers are capable of dealing with such attack techniques at the technology level.*

Secondly, one mitigation strategy (highlighted in magenta in Table 5.7) is deemed too complex. This is called a *too complex mitigation strategy* because it requires a level of technical knowledge

higher than that of a lay user and would therefore be too difficult to understand effectively within the available time. Specifically, understanding the mitigation strategy *Deny unnecessary permissions* requires readers to understand application permissions and determine which permissions are necessary for the intended functionality of a QR code scanner. However, the corresponding attack technique (i.e., "Malicious QR code scanner") is still addressed by at least another mitigation strategy that is actionable by the target population (i.e., "Only use the official app store"). As such, even if this mitigation strategy is non-actionable, no new assumption is required regarding the corresponding attack technique.

Thirdly, Table 5.7 highlights in cyan mitigation strategies that are too limiting to be applicable. Namely, *unreasonable mitigation strategies* are those mitigation strategies that require users to avoid whole types of necessary everyday interactions (e.g., opening online documents) based on the chance that they could contain malicious content. Three attack techniques (i.e., "Standalone," "External Content Manipulation," and "QRL-Jacking") are solely addressed by an unreasonable mitigation strategy. The mitigation strategy associated with the *Standalone* attack technique requires users to avoid scanning standalone QR codes altogether. Similarly, the mitigation strategy associated with the *External Content Manipulation* attack technique requires users to avoid opening user-generated online documents. Lastly, the mitigation strategy associated with the *QRL-Jacking* attack technique requires users to avoid authenticating to services through QR code login. In all three cases, the mitigation strategy requires users to avoid an entire type of everyday interaction rather than providing guidance on how to perform it securely. This led to the formulation of the following security assumptions: $\mathbf{A_7}$. *If the readers of QRNope scan a Standalone malicious QR code, they are able to thwart any payload encoded in such QR codes through the mitigation strategies covered by QRNope, or the payload is addressed through the corresponding security assumptions.* $\mathbf{A_8}$. *If the readers of QRNope scan a QR code containing a QRL-Jacking attack, no harm comes to them, because system administrators and developers are capable of dealing with such attack techniques at the technology level.* $\mathbf{A_9}$. *If the readers of QRNope scan a QR code containing an External Content Manipulation attack, no harm comes to them, because system administrators are capable of dealing with such attack techniques at the technology level.*

Lastly, there are constraints connected to how QR codes are expected to be used, i.e., a type of technology meant to both simplify and speed up the exchange of information, the latter being provided directly after scanning a QR code without further steps from the users. Thus, *interruption mitigation strategies* are those mitigation strategies (highlighted in blue in Table 5.7) that require users to interrupt their interaction with a QR code (e.g., copying a short URL into a URL expansion service). Interruption mitigation strategies are unlikely to be followed because they are at odds with the intended use of QR code technology. Hence, readers should not be expected to apply interruption mitigation strategies, which should instead be the onus of developers and the

technology itself. For example, in the case of short URLs, the iOS Camera should directly show the URL hidden behind the short URL, as suggested for other apps, e.g., by Dudheria (2017), and as done within the SEQR app in Mossano et al. (2026b).

Nevertheless, a different consideration is given to the interruption mitigation measure "Always look for contacts yourself". In this case, even if the interaction with the QR code is indeed interrupted, the final outcome is the same, i.e., the users do not need to scan the QR code again to obtain the contact information, but rather they can directly call the phone number found online.

The remaining interruption mitigation strategies are associated with attack techniques that are still addressed by at least one other actionable mitigation strategy. Consequently, the only attack technique solely addressed by an interruption mitigation strategy (i.e., "URL Shortener") is not addressed by QRNope.

This led to the formulation of the following security assumption: $\mathbf{A_{10}}$. *If the readers of QRNope scan a QR code containing a URL shortener attack, no harm comes to them, because developers are capable of dealing with such attack techniques at the technology level.*

5.2.3.5 Attack techniques with actionable mitigation strategies to consider

As described in Section 5.2.3.4, four attack techniques are addressed through security assumptions and are therefore not covered by QRNope. Table 5.8 shows the remaining attack techniques that are remaining.

5.2.3.6 Security Assumptions (End of *Step 3*)

This section contains the security assumptions formulated at Step 3 after the determination of the attack techniques with actionable mitigation strategies. As described in Section 5.2.1.8, each assumption is followed by a description of the rationale behind it, as suggested in Van Landuyt and Joosen (2022).

$\mathbf{A_5}$. *If the readers of QRNope scan a QR code containing a redirection URL attack, no harm comes to them, because developers are capable of dealing with such attack techniques at the technology level.*

A_5 is based on the conflict with A_3, i.e., it is assumed that readers will not use the extension function of the iOS Camera as doing so is likely to confuse them. Unfortunately, the only viable way for a user to determine the destination URL targeted by a redirection URL is to check the path of the redirection URL, meaning that there is no way for a user to easily check the

Table 5.8: Attack techniques with actionable mitigation strategies remaining.

Attack technique	Mitigation strategy
Embedded	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext
Self created	Anyone can create and disseminate QR codes
Altered	Always check QR codes for tampering
Cloned	
Added	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext
Replaced	Any pretext can be copied, so a QR code legitimacy cannot be based on its pretext
Malicious QR code generator	Only use the official app store
Malicious QR code scanner	Only use the official app store
Malicious Phone Number	
Impersonation	Always look for contacts yourself
	Never use the contacts provided in the QR code or pretext
USSD Code Exploitation	Always look for contacts yourself
	Never use the contacts provided in the QR code or pretext
Premium Rate Service Exploitation	Always look for contacts yourself
	Never use the contacts provided in the QR code or pretext
Specific App MQRC	
Fake payment	Always check the receiver of a payment
Malicious URL	
Mismatch	
Entire URL in Pretext	Never trust URLs in pretexts, always check what the app shows
Part of the URL in Pretext	Never trust URLs in pretexts, always check what the app shows
Unrecognizable domain	
IP Address	Stop when IP address
Random Characters	Stop when registered domain not plausible
Use of Fitting Keywords	Distrust URLs with buzzwords
Domain Extension	Distrust URLs with buzzwords
Subdomain	Always focus on the registered domain
Different Top-Level-Domain	Carefully read the URL and make sure that its last part corresponds to the legitimate one
Mangle	Carefully read the registered domain

destination of a redirection URL without checking the full URL. However, redirection URLs attacks should be addressed at the technology level, by integrating functions that directly present the destination URL to the users, as done in, e.g., the SEQR tool presented in Mossano et al. (2026b). Similarly to A_4, a trust assumption is formulated, delegating the responsibility for addressing this attack technique to developers rather than users. As such, the attack technique is assumed to be addressed.

A_6. *If the readers of QRNope scan a QR code containing a Custom Scheme and Deep-Link attack, no harm comes to them, because developers are capable of dealing with such attack techniques at the technology level.*

A_6 follows a similar rationale to A_5. A_6 is in conflict with A_3, because the iOS Camera visualization removes the scheme of all URIs[3], which can only be restored by checking the extended view function, thus risking confusing the users. Following the rationale behind A_4 and A_5, A_6 is another trust assumption that assumes developers can deal with dangerous schemes at the

[3] A Uniform Resource Identifier (URI, defined in Berners-Lee et al. (2005) and updated in Nottingham (2020)) is a string identifying a resource or action target; URLs are a subtype of URIs, as are application-specific schemes such as `mailto:` or `whatsapp:`.

technology level by warning the users and stopping them from opening such URIs. As such, the attack technique is assumed to be addressed.

A_7. *If the readers of QRNope scan a Standalone malicious QR code, they are able to thwart any payload encoded in such QR codes through the mitigation strategies covered by QRNope, or the payload is addressed through the corresponding security assumptions.*

A_7 is based on the fact that any payload encoded in an embedded QR code can be just as well encoded in a standalone QR code. Even though a standalone QR code relies on a different hook than an embedded one, as described in Section 4.1.3, the encoded payload is nonetheless bound by the same technology. Furthermore, if the payload corresponds to an attack technique assumed to be addressed through one of the security assumptions, then the corresponding assumption equally applies to the Standalone QR code. As such, any mitigation strategy covered by QRNope (or assumed so) capable of thwarting a payload encoded in an embedded QR code could be just as well used to thwart a standalone QR code, and no additional mitigation strategies are required.

A_8. *If the readers of QRNope scan a QR code containing a QRL-Jacking attack, no harm comes to them, because system administrators and developers are capable of dealing with such attack techniques at the technology level.*

A_8 originated from the *QRL-Jacking* attack technique. This attack takes advantage of QR code login mechanisms in which a session token is encoded in a QR code and scanned by the user to authenticate a session. The attacker presents the user with a QR code containing the session token of an attacker-controlled session. By scanning that QR code, the user authenticates the attacker's session, allowing the attacker to hijack it. Users have no effective way to prevent the attack other than avoiding QR code login altogether, as its mitigation depends on how the QR code login mechanism is implemented by the corresponding service. For example, by implementing token validation. As such, as done in A_4, a trust assumption is formulated, and the burden of addressing this attack technique is left to the developers and system administrators.

A_9. *If the readers of QRNope scan a QR code containing an External Content Manipulation attack, no harm comes to them, because system administrators are capable of dealing with such attack techniques at the technology level.*

A_9 follows a similar rationale to A_4, A_5, A_6, and A_7. As described in Section 5.2.3.4, the mitigation strategy identified for External Content Manipulation was considered unreasonable for the target population, as they cannot avoid all online content. The system administrators of each online platform that offers hosting of user-generated content should implement mechanisms that can identify and remove malicious content without relying on the users to do so themselves. Thus, A_9 is another trust assumption that assumes online platforms deal with malicious content at the technology level. As such, the attack technique is assumed to be addressed.

A_{10}. *If the readers of QRNope scan a QR code containing a URL shortener attack, no harm comes to them, because developers are capable of dealing with such attack techniques at the technology level.*

A_{10} follows a similar rationale to A_6. As described in Section 5.2.3.4, the mitigation strategy identified for URL shorteners was considered an interruption mitigation strategy. However, just like redirection URLs, shortened URLs should be addressed at the technology level, for example by directly presenting their destination URLs to users, as done, e.g., in Mossano et al. (2026b). Thus, A_{10} is another trust assumption that assumes developers deal with URL shorteners. As such, the attack technique is assumed to be addressed.

The security assumptions are then added to those formulated in Step 2, and shown in Table 5.9.

Table 5.9: Attack techniques assumed to be addressed at the end of Step 3.

Attack technique	**External mechanism**	**Assumption**
Standalone	Remaining mitigation strategies	A_7
Specific App MQRC		
QRL-Jacking	System administrators and developers	A_8
Injection attack	Input validation and sanitization	A_4
Malicious URL		
Hexadecimal encoding	URL visualization	A_2, A_3
External Content Manipulation	System administrators	A_9
URL Path/Query/Fragment		
Domain in Path	URL visualization	A_2, A_3
Entire URL in Path	URL visualization	A_2, A_3
Luring-Label with Domain	URL visualization	A_2, A_3
Luring-Label with URL	URL visualization	A_2, A_3
Look-Alike Domain	URL visualization	A_2, A_3
Omitting Slash	URL visualization	A_2, A_3
UserInfo Field	URL visualization	A_2, A_3
URL Shortener	Developers	A_{10}
URL Redirect	Developers	A_5
Homographic Spoofing	URL visualization	A_2, A_3
Exceedingly Long URL	URL visualization	A_2, A_3
Custom-Scheme and Deep-Links	Developers	A_6

Given the assumptions, Table 5.8 contains the attack techniques and the mitigation strategies remaining at the end of Step 3.

Note: No further security assumptions are added after this point. As such, this is the final set of security assumptions.

5.2.4 Step 4 – Attack techniques, mitigation strategies, and misconceptions to cover

This section covers *Step 4* of the development process (shown in Figure 5.5). Namely, it covers the awareness of the target population (Section 5.2.4.1), the attack techniques with actionable mitigation strategies to cover and the actionable mitigation strategies themselves (Section 5.2.4.2). The misconceptions on attack techniques (Section 5.2.4.3) and on mitigation strategies (Section 5.2.4.4) to cover are then determined.

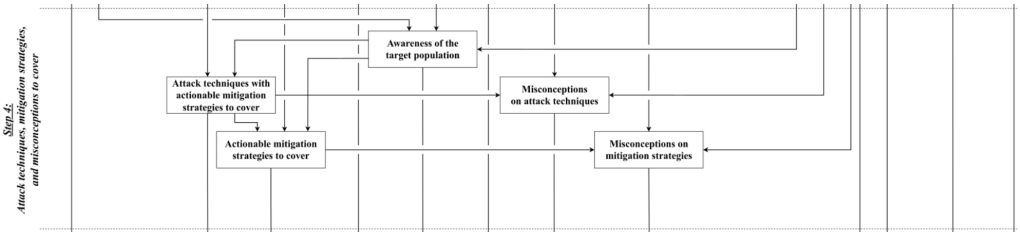


Figure 5.5: *Step 4* of the framework for developing awareness interventions.

5.2.4.1 Awareness of the target population

Given that QRNope was intended to be effective for the least-aware members of the target population (as described in Section 5.2.1.2), the awareness of the target population was set to the lowest possible level, i.e., no previous QRishing or general phishing awareness was assumed. In other words, the starting awareness was lower than the lowest level of the awareness maturity curve, i.e., the target population had no information about QRishing. As such, no attack technique or mitigation strategy was assumed to be already known based on the previous awareness of the target population. This decision was also consistent with the findings of Chapter 2, where participants demonstrated a generally low awareness of QRishing and of the actions to take to interact securely with QR codes.

5.2.4.2 Attack techniques with mitigation strategies to cover and Actionable mitigation strategies to cover

Given that no attack techniques nor mitigation strategies were further assumed in Section 5.2.4.1, the attack techniques with actionable mitigation strategies to cover and the mitigation strategies to cover themselves remained those identified at the end of Step 3 (Section 5.2.3.5).

5.2.4.3 Misconceptions on attack techniques

The misconceptions on attack techniques identified in Section 4.2.5 were then examined in light of the security assumptions defined in the previous steps and according to possible redundancies between misconceptions.

Addressing misconception B, i.e., "Visiting a website without checking the URL in the QR code is safe", required making readers aware that no website should be visited without first checking the URL in the QR code scanner. This also addressed misconception 2, i.e., "If a website is known to the user or looks official, it is safe to enter credentials". Misconception 2 assumed that users had already reached the website. If readers were made aware that the URL should be checked before visiting the website, then the earlier step of the attack was addressed, and misconception 2 did not require separate coverage.

Misconception 5, i.e., "If a URL shows HTTPS it means that it is safe to click the link", assumed that the scheme was shown to users, who then misinterpreted HTTPS as a clue of legitimacy. However, the main URL visualization of the iOS Camera did not show the scheme, which was only accessible through the extended view. As stated in assumption A_3, readers were assumed not to use such function, meaning that they would never see the HTTPS scheme. As such, misconception 5 did not require coverage in QRNope under assumption A_3.

Table 5.10 shows the misconceptions on attack techniques from Section 4.2.5, whether they required coverage in QRNope, and the rationale when they did not.

The final list of misconceptions on attack techniques to cover in QRNope was:

- A. If a poster looks harmless it is safe to visit the QR code.
- B. Visiting a website without checking the URL in the QR code is safe.
- 6. If a familiar name is in the URL then the URL is safe.
- 7. If the sender of a message is someone from the social circle of the user, the message is trustworthy.

Table 5.10: Misconceptions on attack techniques, whether they require coverage in QRNope, and rationale for those that do not.

Misconception	Requires coverage	Rationale
Misc. A	Yes	–
Misc. B	Yes	–
Misc. 2	No	Addressed by misconception B
Misc. 5	No	Not relevant under assumption A_3
Misc. 6	Yes	–
Misc. 7	Yes	–

5.2.4.4 Misconceptions on mitigation strategies

Similarly, the misconceptions on mitigation strategies identified in Section 4.2.5 were then examined in light of the security assumptions defined in the previous steps and according to possible redundancies between misconceptions.

Misconception 9, i.e., "SPAM and phishing messages are the same" seemed at first sight relevant, because SPAM and phishing require different mechanisms to be taken care of (as highlighted in Section 4.2.4.9). However, considering that the recommendation in that case would be make users aware of where to report the phishing attack (and by extension, QRishing attacks), it was possible to merge Misconception 9 with Misconception 15, i.e., "Reporting phishing is not a user's duty". In other words, if when addressing misconception 15, it was also mentioned where to report, highlighting that SPAM and phishing are different became relatively inconsequential, security-wise. As such, misconception 9 did not require separate coverage, as long as misconception 15 also addressed to whom and how to report QRishing.

Misconception 11, i.e., "Cybersecurity warnings always identify phishing or anyway dangerous messages" was not relevant for QRNope due to the context declared in Section 5.2.1.3. As stated in assumption A_1, the readers of QRNope only used the iOS Camera to scan QR codes, which did not provide any warning or other feedback regarding the legitimacy of a QR code. Thus, misconception 11 would not have triggered, given that no warning would have been shown in the first place. As such, misconception 11 was deemed not relevant under assumption A_1.

The list of misconceptions on mitigation strategies covered in QRNope was:

- 1. Strong passwords are enough to protect from phishing.
- 10. Technical solutions (e.g., filters, anti-virus) always protect from phishing.
- 13. Enabling anti-tracking in the smartphone options protects from phishing.

- 14. Blocking cookies protects from phishing.
- 15. Reporting phishing is not a user's duty.

5.2.5 Step 5 – Content creation

This section covers *Step 5* of the development process (shown in Figure 5.6). Specifically, it covers the incentives for the target population to engage with the awareness intervention (Section 5.2.5.1), and the content of the awareness intervention (Section 5.2.5.2).

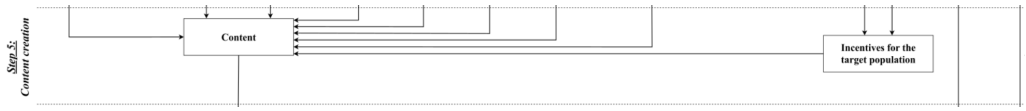


Figure 5.6: *Step 5* of the framework for developing awareness interventions.

5.2.5.1 Incentives for the Target Population

As the method to incentivize the target population to engage with QRNope, it was decided to use a common method used in security: "fear appeals", i.e., creating a situation that elevates the perceived threat of a population to shock them into compliance (as explained in, e.g., Dupuis and Renaud (2021)). However, Renaud and Dupuis (2019) suggest that, in accordance with the "Protection Motivation Theory" (hereafter, *PMT*) from Rogers (1975), any fear appeal is only effective if followed by an actionable mitigation applicable by the target population, i.e., a way for the target population to reach a self-efficacy level high enough to feel capable of reacting to the object of the fear appeal. Albeit this seems a reasonable approach, Witte (1992) points out that, of the constructs of the PMT model (Severity, Vulnerability, Response Efficacy, Self-Efficacy, Rewards, and Response Cost, leading to Threat Appraisal and Coping Appraisal), none considers the actual emotional response, i.e., the effect of fear.

As such, a different model was selected as the theoretical model for fear appeals within QRNope: the *Extended Parallel Process Model* (EPPM, originally from Witte (1992)). The EPPM is based on the *Parallel Process Model* (hereafter, *PPM)* from Leventhal (1971), i.e., a framework showing that a change in the protective behavior of an individual comes from danger control. The PPM was then extended by the PMT constructs Threat Appraisal and Coping Appraisal. Figure 5.7 shows an illustration of how the various constructs interact with each other.

Within the EPPM, any external stimulus with a fear appeal (e.g., an awareness intervention) has four components: (1) *self-efficacy* (i.e., how competent an individual feels to respond to a threat),

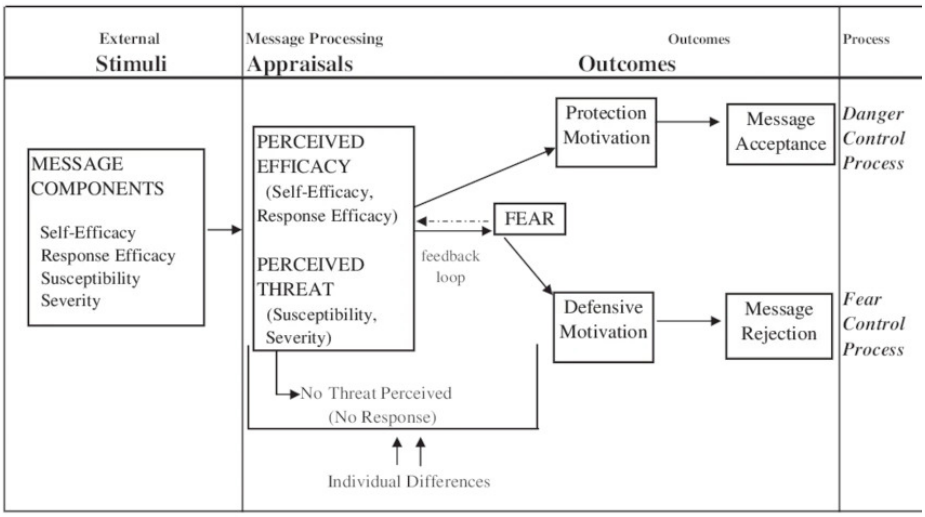


Figure 5.7: The Extended Parallel Process Model from Witte (1992) (image from Gore and Bracken (2005)).

(2) *response efficacy* (i.e., how the actions recommended can successfully control the risk of the threat), (3) *susceptibility* (i.e., how likely a threat is to impact an individual), and (4) *severity* (i.e., how impactful a threat is on an individual).

The fear appeal itself initiates two appraisals, i.e., two cognitive processes using two of the four components each, leading to two "perceptions": *perceived efficacy* (self-efficacy and response efficacy), and *perceived threat* (susceptibility and severity). First, an individual appraises the perceived threat of a hazard. When the perceived threat is low, the stimulus fails to motivate the individual to process the stimulus itself further, leading to a *No response* outcome, i.e., the cessation of the interaction with the stimulus.

Conversely, when the threat appraisal results in moderate to high perceived threat, then the stimulus succeeds in eliciting fear in an individual. The fear then motivates the initiation of the second appraisal. The outcome of the second appraisal is one of two mutually exclusive processes:

- *Fear control.* When the perceived threat is high, but the perceived efficacy is deemed insufficient to answer it, then the fear originally evoked is increased. As such, an individual tries to cope with their fear (*defensive motivation*), instead of answering to the threat. This causes the rejection of the threat and the message (e.g., denial).
- *Danger control.* When the perceived threat is high, and that perceived efficacy is deemed sufficient to address it, then an individual is motivated to engage in *danger control* processes.

This leads to either the development of strategies to avert the threat, or the application of the recommended mitigation strategies provided to them.

The EPPM informs the development of QRNope in several ways. First and foremost, to avoid a *No response* output, QRNope clearly states how malicious QR codes could be encountered in the readers' everyday life (e.g., parking payments), how easy it is for attackers to spread malicious QR codes, and how serious the effect of a malicious QR code could be on the readers' personal life (e.g., fraud, device infection, identity theft). To avoid a *Fear control* process, QRNope unequivocally states that by following the recommendations contained, the readers can avoid the most likely attacks they would encounter. The likelihood of a *Danger control* response is further increased by i) using a language understandable by the readers (see Section 5.2.5.2 on how the readability is calculated), ii) employing explanatory images that facilitate understanding, and iii) keeping the number of mitigation strategies to a contained number by grouping them in seven rules applicable without previous specialized IT knowledge.

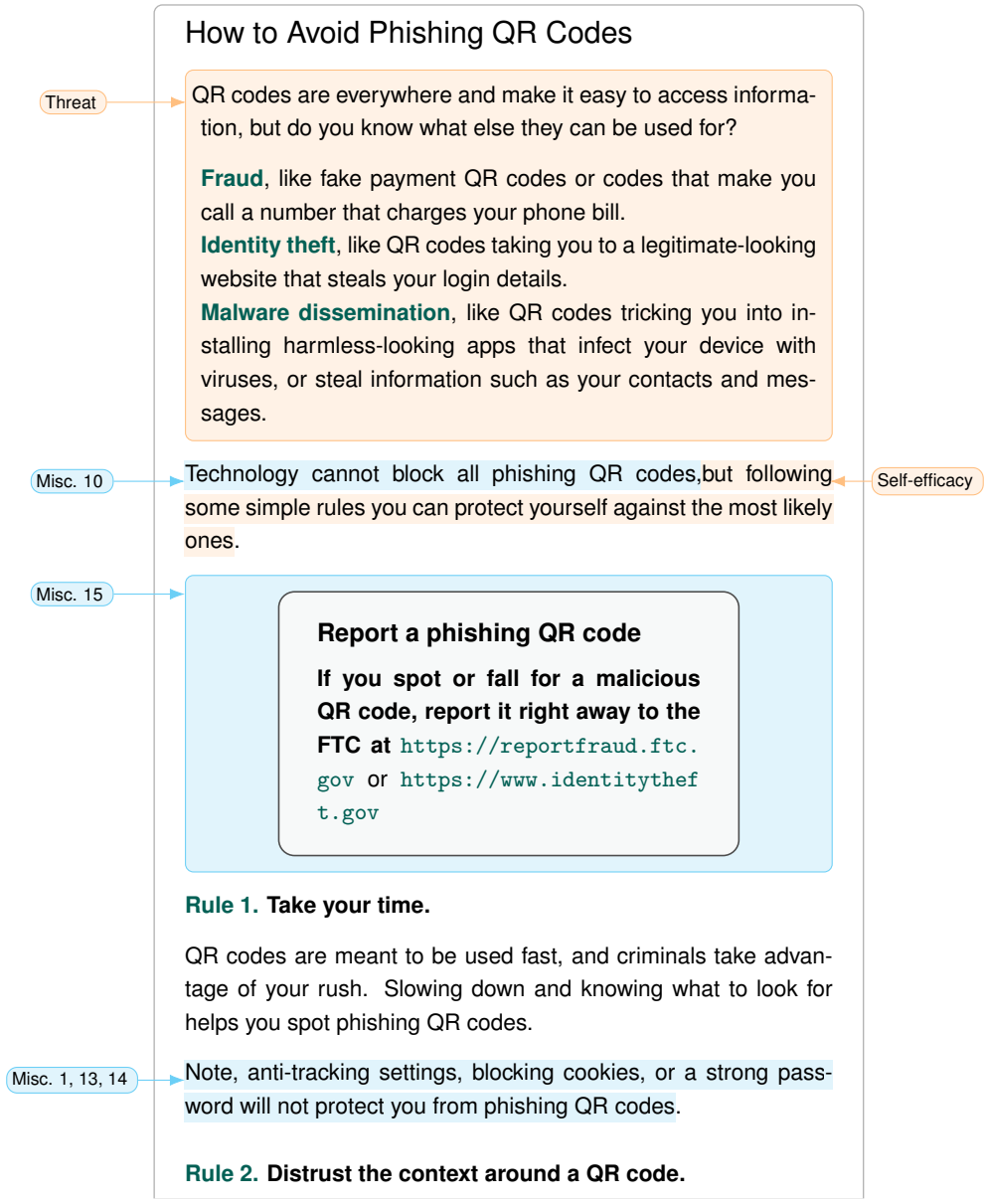
5.2.5.2 Content

This section presents the initial version of the QRNope content. First, the full content is shown, with highlights showing which parts address which attack / EPPM construct / misconception. Afterwards, the readability of QRNope is calculated, and the reading time estimated again to compare it against the goal set in Section 5.2.2.1.

Note: Although the content itself remains static after this stage (i.e., no new attack techniques or mitigation strategies are added), the way it is presented, as well as the information depth, changed due to the feedback received from both experts and target population. These changes are covered in Section 5.2.7. As such, what is shown in this section is the content used in QRNope v. 1.0.

5.2.5.2.1 Full content. The suggested problem-solution ordering described by Fuller et al. (2011) indicates that learners are more motivated and effective at acquiring knowledge when presented with the problem before the solution (or $problem \rightarrow solution$). In the case of QRNope, however, this ordering might not work as intended. Some of the Rules (e.g., *Rule 2*) require presenting a considerable amount of information about the problem before introducing the solution, and the lengthy threat description might increase the likelihood of a fear-control response (see Section 5.2.5.1). Therefore, QRNope instead follows a $rule \rightarrow problem \rightarrow rule$ structure to introduce an initial self-efficacy signal before describing the threat.

Figure 5.8 shows the content, with attack techniques highlighted in olive, misconceptions on attack techniques in magenta, misconceptions on mitigation strategies in cyan, and EPPM constructs in orange.

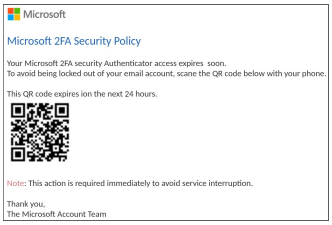


Embedded
Added
Replaced

No QR code is safe until checked, even if (apparently) from your bank, a friend, or a family member. Anyone can create a QR code. Any design can be copied, any sender of a message faked, any QR code added or replaced.

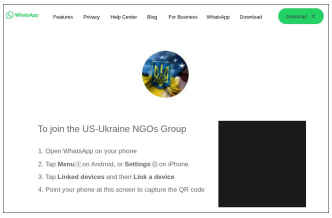
Misc. 7

Self-created



Criminals have no shame exploiting important events you care about, tragic ones too. Even harmless or well-intended messages and posters can hide a phishing QR code.

Misc. A



Altered

Always check for signs of tampering. If you feel unsure about the QR code legitimacy, do not scan the QR code.



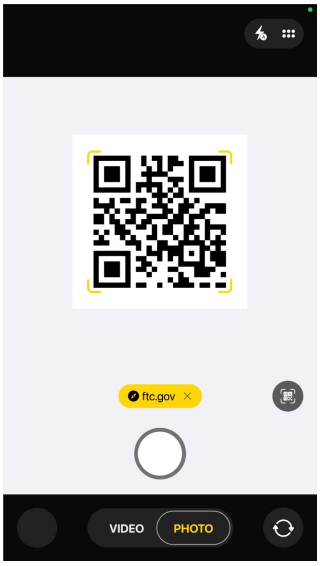




Rule 3. Carefully check the domain before visiting the webpage.

If a QR code contains a URL, the *domain* (that is, where you are going) is shown at the bottom of your camera app. Never trust the URL printed next to a QR code, but always check what your camera app shows. Here is a scanned QR code leading to ftc.gov.

Mismatch



Always read the domain carefully. It can tell you if a QR code is phishing or not. Some phishing domains are easily recognizable:

Random → aldslkskmskoewlc.at

Puny code (as random) → ftc.xn–gv-fmc

IP address → 142.250.74.206

Other phishing domains look more convincing, using familiar names to trick you, and need a closer inspection: ← Misc. 6

Subdomain → ftc.information.com (domain is information.com)

Different TLD → ftc.com (.com instead of .gov)

Sometimes the difference is just one letter or adding an extra letter:

Mangle → flc.gov (l instead of t)

Mangle → fttc.gov (double t)

A phishing domain may also use reassuring or fitting words to trick you:

Extension → secure-ftc.com

Fitting words → fraud.support

Extension → report-to-ftc.com

Misc. B → You should never visit a website without checking the domain. If the domain is not exactly what you expect, stop and do not visit the URL.

Rule 4. Check the recipient of your money transfer.

Criminals might replace, add or modify payment QR codes in shops or other places, such as parking meters..

Fake payment → Before sending money or paying through a QR code, always check that the recipient is correct.

Rule 5. Beware QR code generators and scanners not from the App Store.

Criminals create malicious QR code generators and scanners, and wait for people to use them.

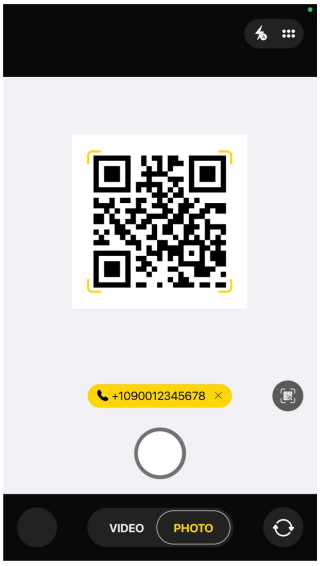
A website offering free QR code generation may secretly replace your URL with one controlled by a criminal or with their phone number. A malicious QR code scanner may install malware.

QR code gen.
QR code scanner

It is safer to use well-known apps from the official App Store. Some malicious apps may slip through, but security checks usually remove them before they become widely used.

Rule 6. Never call phone numbers from QR codes of unsure origin.

QR codes can contain phone numbers, but you should never call such numbers.



Criminals may trick you into calling premium phone numbers that greatly increase your phone bill. In other cases, the phone number may forward your future calls to the criminals.

Malic. phone num.

In most cases, it is safer to look up the contact information online.

Rule 7. When in doubt, do not scan the QR code.

You can always ask for a paper menu or look up the information online yourself.

> QR codes are meant to make things easier, but if you are unsure, you risk your safety just to save some seconds.

Figure 5.8: First version of the QRNope content.

5.2.5.2.2 Readability estimation. This paragraph covers the process followed to estimate the readability of the content of QRNope.
As stated in Section 5.2.2.1, one goal is for QRNope to be readable within five minutes. To this end, a new estimate of the reading time is conducted at this stage, using the same methodology as before. This estimate puts the reading time for an English speaker at 2 minutes and 58 seconds (rounded up to 3 minutes), hence two minutes lower than the acceptable five minutes threshold. However, reading time is but one of the metrics. As stated in Section 5.2.3.3, the reading level for QRNope was set at approximately a 7th-grade reading level. To estimate whether the language of QRNope meets the reading-level constraint, two measurements were employed:

- Flesch Reading Ease Score (FRES). Originally published in Flesch (1979), it measures on a scale from 0 to 100, with 100 as the best result, how easy it is to read a text in English.
- Flesch-Kincaid Grade Level Formula (FKGLF). Published in Kincaid et al. (1975), it is a modified version of the FRES that directly returns the minimum USA education grade level necessary to understand the content of a text.

These two measurements were not implemented directly, but rather obtained using the aforementioned koRpus R package (also used to calculate the number of words in Section 5.2.2.1). The calculated FRES was 75.57, on a maximum of 100, indicating that the text is "fairly easy" to read. Conversely, the FKGLF value returned was 5.02, i.e., a fifth-grade US education level is sufficient to understand the text, meaning that QRNope is understandable without a particularly high educational level. Technical terminology is also kept extremely limited (i.e., URL, domain, phishing), but not completely absent. This was a design choice so that, if a reader was to search online for further information, they would find the exact results for the terminology searched (e.g., on Wikipedia).

5.2.6 Step 6 – Intervention implementation

This section covers *Step 6* of the development process (shown in Figure 5.9). Namely, it covers the adaptation of the content developed in Step 5 into a deployable HTML webpage, including the implementation of the accessibility requirements identified previously (Section 5.2.6.1), the sanitization review conducted before deployment (Section 5.2.6.2), and the resulting version of the intervention (Section 5.2.6.3).

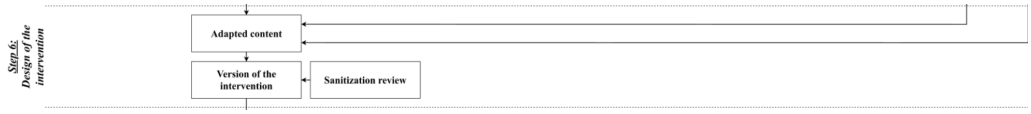


Figure 5.9: *Step 6* of the framework for developing awareness interventions.

5.2.6.1 Adapted Content

The implementation of the content developed in *Step 5* does not modify the content, structure, or sequencing developed previously, but rather provides a deployable webpage version of the intervention. The webpage does not rely on external scripts, stylesheets, or third-party resources beyond locally referenced images.

Note: Throughout this section, references to WCAG 2.2 refer to the *Web Content Accessibility Guidelines* version 2.2, accessible at Campbell et al. (2024). References to WAI-ARIA refer to the *Accessible Rich Internet Applications* specification version 1.2, accessible at Diggs et al. (2023).

Readability of the content was previously evaluated in Section 5.2.5.2. As the HTML implementation did not introduce changes to the wording, structure, or ordering of the content, the results of that evaluation remain unchanged. Accordingly, the content continues to satisfy the readability requirements identified during the design process.

Particular attention was given to the accessibility requirements identified in Section 5.2.1.7. Regarding the accessibility requirements related to assistive technologies, the webpage structure is implemented using semantic HTML5 elements, including `main`, `article`, `header`, and `section`, together with a hierarchical heading structure. This organization allows assistive technologies to identify the relationships between different parts of the content and is consistent with WCAG 2.2 Success Criterion 1.3.1 (Info and Relationships). Furthermore, the webpage declares English as its primary language through the `lang="en"` attribute, consistent with WCAG 2.2 Success Criterion 3.1.1 (Language of Page).

Regarding the accessibility requirements related to non-text content, all images are embedded using the `figure` element and associated with captions through `figcaption`. Furthermore, all images contain descriptive alternative text through the `alt` attribute, providing text alternatives for non-text content in accordance with WCAG 2.2 Success Criterion 1.1.1 (Non-text Content). Similarly, abbreviations are identified through the `abbr` element and expanded using the `title` attribute, consistent with WCAG 2.2 Success Criterion 3.1.4 (Abbreviations).

Regarding the accessibility requirements related to visual presentation, the webpage includes both a high-contrast mode and an enhanced readability mode. Color combinations were selected to satisfy the contrast requirements of WCAG 2.2 Success Criterion 1.4.3 (Contrast Minimum), while the high-contrast mode additionally satisfies WCAG 2.2 Success Criterion 1.4.6 (Contrast Enhanced). Links are visually distinguishable through both color and underlining, ensuring that

color is not the sole means of conveying information, in accordance with WCAG 2.2 Success Criterion 1.4.1 (Use of Color).
The layout is responsive and adapts to different viewport sizes. Images scale proportionally to the available space, content reflows without loss of information or functionality, and accessibility modes reduce line length and increase spacing. These measures are consistent with WCAG 2.2 Success Criterion 1.4.10 (Reflow) and with the readability requirements identified previously. Furthermore, text is left-aligned throughout the intervention, avoiding justification-related spacing variations and aligning with WCAG 2.2 Success Criterion 1.4.8 (Visual Presentation).
Regarding the accessibility requirements related to navigation and interaction, accessibility-related functionality is implemented through native HTML controls wherever possible. Interactive functionality is limited to controls for the accessibility modes, implemented through standard HTML buttons. Their current state is exposed through the `aria-pressed` attribute, consistent with WCAG 2.2 Success Criterion 4.1.2 (Name, Role, Value) and the WAI-ARIA 1.2 specification. Keyboard accessibility is provided through standard browser behavior, satisfying WCAG 2.2 Success Criterion 2.1.1 (Keyboard), while visible focus indicators are provided for interactive elements, consistent with WCAG 2.2 Success Criterion 2.4.7 (Focus Visible).
A skip-navigation link is provided at the beginning of the document, allowing users to bypass repeated content and move directly to the main content area. This implementation satisfies WCAG 2.2 Success Criterion 2.4.1 (Bypass Blocks). In addition, any motion-related behavior is disabled when users indicate a preference for reduced motion, consistent with WCAG 2.2 Success Criterion 2.3.3 (Animation from Interactions). No form inputs are present within the intervention and therefore criteria related to input assistance are not applicable.
Beyond the accessibility requirements identified in Section 5.2.1.7, alignment with WCAG 2.2 also facilitates compliance with several commonly used accessibility standards and regulatory frameworks. The *ISO/IEC 40500:2025* standard (accessible at International Organization for Standardization (2025)) corresponds to WCAG 2.2, meaning that alignment with WCAG 2.2 also aligns with ISO/IEC 40500:2025. Conversely, both the European Union standard *EN 301 549* (accessible at European Committee for Standardization (2021)) and the US *Section 508* legislation (accessible at U.S. Access Board (2018)) incorporate earlier WCAG requirements, namely WCAG 2.1 Level A and AA for EN 301 549 and WCAG 2.0 Level A and AA for Section 508. Because WCAG 2.2 is backward compatible with WCAG 2.1, which in turn is backward compatible with WCAG 2.0, conformance with WCAG 2.2 Level AA also entails conformance with the WCAG-derived accessibility requirements incorporated into EN 301 549 and Section 508 within their respective scopes. As such, the webpage satisfies the accessibility requirements derived from all the aforementioned standards and legislation.

5.2.6.2 Sanitization Review

The intervention underwent a sanitization review to ensure that no element of it could expose readers to unnecessary risk.
Regarding the examples used throughout the intervention, all domains and IP addresses were manually verified. Each example was visited and inspected to ensure that it either referred to a legitimate and non-malicious resource (e.g., websites operated by the Federal Trade Commission) or to an inactive resource hosting no content. This review was conducted to prevent the inclusion of examples that could inadvertently direct readers to harmful or otherwise inappropriate content.
Regarding QR codes obtained from third-party sources, all QR codes visible in images were covered to prevent readers from scanning them. This measure was adopted because the content encoded in such QR codes could not be controlled and could potentially change over time.
Regarding QR codes introduced specifically for the intervention, any QR code added by the authors for illustrative purposes was replaced with a self-generated dummy QR code. These QR codes do not direct readers to external resources and simply decode to the text "This is an example QR code" when scanned. This approach preserves the visual appearance of a QR code while eliminating any risk associated with scanning it.
As a result of this sanitization review, all QR codes visible in the intervention are either non-functional examples created specifically for the intervention or have been rendered unscannable. Furthermore, all domains and IP addresses included as examples were verified during the sanitization review. However, sanitization is not a one-time activity. Because the status of external resources may change over time, a mechanism for periodically repeating the sanitization review should be established to ensure that the intervention remains safe to use. The design and implementation of such a mechanism are beyond the scope of this thesis.

5.2.6.3 Version of the Intervention

At this stage, the sanitization review is incorporated into the adapted content, and the first version of QRNope, i.e., QRNope v1.0, is created. The complete HTML implementation of the intervention at this stage is available at https://doi.org/10.5281/zenodo.20927603.

5.2.7 Step 7 – Empirical Evaluation (Feedback stages only)

This section covers *Step 7* of the development process (shown in Figure 5.10). Specifically, it covers the refinement of the intervention through feedback loops between the *Expert feedback on the intervention* stage and the *Informal feedback from target population* stage (Section 5.2.7.1), and the version of QRNope resulting from these feedback loops and evaluated in the next chapter (Section 5.2.7.2).

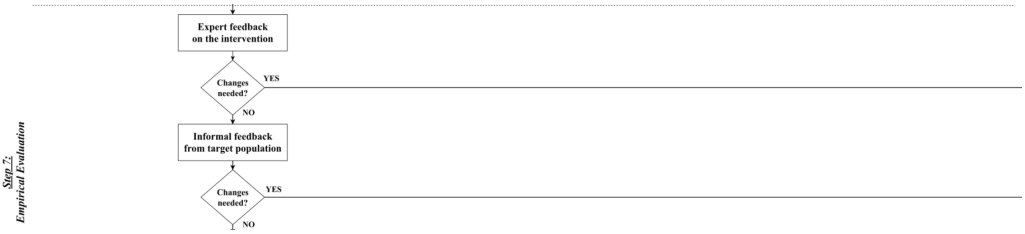


Figure 5.10: *Step 7* of the framework for developing awareness interventions. The evaluation part is omitted, because part of Chapter 6.

5.2.7.1 Expert Feedback and Informal Feedback from the Target Population

After the first implementation, QRNope underwent several feedback loops iterating between the *Expert feedback on the intervention* stage and the *Informal feedback from target population* stage. The goal of these feedback loops was to identify issues affecting the understandability and clarity of the intervention before evaluating its effectiveness. The feedback collected during these iterations, together with the observed difficulties in applying the recommendations, was used to refine how the selected content was presented to the target population.

The first complete version of the awareness intervention, QRNope v1.0, was reviewed by the supervisor and the co-supervisor. Based on their feedback, several refinements were introduced. First, Rule 4 was extended with a visual example of a fraudulent payment request. Specifically, an image of a fake PayPal payment request masquerading as a payment to Wendy's was added to illustrate how attackers may exploit QR codes in payment scenarios. Secondly, the original Rule 5 was divided into two separate rules, one addressing malicious QR code generators and one addressing malicious QR code scanners, allowing each attack technique to be explained independently.

Following the aforementioned refinements, the language of QRNope was examined by language experts from DePaul University (Chicago, IL, US). Several linguistic clarifications and stylistic refinements were introduced to increase the readability of the intervention while preserving the technical correctness of its recommendations. The resulting intervention constituted QRNope v2.0.

QRNope v2.0 was subsequently used in the first feedback loop involving members of the target population. The user study described in Chapter 6 was used as a means to collect feedback from the target population, albeit with some differences compared to the final evaluation of QRNope. The target population was asked to provide feedback on QRNope through an open question and to determine the legitimacy of the same examples described in Section 6.2.4. This allowed collecting both qualitative feedback on QRNope and descriptive data regarding how participants applied the recommendations presented in the intervention. The goal of collecting this data was not to evaluate the effectiveness of QRNope, both due to the number of participants being insufficient to

conduct statistical analyses, and because that was not the objective of the *Informal feedback from target population* stage. Rather, the goal was to identify potential issues in the presentation of the recommendations that could be indicated by the descriptive statistics (alongside the feedback). For example, if the descriptive statistics for one of the pretexts were relatively low, this was interpreted as an indication that the presentation of the corresponding attack technique or recommendation might require further refinement. The first feedback loop did not reveal substantial shortcomings of QRNope, but highlighted potential issues in how some recommendations were presented (e.g., fraudulent QR code payments and phishing domains). For this reason, several refinements were introduced to make the intended guidance more explicit, and the resulting intervention was again reviewed as part of the *Expert feedback on the intervention* stage.
This cyclical refinement process was subsequently repeated until the data collected during the *Informal feedback from target population* stage showed no apparent misunderstandings and approximately 90% of the participants correctly distinguished phishing from legitimate QR codes. This version, i.e., QRNope v5.0, was then used in the empirical evaluation.
Compared to QRNope v1.0, QRNope v5.0 contains additional examples (especially visual ones), more explicit recommendations, and several linguistic clarifications introduced throughout the refinement cycles. Furthermore, it uses several visual elements, e.g., green check marks and red crosses to indicate legitimate domains and phishing domains. Nonetheless, these revisions did not change what content was covered by the intervention, only its presentation. QRNope v5.0 is described in Section 5.2.7.2.

5.2.7.2 QRNope v5.0

This section covers QRNope v5.0, i.e., the final version resulting from the feedback loops and used in the evaluation in Chapter 6. Figure 5.11 shows the whole awareness intervention, while the full implementation is available at https://doi.org/10.5281/zenodo.20927603. Hereafter, every time QRNope is mentioned, it refers to this version.

How to Avoid Phishing QR Codes

QR codes are everywhere and make it easy to access information, but do you know what else they can be used for?

Monetary theft, like QR codes with fake payment or that make you call a number that charges your phone bill.
Identity theft, like QR codes taking you to a legitimate-looking website that steals your login details.
Malware distribution, like QR codes tricking you into installing harmless-looking apps that infect your device with viruses, or steal information such as your contacts and messages.

Technology cannot block all *phishing* QR codes, that is, QR codes containing malicious data aimed at harming whoever access its content.

Following some simple rules, and reporting phishing QR codes, you can protect yourself and others against the most likely ones.

Report a phishing QR code

If you spot or fall for a phishing QR code, report it right away to the FTC at https://reportfraud.ftc.gov or https://www.identitytheft.gov

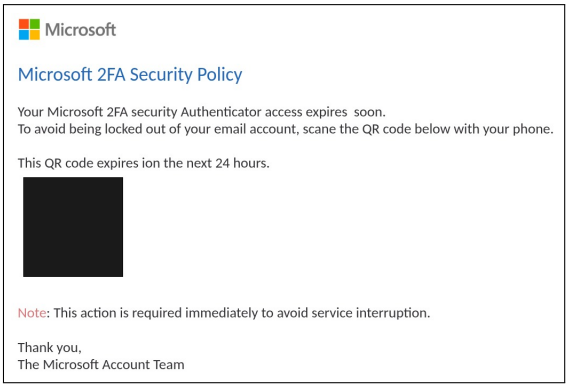
Rule 1. **Take your time.**

QR codes are meant to be used fast, and criminals take advantage of your rush. Slowing down and knowing what to look for helps you spot phishing QR codes.

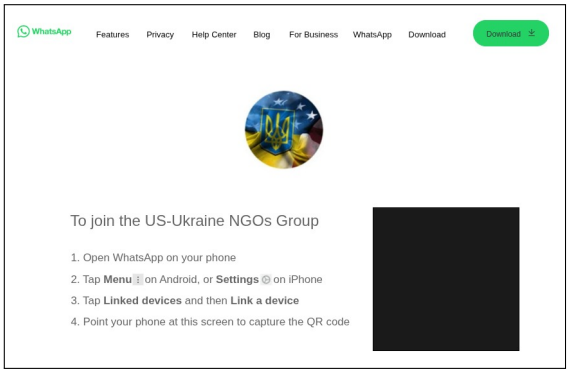
Note: Anti-tracking settings, blocking cookies, or a strong password cannot protect you from phishing QR codes.

Rule 2. **Distrust the context around a QR code.**

No QR code is safe until checked, even if (apparently) from your bank, a friend, or a family member. Anyone can create a QR code and there is no way to tell a phishing QR code from a legitimate one just by looking at it or from its context. Any design can be copied, any sender of message faked, any QR code added or replaced.

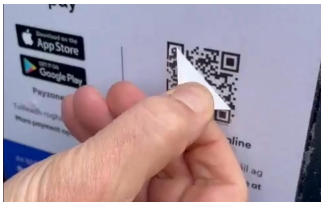


Criminals have no shame exploiting important events you care about, tragic ones too. Even harmless or well-intended messages and posters can hide a phishing QR code.



Always check for signs of tampering. If you feel unsure about the QR code legitimacy, do not scan the QR code.







Rule 3. **Carefully check the domain before visiting the URL.**

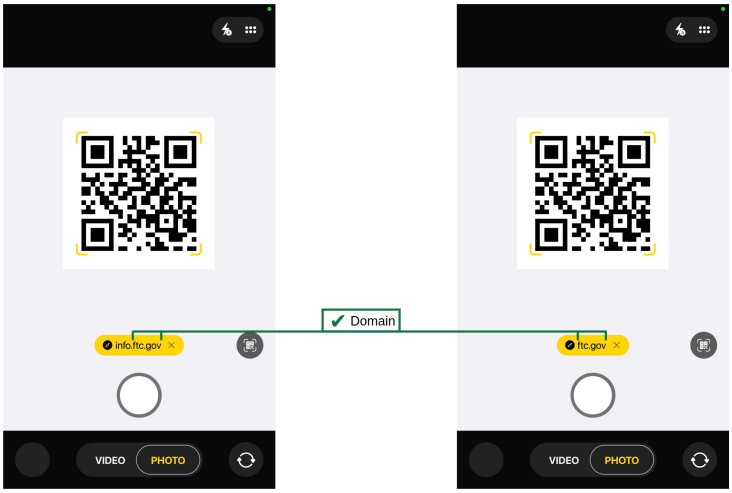
QR codes often contain a *URL* (that is, a web address). **Never** trust the URL printed next to a QR code, but **always** check what your camera app shows.

The *domain* of a URL (that is, where you are going) is shown at the bottom of your camera app.

The domain is usually made of the *two parts divided by a dot* (for example, ftc.gov). If there are three parts (for example, info.ftc.gov), the *two rightmost parts preceded by a dot* are the domain.

Especially important is that it is possible to write whatever one wants in the part before the domain. You should **never** trust this first part, and only focus on the domain.

Here are two scans of two legitimate QR codes leading to **info.ftc.gov** (leftmost) and to **ftc.gov** (rightmost). The green highlighting shows you where the domain is located.



A *phishing domain*, just like a phishing QR code, is a domain that leads to a webpage with content intended to harm anyone who visit it.

Always read the domain carefully. It can tell you if a QR code is phishing or not.

Some phishing domains have random or suspicious names:

✔ ftc.gov
✔ info.ftc.gov
✘ aldslkskmskoewlc.at
✘ ftc.xn–gv-fmc
✘ 142.250.74.206

Other phishing domains look more convincing. They might use different endings, or use a short first part before the domain to trick you:

✔ ftc.gov
✔ info.ftc.gov
✘ ftc.com (.com instead of .gov)
✘ ftc.info.com (short first part tricks you; the domain is info.gov, **not** ftc.info)

Sometimes the difference is just one letter or adding an extra letter:

✔ ftc.gov
✔ info.ftc.gov
✘ flc.gov (l instead of t)
✘ info.fttc.gov (double t)

A phishing domain may also use reassuring or fitting words to trick you, maybe extending the legitimate domain with a "-" (dash) character:

✔ ftc.gov
✔ info.ftc.gov
✘ info-ftc.gov (info-ftc.gov is the domain, **not** only ftc.gov)
✘ report-to-ftc.gov (report-to-ftc.gov is the domain, **not** only ftc.gov)
✘ fraud.support

Never visit a website without checking the domain.

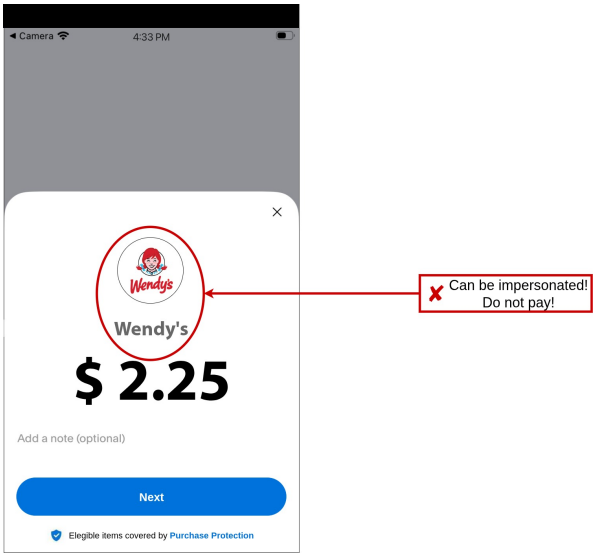
If the domain is not exactly what you expect, because it uses dashes instead of points, or it is in a different order than usual, stop and do not visit the URL.

Rule 4. **Never use direct payment QR codes in apps without impersonation-resistant identifiers.**

QR codes can be used to open an app and directly pay for a service (for example, in a restaurant). It is trivial for criminals to create perfect replicas of legitimate QR codes, and then replace, add or modify the legitimate QR codes.

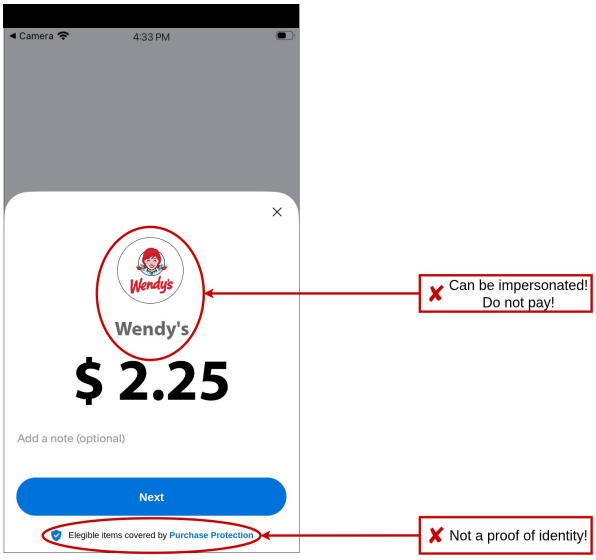
Criminals can use fake names and logos, impersonating whoever they want, even well known organizations. Only certain *impersonation-resistant identifiers* (for example, bank coordinates or social media handles) cannot be impersonated.

If the app you are using to pay shows **no** impersonation-resistant identifier, **never** pay directly through it by scanning a QR code.



If an impersonation-resistant identifier is shown in the app, **always** check that the identifier is the correct one. If you do not know the correct impersonation-resistant identifier, search for it online or ask the cashier.

Note: Images with check-marks, or other small logos that communicate trust are not sufficient proofs of identity.



Rule 5. **Beware QR code generators not from the App Store.**

Criminals create malicious QR code generators and wait for people to use them. A website offering free QR code generation may secretly replace your URL with a URL created by a criminal or with their phone number.

It is safer to use well-known QR code generators from the official App Store, because, even if some malicious apps may slip through, the security checks usually remove them before they become widely used.

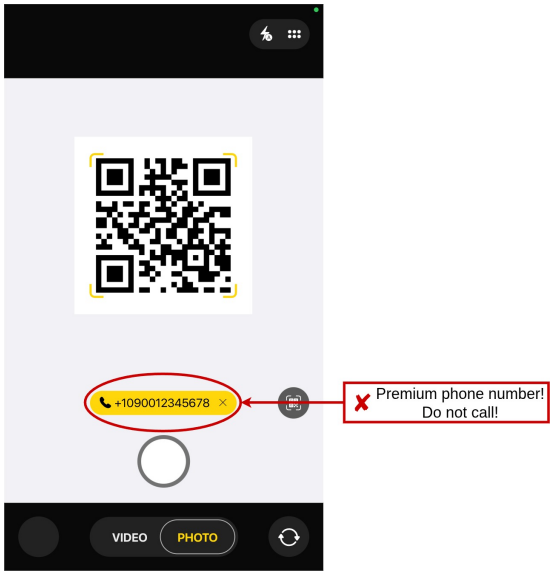
Rule 6. **Beware QR code scanners not from the App Store.**

Criminals develop malicious QR code scanners and then try to convince people to install them. Criminals develop malicious QR code scanners and then try to convince people to install them. Such malicious QR code scanner can contain malware that steals your information, or lock you out of your device.

Just like QR code generators, it is safer to use well-known QR code scanners from the official App Store, because even if some malicious QR code scanners slip through, they are usually quickly removed.

Rule 7. **Never call phone numbers from QR codes.**

QR codes can contain phone numbers, but you should **never** call such numbers.



Criminals may trick you into calling premium phone numbers that greatly increase your phone bill. In other cases, the phone number may forward all of your future calls to the criminals.

Even if you recognize the area code of the phone number, and it seems plausible given the context, criminals might have simply substituted the legitimate number with their own.

Always look up the contact information online yourself, and never call phone numbers obtained directly from a QR code.

Rule 8. **When in doubt, do not scan the QR code.**

You can **always** ask for a paper menu, pay by other means, or look up the contact information online yourself.

Figure 5.11: QRNope v5.0.

Regarding reading time, QRNope v5.0 is estimated to take exactly 5 minutes to be read, keeping its reading-time within the maximum time available to the target population determined in Section 5.2.2.1.
Regarding the readability of QRNope v5.0, the calculations used the same methodology outlined in Section 5.2.5.2.2. QRNope v5.0 estimated FRES score was 71.5, so 3.5 points lower than the first version, but still considered "fairly easy" to read. Regarding the FKGLF measure, the estimate returned a 5.8, meaning that QRNope v5.0 would require almost a whole US education grade more than the first version to be understood. Even so, a 6th-grade reading level is still within the 7th-grade reading level upper limit determined in Section 5.2.3.3.

5.3 Conclusion

This chapter described the application of Framework 2.0 to the development of QRNope. Starting from the attack techniques and misconceptions identified in Chapter 4, each stage of the framework was applied to develop, implement, and iteratively refine the intervention through expert feedback and informal feedback from the target population.
The resulting intervention, QRNope v5.0, satisfies the design requirements established throughout the development process while preserving the intended content and awareness goals. The effectiveness of QRNope is evaluated in Chapter 6.

6 Evaluation of *QRNope*

This chapter presents the empirical evaluation of QRNope. Following the development process described in Chapter 5, the effectiveness of QRNope was evaluated through an online between-subjects study. Before conducting the final evaluation, the study protocol was pilot-tested to identify potential issues affecting the study itself.

To evaluate whether QRNope improved its readers' ability to distinguish phishing QR codes from legitimate ones, the awareness intervention was compared against both a control group, representing the absence of an awareness intervention, and an existing QRishing awareness intervention. The comparison with an existing awareness intervention was intended to determine whether the framework-guided development of QRNope resulted in an awareness intervention that outperformed awareness interventions currently available to the target population.

The comparison against an existing QRishing awareness intervention required the selection of an appropriate comparator. As the target population consisted of users residing throughout the US, the selected comparator should likewise be applicable to the entire target population. Selecting an awareness intervention published by a state-level organization or a private entity could have introduced an uneven coverage of the target population, given that such an awareness intervention may only be relevant to specific geographical regions or customer bases. For example, awareness interventions published by a state-level organization, such as the California Attorney General's Office, may only be directly applicable within a particular state; conversely, an awareness intervention published by a private organization, such as a bank or cybersecurity company, may only be relevant to its customers or primary area of operation. Consequently, it was decided to select an intervention published by a federal organization with nationwide relevance. To this end, the QRishing awareness recommendations published by the Federal Trade Commission (FTC) were selected. As the US federal agency responsible for consumer protection, the FTC provides publicly available awareness interventions aimed at helping consumers recognize and avoid fraud, while also serving as a federal reporting authority for fraud. As such, the FTC also provides an awareness intervention on QRishing, which constitutes an appropriate benchmark against which to evaluate QRNope.

The final evaluation was conducted as an online between-subject study involving 240 participants recruited through Prolific. Overall, the results showed that QRNope significantly outperformed both the FTC recommendations and the control condition while also reducing the prevalence of several QRishing misconceptions.

The chapter first discusses the related work in Section 6.1. Subsequently, the evaluation methodology is presented in Section 6.2, including the study design, pilot study, participant recruitment, and data analysis. The obtained results are then presented in Section 6.3 and discussed in Section 6.4.

Contributions described in this chapter:

- Evaluate the effectiveness of QRNope through an online between-subjects study.

6.1 Related Work

To the best of the present knowledge, no previous study has evaluated the effectiveness of a QRishing awareness intervention. As such, no directly comparable evaluation methodology exists. However, several methodological aspects of the evaluation presented in this chapter were informed by previous work on information security awareness interventions and QR code scanner evaluations. The remainder of this section discusses these works and their influence on the design of the present study.

Online between-subjects user studies constitute an established methodology for evaluating information security awareness interventions, including textual interventions (e.g., Jansen and Van Schaik (2019)), video-based interventions (e.g., Berens et al. (2024b)), as well as more complex phishing training interventions combining multiple educational components (e.g., Schöni et al. (2024)). These studies demonstrate that online between-subjects designs are suitable for evaluating awareness interventions delivered through different formats. Consequently, QRNope was evaluated through an online between-subjects user study.

A central component of the evaluation methodology employed in this chapter is the interaction with an emulated QR code scanner. This methodology builds upon the work of Mossano et al. (2026b), which proposed an online methodology for evaluating QR code scanner interfaces. To this end, the authors conducted two between-subjects online user studies. The first compared Samsung and iOS-adjacent QR code scanner interfaces against the first version of SEQR (Security Enhanced QR code scanner), while the second compared the second version of SEQR (SEQR 2.0) against the Privacy Friendly QR Code Scanner. In both studies, the participants interacted with standalone QR codes, as the objective was to evaluate the usability of the QR code scanner interfaces themselves rather than the context in which QR codes are encountered. The present work adopts the same interaction methodology while adapting it to the goal of the present study, i.e., to evaluate the effectiveness of QRNope. QR codes are therefore presented as part of realistic pretexts before the participants interact with a high-fidelity emulation of the iOS Camera application running on an Apple iPhone SE 2022 with iOS 26.2. Furthermore, the flow of the interaction was refined by reproducing additional interface elements, including the Home button, allowing participants to terminate interactions as they would on a real device. These

modifications build upon the methodology proposed by Mossano et al. (2026b) while adapting it to the evaluation of an awareness intervention. The resulting interaction provides a controlled approximation of everyday QR code interactions, while allowing the recruitment of a substantially larger participant sample than would typically be feasible in a laboratory or field study.
Besides evaluating whether participants correctly distinguish phishing QR codes from legitimate ones, the user study also evaluates whether QRNope successfully corrects the misconceptions addressed by the intervention (described in Section 5.2.4). The methodology adopted for this purpose follows the approach proposed by Mayer and Volkamer (2018). The authors first identified misconceptions about passwords through a systematic literature review and subsequently developed an awareness intervention addressing them. The effectiveness of the intervention was then evaluated not only through participants' behavior, but also through dedicated misconception statements designed to determine whether the targeted misconceptions had been corrected. The present user study likewise employs dedicated statements to assess whether the QRishing misconceptions addressed by QRNope had been successfully corrected. Furthermore, the methodology was extended to evaluate the mitigation strategies for two attack techniques, namely *Malicious QR Code Generator* and *Malicious QR Code Scanner*, which could not be realistically evaluated through the interaction scenarios employed in the study.

6.2 Methodology

This section starts by presenting the research questions in Section 6.2.1, followed by the study groups (Section 6.2.2) and the study design (Section 6.2.3). The pretexts employed throughout the evaluation are then described in Section 6.2.4, followed by the recruitment strategy and ethical considerations (Section 6.2.5) and the pilot studies (Section 6.2.6). Subsequently, the data analysis process is described in Section 6.2.7, and the section concludes with the limitations of the evaluation in Section 6.2.8.

6.2.1 Research Questions

The research in this chapter aimed at answering the following four research questions:

- **RQ1.** Are readers of QRNope more effective than readers provided only with general information on QR codes in distinguishing phishing QR codes from legitimate ones?
- **RQ2.** Are readers of QRNope more effective than readers of an existing QRishing awareness intervention in distinguishing phishing QR codes from legitimate ones?
- **RQ3.** Are readers of QRNope more effective than readers provided only with general information on QR codes at correcting the misconceptions on QRishing identified in the literature?

- **RQ4.** Are readers of QRNope more effective than readers of an existing QRishing awareness intervention at correcting the misconceptions on QRishing identified in the literature?

To answer the first research question, the participants were presented with a *placebo* text, i.e., a text providing general information on QR codes without any security-relevant information (shown in Figure 6.1). Conversely, QRNope was specifically developed to provide such guidance. Therefore, the following hypothesis is formulated:
H1. Readers of QRNope are more effective than readers of the placebo text in distinguishing phishing QR codes from legitimate ones.

What are QR codes?

QR codes are square, black-and-white patterns that can store information. The name “QR” stands for “Quick Response,” because these codes are designed to be read quickly by a phone or another device. A QR code is a type of barcode, but it can hold more information than the line-shaped barcodes often seen on products in stores.

A QR code is made of many small squares arranged in a grid. When you scan it, your phone reads the pattern and turns it into useful information. This might be a *URL*, which is a web address, a short piece of text, a phone number, or a *vCard*, which is a digital contact card.



Example of a QR code

QR codes were invented in 1994 by the Japanese company Denso Wave. At first, they were used in the car industry to track parts during manufacturing. Later, they became common in many other places because they are quick to scan and easy to use.

Today, QR codes are used in many everyday situations. A restaurant may use a QR code so customers can open the menu on their phones. A poster may include one that leads to

a website with more information. Tickets for concerts, trains, or events may also use QR codes so they can be checked at the entrance.



Example of a QR code on a can of Pepsi.

QR codes can also help people share information quickly. For example, one QR code can help you save someone's contact details to your phone. Another can help you connect to a Wi-Fi network without typing the password by hand.

One useful feature of QR codes is that they can sometimes still work even if a small part of the code is damaged, dirty, or covered. This does not mean every damaged QR code will work, but many can still be scanned if enough of the pattern is readable.



Example of a damaged QR code that is still readable.

Scanning a QR code is usually simple. On most smartphones, you can open the camera app and point it at the code. You do not need to take a picture. After a moment, a message or link should appear on the screen. If you tap it, your phone will open the content connected to the QR code.

In short, QR codes are a fast and convenient way to store and share information. They are common because they are easy to scan and can be used in many different situations.

Figure 6.1: Placebo text used in the QRNope evaluation.

Regarding the second research question, the existing QRishing awareness intervention selected for comparison is the one published by the FTC, following the rationale discussed in the chapter introduction. To formulate a hypothesis, it is therefore necessary to determine which attack techniques, mitigation strategies, and misconceptions are covered by the FTC intervention, and whether they are as comprehensive as those covered by QRNope. The intervention is shown in Figure 6.2 and accessible at FTC (2023). Highlighted are the attack techniques and misconceptions covered.

Scammers hide harmful links in QR codes to steal your information

QR codes seem to be everywhere. You may have scanned one to see the menu at a restaurant or pay for public parking. And you may have used one on your phone to get into a concert or sporting event, or to board a flight. There are countless other ways to use them, which explains their popularity. Unfortunately, scammers hide harmful links in QR codes to steal personal information. Here's what to know.

There are reports of scammers covering up QR codes on parking meters with a QR code of their own. And some crafty scammers might send you a QR code by text message or email and make up a reason for you to scan it. These are some of the ways they try to con you:

- they lie and say they couldn't deliver your package and you need to contact them to reschedule
- they pretend like there's a problem with your account and you need to confirm your information

- they lie, saying they noticed suspicious activity on your account, and you need to change your password

These are all lies they tell you to create a sense of urgency. They want you to scan the QR code and open the URL without thinking about it.

Misc. 7

A scammer's QR code could take you to a spoofed site that looks real but isn't. And if you log in to the spoofed site, the scammers could steal any information you enter. Or the QR code could install malware that steals your information before you realize it.

So how can you protect yourself?

Misc. B

- **If you see a QR code in an unexpected place, inspect the URL before you open it.** If it looks like a URL you recognize, make sure it's not spoofed—look for misspellings or a switched letter.

Mangle

- **Don't scan a QR code in an email or text message you weren't expecting—especially if it urges you to act immediately.** If you think the message is legitimate, use a phone number or website you know is real to contact the company.

Malic. phone number

- **Protect your phone and accounts.** Update your phone's OS to protect against hackers and protect your online accounts with strong passwords and multi-factor authentication.

Misc. 1



Figure 6.2: Awareness intervention from the FTC used in the QRNope evaluation.

As shown in Figure 6.2, only a relatively small number of attack techniques are covered. As such, it seems plausible that readers of QRNope will be able to detect and mitigate a higher number of QRishing attempts than the readers of the intervention from the FTC. Hence, a hypothesis is formulated with respect to RQ2:

H2. The readers of QRNope are more effective than the readers of the intervention from the FTC in distinguishing phishing QR codes from legitimate ones.

Regarding the third research question, just as for H2, the placebo text does not cover misconceptions. As such, the following hypothesis is formulated:

H3. Readers of QRNope are more effective than readers of the placebo text at correcting the misconceptions on QRishing identified in the literature.

Regarding the fourth research question, the intervention from the FTC addresses only one of the misconceptions identified in the literature, i.e., Misconception 1 "Strong passwords are enough to protect from QRishing". Furthermore, as highlighted in Figure 6.2, Misc. 1 "Strong passwords are enough to protect from QRishing" is reinforced rather than corrected, as the FTC intervention clearly states that strong passwords would help against QRishing attacks. As such, the following hypothesis is formulated:

H4. Readers of QRNope are more effective than readers of the intervention from the FTC at correcting the misconceptions on QRishing identified in the literature.

6.2.2 Study Groups

To answer the research questions, the participants were randomly assigned to one of three study groups:

- **Control.** Participants in this group read a placebo text conveying general information about QR codes (e.g., legitimate uses of QR codes), without mentioning that QR codes can be used for phishing, how to detect such attacks, or how to avoid becoming a victim.
- **FTC.** Participants in this group read the awareness intervention developed by the U.S. Federal Trade Commission.
- **QRNope.** Participants in this group read the QRNope awareness intervention.

6.2.3 Study Design

This section describes the study design followed to conduct the final evaluation of QRNope. The following paragraphs describe each step of the study design. Figure 6.3 shows the study design in its entirety.

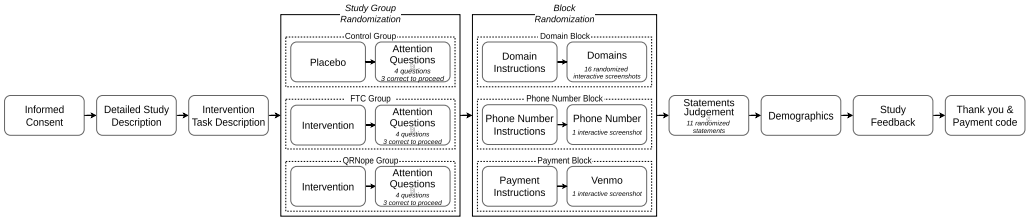


Figure 6.3: Study design followed to evaluate QRNope.

Informed consent. At the beginning of the study, the participants were informed of the purpose of the study. They were also informed that all data would be collected anonymously, that no personally identifiable information was recorded, and that they could withdraw from the study at any time by simply closing the browser window. This last action would have led to the deletion of the participants' data.

Detailed study description. At this step, the participants were informed of the specific tasks they would have been asked to perform, how much time the whole study would have taken, and that there would have been attention questions. They were also informed that answering the attention questions incorrectly would have caused the study to end before they could obtain the code to receive their payment from Prolific.

Intervention task description. At this step, the participants were reminded that they would be given an intervention to read, and that they would have been asked a set of attention questions immediately afterwards. They were also reminded of what would have happened if they answered the attention questions incorrectly.

Study group randomization. When participants reached this point, they were randomly assigned to one of the study groups. Regardless of whether they saw the placebo text or one of the interventions, they were then presented with a page containing four attention questions that could be answered if they had carefully read what they were shown. Three out of the four questions needed to be answered correctly to proceed.

Block randomization. The study comprised three blocks: *Domain-Block*, *Phone Number-Block*, and *Payment-Block*. Each block began with a set of instructions explaining how to interact with the corresponding study environment, and was then followed by the pretexts and QR codes to judge. The Domain-Block consisted of 16 pretexts, while the Phone Number-Block and the Payment-Block each consisted of a single pretext. The order of the three blocks was randomized, and the 16 pretexts within the Domain-Block were also presented in random order. Each of the pretexts is described in detail in Section 6.2.4.

Statements Judgment. In this step, the participants were shown 11 statements and asked whether they were correct or not. Nine of the statements represented the misconceptions addressed by QRNope, while the remaining two addressed the Malicious QR Code Generator attack technique and the Malicious QR Code Scanner attack technique. Unlike the other attack techniques considered in the study, these two could not be realistically evaluated through QR code pretexts. The Malicious QR Code Generator attack technique would have required an interaction with a malicious QR code generator and the detection of the substituted QR code content. Similarly, the Malicious QR Code Scanner attack technique would have required knowledge of the original QR code content and the recognition that it was modified by the scanner. Furthermore, the recommended mitigation strategy for both is to only download such applications from the official App Store. As such, evaluating this mitigation strategy was deemed a fair compromise. Six of the 11 statements (five of the misconceptions and the statement on the Malicious QR Code Scanner) were inverted and therefore needed to be evaluated as incorrect. This methodology is the same as the one used in Mayer and Volkamer (2018) for password misconceptions.

Demographics. The participants were asked about their age, gender, level of IT knowledge, whether they had previously received phishing awareness training, from where, and how often they scan QR codes.

Study feedback. At this step, the participants were asked whether they had any feedback on the study itself, for example if they noticed any issues. Furthermore, they were asked whether they had completed the study attentively, specifying that their answer would not have affected their remuneration.

Thank you & Payment code. Finally, the participants were thanked for their time, provided with the code needed to claim their remuneration, and directed to a SECUSO webpage containing awareness material on phishing[1].

6.2.4 Pretexts Used in the Study

The evaluation was designed to cover all attack techniques addressed by QRNope. Rule 2 groups URL-based attack techniques into four broad categories: phishing domains employing random or suspicious names, subdomains and different top-level domains, mangled domains (i.e., typographical variations of legitimate domains), and domains employing reassuring or contextually fitting words. To evaluate the participants' ability to identify each category, two independent domain examples were developed for each one, resulting in eight domain-based scenarios. The domains employed in the evaluation were partially reused from Mossano et al. (2026b) and partially created specifically for the present study. In addition to the domain-based scenarios, one scenario employing a phone number and one employing a payment request were included, as the QRNope version used in the evaluation explicitly recommends rejecting both types of QR codes (under Rule 4 and Rule 7). Consequently, the evaluation comprised ten distinct scenarios, each one tied to a different organization.

Six of the ten organizations, i.e., Facebook, BBC, CNN, PayPal, Google, and Bank of America, were selected because they had already been employed in previous phishing and domain-based user studies conducted in the US (e.g., Mossano et al. (2026b), Reynolds et al. (2020), Albakry et al. (2020)). However, four additional organizations were required to cover the remaining scenarios. Amazon, Pizza Hut, Netflix, and Venmo were selected due to being widely recognized among users in the United States: Amazon accounts for approximately 40% of the U.S. e-commerce market and around 82% of U.S. households subscribe to Amazon Prime (as shown in Market.us (2025)). Netflix has approximately 67 million subscribers in the United States (as reported by Statista (2025)), Venmo reports more than 90 million active accounts (as shown in PayPal (2025)), and Pizza Hut operates more than 6,000 restaurants across the country (as mentioned in Pizza Hut (2026)).

All pretexts except the payment one (discussed later in this paragraph) were adapted from real-world examples found through searches using DuckDuckGo Images or from personal emails

[1] https://secuso.aifb.kit.edu/english/642.php

received by the author of the thesis. The original material was only modified where necessary to fit the study, i.e., through the addition of a QR code. Consequently, the type of pretext associated with each organization depended on the real-world material that could be found. For example, if a poster advertising a service from Bank of America was identified, that poster became the basis for the corresponding study pretext. The only exception was the Venmo payment pretext, which was implemented as a fictional taco truck to represent a realistic small-business payment setting. A fictional business was preferred to avoid associating the payment scenario with an existing merchant while still providing participants with a familiar purchasing context. Table 6.1 summarizes all the pretexts used in the Domain-block, along with the associated organization and the corresponding domains.

Table 6.1: List of the organizations used in the evaluation, their type, the pretext type, and the corresponding domains and subdomains.

Organization	Type	Pretext type	Domain & subdomain shown
Bank of America	Legitimate	Poster	bankofamerica.com
Pizza Hut	Legitimate	Poster	pizzahut.com
CNN	Legitimate	Poster	info.cnn.com
Google	Legitimate	Email	google.com
BBC	Legitimate	Flyer	shop.bbc.com
PayPal	Legitimate	Flyer	paypal.com
Facebook	Legitimate	Flyer	facebook.com
Netflix	Legitimate	Email	netflix.com
Bank of America	Random	Poster	hostaddress58.com
Pizza Hut	IP address	Poster	87.147.12.350
CNN	Subdomain	Poster	cnn.com.info
Google	Different TLD	Email	google.tech
BBC	Mangle	Flyer	shop.bbo.com
PayPal	Mangle	Flyer	paypla.com
Facebook	Fitting word	Flyer	profile.com
Netflix	Fitting word	Email	stream-netflix.com

The domains used in the study were verified before the evaluation began by visiting each one in a browser. They were either unused, controlled by the corresponding organization, or controlled by the SECUSO group. However, to eliminate any possibility of accidental interaction with phishing domains, none of the QR codes shown in the study encoded these resources. Instead, every QR code encoded the message “This is an example QR code”. The phone number used in the study consisted of a randomly generated sequence of digits preceded by the area code of

Union County, US (i.e., “908”). This area code was selected because the corresponding Amazon employment poster used as the study pretext was originally published by the Union County American Job Center. At the time of the evaluation, calling this number returned an “Nonexistent phone number” message.

Each scenario began with a short instruction text reminding the participants of the task they were expected to perform. The corresponding pretext containing the QR code was displayed immediately below the instruction text, as shown in Figure 6.4. For the domain-based scenarios, the same pretext was used for both the phishing and the legitimate versions, with the only difference being the domain displayed after scanning the QR code. This ensured that the phishing and legitimate versions differed only in the scanned domain while all other aspects of the pretext remained identical.

Below the pretext, the participants were presented with an emulation of an Apple iPhone SE 2022 reproducing the behavior of the iOS Camera application from iOS 26.2. The participants were instructed to press the *Scan now* button to begin the interaction. Once the QR code had been scanned, they could interact with the emulated device as they would with the original iOS Camera application. The interaction available to the participants depended on the type of QR code that had been scanned.

For Domain-based scenarios, scanning the QR code displayed the corresponding domain. Participants could either open the domain or dismiss it. The interaction could be terminated either by pressing the *X* button displayed next to the domain or by pressing the Home button.

For phone number scenarios, selecting the displayed phone number opened a second screen asking the participant whether they wished to place the call or cancel it. The interaction could be terminated both from the initial screen and from the confirmation screen by pressing the Home button.

For payment scenarios, scanning the QR code directly opened a reproduction of the Venmo application. This implementation was chosen because the iOS Camera application itself only indicates that the Venmo application will be opened, without providing any information allowing users to determine whether the QR code is legitimate. Within the simulated Venmo application, participants could either proceed with the payment or terminate the interaction using either the *X* button or the Home button.

All interactions available to the participants were explained before the corresponding study block through dedicated instruction pages. The instructions are shown in Appendix A.4. The pretexts and what the scanning showed to the participants are available at https://doi.org/10.5281/zenodo.20927603.

Study

Your task is to classify the examples as either:

- **Legitimate**, i.e., leading you to a website address that you expect to visit, calling a number that you expect to call, or paying the correct organization.
- **Phishing**, i.e., *NOT* leading you to a website address that you expect to visit, *NOT* calling a number that you expect to call, or *NOT* paying the correct organization.

based on the information you read in the awareness intervention.

To do this, you can interact with the simulated phone at the end of the page as you would normally do with the iOS QR code scanner in the Camera App.

You have the option of opening the website address, completing the payment, or calling the phone number if you determine that it is legitimate.

You also have the option to cancel the opening of the website address, payment, or phone call if you think the QR code is a phishing one.

Please note: Imagine that you are scanning a QR code from the image below:





Figure 6.4: Example of the interaction in the Domain-block part of the study.

6.2.5 Recruitment and Ethical Considerations

The study was approved by the Karlsruhe Institute of Technology (KIT) Ethics Commission under the umbrella ethical approval covering online phishing awareness intervention evaluations.
The required sample size was determined through an *a priori* power analysis conducted in R, the corresponding script being available at https://doi.org/10.5281/zenodo.20927603. The analysis assumed the use of a one-way ANOVA omnibus test with a significance level of $\alpha = .05$ and a statistical power of $1-\beta = .90$. Because significant omnibus results were expected to be followed by pairwise post-hoc comparisons, a Bonferroni correction was applied to the significance level when determining the required sample size. The resulting analysis indicated a required sample of 87 participants per study group, corresponding to a total of 261 participants.
Both the pilot studies and the main evaluation were conducted using the Prolific recruitment platform. The eligibility criteria were enforced directly through Prolific's prescreening system, meaning that only participants satisfying all inclusion criteria were shown the study and could choose to participate. Specifically, participants were required to be at least 18 years of age, reside in the United States, speak English, own and regularly use an iOS device, and not have previously participated in QRishing-related studies conducted by SECUSO. The restrictions regarding the country of residence, the language, and the operating system reflect the target population for which QRNope was developed, as discussed in Section 5.2.4.1. The latter criterion also applied to the pilot studies, ensuring that participants who took part in one pilot could not participate in subsequent pilots or in the main evaluation, thereby avoiding potential learning effects.
The sample was not selected to be representative of the US population. On Prolific, recruiting a representative sample requires at least 300 participants, exceeding the sample size determined by the *a priori* power analysis. Rather than increasing the sample size beyond what was statistically required, a stratified sampling approach was adopted. Specifically, equal quotas were defined for each age group (18–24, 25–34, 35–44, 45–54, 55–64, and 65–100 years), and participants within each age group were balanced on gender.
Based on the expected duration of the study, the completion time was estimated at 16 minutes, a value that was subsequently confirmed during the pilot studies. Participants were remunerated at a rate of £12.00 per hour, corresponding to £3.20 for completing the study.
Recruitment continued until 261 valid participants had been obtained for the main evaluation. An internal counter implemented in SoSci Survey was used to keep track of the number of valid participants assigned to each study group, allowing the groups to remain approximately balanced throughout the recruitment process. Exact balancing between groups could not be guaranteed due to the implementation constraints of SoSci Survey. The study was implemented in SoSci Survey because the ethics approval required the use of an approved survey platform and because SoSci Survey complies with the General Data Protection Regulation (GDPR). Although the study recruited participants residing in the United States, it was conducted by a European institution, and

the data collection was therefore implemented according to European data protection standards. Participants who did not provide informed consent, failed the attention questions, or exceeded the maximum completion time of 38 minutes established by the Prolific platform for the study were excluded from participating and automatically replaced through the recruitment process until the required number of 240 valid participants had been obtained. Participants who attempted to access the study a second time were automatically prevented from doing so, as their Prolific ID was blocked after their first access.

6.2.6 Pilot Studies

Before conducting the main evaluation, two pilot studies were conducted to verify that the study implementation in SoSci Survey functioned as intended. In particular, the pilot studies were used to assess the clarity of the study instructions, the correctness of the study flow, the functioning of the interaction with the emulated QR code scanner, the randomization procedure, the completion time, and the questionnaires employed throughout the study. Participants were also invited to report any technical issues or ambiguities they encountered while completing the study.
The first pilot study involved twelve participants, with four participants randomly assigned to each study group. Two issues were highlighted. First, due to an implementation error, the validation of the attention questions in the QRNope condition checked the wrong variables, preventing participants from proceeding. Secondly, several participants reported that the font size used throughout the study was too small. The validation procedure was therefore corrected and the font size was increased throughout the study.
The user study was subsequently validated in a second pilot study involving twelve participants again. The attention question validation functioned correctly, and no further issues were reported by the participants. Consequently, the study protocol was considered ready for the main evaluation.

6.2.7 Data Analysis Process

The data cleaning process followed three steps. First, participants who did not answer the feedback question in the *Study Feedback* step were excluded, as they had not completed the study in its entirety. This resulted in the exclusion of three participants, reducing the sample from 261 to 258 participants.
Second, participants who explicitly stated in the *Study Feedback* step that they had not completed the study mindfully were excluded from the analysis. This resulted in the exclusion of a further three participants, leaving a sample of 255 participants.
Third, participants exhibiting clear response patterns indicative of indiscriminate answering were removed. Specifically, participants were excluded if they consistently selected the same response throughout each block by either opening every domain, calling the phone number, and completing

the payment, or rejecting every domain, not calling the phone number, and not completing the payment. This resulted in the exclusion of 15 additional participants, yielding a final sample of 240 participants (Control: $n = 84$, FTC: $n = 76$, QRNope: $n = 80$).

To determine whether the exclusions substantially affected the planned statistical power, the original power analysis presented in Section 6.2.5 was repeated using the final sample size. Using the smallest study group (i.e., FTC, $n = 76$), while maintaining the same assumptions adopted for the original power analysis (medium effect size, Cohen's $f = 0.25$, and Bonferroni-adjusted significance level $\alpha = .05/3$), the updated analysis yielded a statistical power of $1 - \beta = .85$. For the QRNope group ($n = 80$), the power analysis yielded a statistical power of $1 - \beta = .87$.

As the intended statistical analysis was an omnibus ANOVA followed by a Tukey post hoc test, the parametric assumptions were assessed through statistical tests. The Shapiro–Wilk test indicated that the residuals of the overall model deviated significantly from a normal distribution ($W = .970$, $p < .001$), whereas Levene's test indicated that the assumption of homogeneity of variances was satisfied ($F(2, 237) = 1.063$, $p = .347$). According to Field (2013), ANOVA is relatively robust against moderate departures from normality, particularly when the group sizes are large and approximately balanced (as is the case for the present study). Consequently, to answer RQ1 and RQ2, an ANOVA test followed by a pairwise Tukey's Honest Significant Difference (HSD) post hoc test were used for the inferential analyses. The analysis was then expanded by evaluating the influence of both ordering factors (i.e., the presentation order of the study blocks and the presentation order of the URLs within the Domain-block) and the demographic factors of the sample (i.e., gender, age-group, IT knowledge, QR code scanning frequency, previous phishing knowledge, previous QRishing knowledge, and phishing training frequency). Additionally, the two statements evaluating the mitigation strategies for the *Malicious QR Code Generator* attack technique and the *Malicious QR Code Scanner* attack technique were analyzed separately. As these statements were included solely to provide a descriptive evaluation of the participants' understanding of the corresponding mitigation strategies, only the proportion of correct responses was computed for each study group.

Regarding the variables considered in the inferential analyses, three outcome variables were computed for the statistical analyses. The overall score corresponds to the total number of correct judgments across all 18 pretexts presented to the participants. Additionally, two complementary scores were computed to separately evaluate participants' performance in correctly rejecting pretexts and correctly accepting them, i.e., when to stop the interaction and when to proceed with it. This terminology and scoring approach was chosen because, while the domain pretexts can be classified as either legitimate or phishing, this is not the case for the phone number and payment pretexts. As discussed in Chapter 5, when a phone number is encoded in a QR code, QRNope recommends stopping the interaction and verifying the contact through an independent source. For payment requests, the recommended mitigation strategy is to reject the interaction when no impersonation-resistant identity indicators are available. Since the phone number in the study was

not externally verified and the payment request was made through the Venmo application, which does not provide impersonation-resistant identity indicators, both the phone number pretext and the payment pretext should have been rejected. For this reason, they are treated as *neutral pretexts*, i.e., pretexts for which the correct action in the study was independent of their actual legitimacy. Consequently, these two neutral pretexts contribute equally to both the rejection score and the correct acceptance score. Accordingly, the rejection score comprises the eight phishing domain pretexts together with the two neutral pretexts, while the correct acceptance score comprises the eight legitimate domain pretexts together with the same two neutral pretexts, resulting in 10 items per score. Therefore, whenever percentages are reported, they were computed using the corresponding denominator, i.e., 18 items for the overall score and 10 items for both the rejection score and the correct acceptance score.

To answer RQ3 and RQ4, a generalized linear mixed-effects model with a binomial error distribution was fitted to the correctness of the individual misconception statement evaluations. The intervention was included as a fixed effect, whereas participant and misconception were included as random intercepts to account for the repeated observations contributed by each participant and for potential differences between the misconception statements. To directly compare QRNope with both the placebo text and the FTC intervention, the model was fitted twice using different reference groups. The results of the statistical analysis are presented in Section 6.3.

6.2.8 Limitations

The first limitation concerns the number of scenarios used during the evaluation. Each attack technique was represented by a single scenario, for which both a phishing and a legitimate version were created by modifying only the domain displayed by the simulated iOS Camera application. Similarly, the phone number and Venmo payment scenarios were each represented by a single example. Consequently, although the phishing and legitimate versions of each Domain-based scenario differed only in the displayed domain, the measured performance may nevertheless have been influenced by characteristics of the chosen scenario itself, such as the company involved, the context, or the wording of the pretext.

Including multiple independent scenarios for each attack technique, as well as multiple phone number and payment scenarios, would have reduced the influence of any individual example and allowed a more robust evaluation. However, such a design would have substantially increased the duration of the study. The evaluation already required participants to judge 18 QR code scans, comprising 16 domain-based scenarios together with one phone number and one Venmo payment scenario. Adding a second scenario for each attack technique would have almost doubled the number of decisions required from the participants, substantially increasing participant burden and likely reducing the quality of the collected data because of fatigue. Instead, a compromise was

adopted between breadth of attack technique coverage and study duration, allowing a wider range of attack techniques to be evaluated within the available participant time and financial constraints.
A second limitation concerns the balance between phishing and legitimate examples. The study deliberately employed an equal number of phishing and legitimate pretexts, whereas in real-world settings legitimate QR codes are substantially more common. This balanced design was adopted to avoid introducing response biases and to ensure sufficient statistical power to compare performance across both classes of stimuli. Similar balanced experimental designs have also been employed in previous phishing research (e.g., in Berens et al. (2024b)). Consequently, although the absolute detection rates observed in this study should not be interpreted as estimates of real-world performance, the relative comparison between the three study conditions remains valid.
A further limitation concerns the study environment. Although the evaluation aimed to closely resemble the interaction with a real iPhone, the participants interacted with a web-based replica of the iOS Camera application rather than with a physical device. The replica reproduced the behavior of the iOS Camera application (version 26.2), including the visualization of scanned QR codes and the interactions required to dismiss a QR code, place a phone call, or proceed to a payment application. To further increase the realism of the interaction, participants could either use the interface controls (e.g., the *X* button) or the Home button to dismiss the current view, reproducing the interaction available on the original device. This experimental setup is similar to the one employed by Mossano et al. (2026b), while extending it by also reproducing the Home button interactions.
Nevertheless, interacting with a web-based replica is not identical to interacting with a physical smartphone during everyday use. Consequently, the results should be interpreted as reflecting participants' decision-making in a highly realistic simulated environment rather than during natural smartphone use.
A final limitation concerns the temporal scope of the evaluation. The effectiveness of QRNope was assessed immediately after the participants had read the intervention. Consequently, the present study does not provide evidence regarding the long-term retention of the recommendations or whether their effectiveness decreases over time. A longitudinal evaluation would therefore be required to determine the persistence of the observed effects and whether periodic refresher interventions would be beneficial.

6.3 Results

This section starts by presenting the results regarding the participants' demographic characteristics in Section 6.3.1. Subsequently, the results for the attack techniques (RQ1 and RQ2) are presented in Section 6.3.2, followed by the results for the misconceptions (RQ3 and RQ4) in Section 6.3.3.

Note: Throughout this section, *performance* refers to the percentage of participants who correctly evaluated a single pretext. *Correct Acceptance* and *Correct Rejection* refer to the corresponding aggregate measures described in Section 6.2.7. *Overall performance* refers to the aggregate performance across all 18 pretexts.

6.3.1 Demographics

The final sample after the data cleaning process consisted of 240 participants, with 84 participants assigned to the Control group, 76 to the FTC group, and 80 to the QRNope group. Table 6.2 summarizes the demographic characteristics of the sample.

The gender distribution was balanced, comprising 116 female participants, 120 male participants, and four participants identifying as non-binary. This balance was also maintained within the study groups, with no notable imbalance.

Similarly, the age distribution was relatively uniform across all six predefined age-groups, with the most numerous age-groups being 35–44 and 45–54 (43 participants each), and the least numerous being the 55–64 age-group (35 participants). When considering the distribution across the study groups, the largest imbalance is in the FTC group, which had 20 participants from the 45–54 age-group, while the other age-groups ranged between eight and 15 participants.

Regarding their self-reported IT knowledge, most participants described themselves as either experienced (113 participants) or having average knowledge (91 participants), while only four participants identified themselves as laypersons. Considering the distribution across study groups, the Control group had more experienced participants than the other groups, although this was partly balanced by the FTC group and the QRNope group having more participants who majored in Computer Science.

Most participants indicated that they scanned QR codes occasionally, with the largest groups reporting scanning QR codes less than once a month (72 participants), once a week (68 participants), or once a month (66 participants).

To assess previous phishing awareness, participants were first asked whether they had previously learned about phishing through any source, including flyers, videos, or training courses. Using this broad definition, 143 participants reported prior exposure to phishing awareness material. Participants who answered positively were additionally invited to indicate where they had acquired this knowledge. The responses were categorized into thematic groups, allowing multiple categories to be assigned to a single response. The most frequently reported sources were workplace training, videos, and general online resources. Table A.1 in the Appendix contains the full list of sources.

Participants were then asked whether they had previously heard about QRishing. Overall, 117 participants reported previous knowledge of QRishing, while 123 indicated that they had not. Participants who answered positively were again asked to indicate the source of their knowledge.

Table 6.2: Demographic characteristics of the participants.

Characteristic	Category	Control	FTC	QRNope	Total
Gender	Female	45	34	37	116
	Male	39	39	42	120
	Non-binary	0	3	1	4
Age	18–24	13	15	13	41
	25–34	17	11	13	41
	35–44	16	9	18	43
	45–54	12	20	11	43
	55–64	12	8	15	35
	65+	14	13	10	37
IT knowledge	Layman	0	3	1	4
	Average	30	30	31	91
	Experienced	46	31	36	113
	Majored	8	12	12	32
QR code scanning frequency	Daily	2	0	1	3
	Multiple times a week	13	6	7	26
	Once a week	23	24	21	68
	Once a month	26	16	24	66
	Less than once a month	20	29	23	72
	Never	0	1	4	5
Previous phishing knowledge	Yes	42	52	53	143
	No	43	30	33	97
Previous QRishing knowledge	Yes	38	38	41	117
	No	46	38	39	123
Phishing training frequency	Never	51	31	45	127
	Once	12	13	12	37
	Every two years or less	6	10	7	23
	Around once a year	10	15	8	33
	About once a quarter	3	4	5	12
	More than once a quarter	2	3	3	8
Participants number		**84**	**76**	**80**	**240**

The most frequently reported sources were news media, workplace training, and general online resources. Table A.2 in the Appendix contains the full list of sources.

Finally, participants were asked how often they had received phishing training or participated in a phishing training course. This item specifically referred to formal phishing training rather than general exposure to phishing awareness. Overall, 37 participants reported having received phishing training once, while 76 reported receiving phishing training at various intervals. In

contrast, 127 participants indicated that they had never received formal phishing training. When compared with the phishing awareness question responses, these results indicate that several participants had previously learned about phishing despite never having participated in a formal phishing training course.

6.3.2 Results for the Attacks (RQ1, RQ2)

This section presents the results for RQ1 and RQ2. First, the effectiveness of QRNope is compared with the Control group and the FTC group in Section 6.3.2.1. Subsequently, the influence of additional factors, including the presentation order and participants' demographic characteristics, on the participants' performance is examined in Section 6.3.2.2.

6.3.2.1 Effectiveness of the QRNope group compared with the Control group and the FTC group

Table 6.3 summarizes the participants' overall performance across the three study groups. Participants assigned to the QRNope group achieved the highest overall performance, followed by the FTC group and the Control group.

Table 6.3: Descriptive statistics of the participants' performance across the three study groups. The table also reports the difference between the Correct Acceptance and Correct Rejection. The overall performance is computed over all 18 pretexts, whereas the Correct Acceptance and Correct Rejection percentages are each computed over 10 pretexts.

Study Group	Overall		Correct Acceptance		Correct Rejection		Difference
	Mean	SD	Mean	SD	Mean	SD	
Control	68.9%	2.50	71.9%	1.66	55.7%	2.20	16.2%
FTC	72.8%	2.41	70.9%	1.83	65.7%	1.94	5.2%
QRNope	82.2%	2.32	81.9%	1.54	77.6%	2.16	4.3%

A one-way ANOVA revealed a statistically significant effect of the intervention on the participants' overall performance with a large effect size ($F(2, 237) = 21.33$, $p < .001$, $\eta^2 = .15$). Post-hoc Tukey's HSD tests showed that participants assigned to the QRNope group achieved significantly higher overall performance than participants assigned to both the Control group ($p < .001$) and the FTC group ($p < .001$). In contrast, no statistically significant difference was observed between the Control group and the FTC group ($p = .108$).
Consequently, **both H1 and H2 are supported**. Participants who read QRNope were significantly more effective at distinguishing phishing QR codes from legitimate ones than participants who

read either the placebo text or the FTC intervention. Accordingly, both **RQ1 and RQ2 are answered positively**.

To further investigate the observed differences, a two-way ANOVA was conducted with the intervention and the type of judgment (Correct Acceptance or Correct Rejection) as independent variables. The analysis revealed statistically significant main effects of both the intervention ($F(2, 237) = 32.31$, $p < .001$, $ges = .113$) and the type of judgment ($F(1, 237) = 22.85$, $p < .001$, $ges = .049$). Furthermore, a statistically significant interaction between the two factors was observed ($F(2, 237) = 4.67$, $p = .010$, $ges = .021$). This interaction indicates that the effectiveness of the three interventions differed depending on whether participants had to correctly accept or correctly reject a presented pretext. As shown in Table 6.3, participants assigned to the Control group achieved higher Correct Acceptance than Correct Rejection, with a difference of 16.2%. This difference decreased to 5.2% in the FTC group and to 4.3% in the QRNope group. However, whereas the smaller difference observed for the FTC group was primarily attributable to an improvement in Correct Rejection, participants assigned to the QRNope group achieved higher performance for both Correct Acceptance and Correct Rejection. As an additional analysis, Figure 6.5 presents the performance for each individual pretext across the three study groups.

Overall, participants assigned to the QRNope group achieved the highest performance among the study groups for the Correct Rejection measure. The largest improvements compared with the Control group were observed for the phone number pretext (33.3% to 73.8%), the payment pretext (7.1% to 43.8%), the CNN subdomain pretext (40.5% to 71.2%), the Netflix extended-domain pretext (46.4% to 75.0%), and the Google different-TLD pretext (56.0% to 77.5%). Substantial improvements were also observed for the remaining Correct Rejection measure, indicating that QRNope consistently improved the participants' ability to identify QRishing attempts regardless of the underlying attack technique.

For the Correct Acceptance measure, the performance remained generally comparable across the three study groups. Participants assigned to the QRNope group achieved higher performance than the Control group for six of the eight legitimate Domain-block pretexts, with the largest improvements observed for the Google legitimate pretext (72.6% to 83.8%), the Facebook legitimate pretext (77.4% to 86.2%), and the CNN legitimate pretext (78.6% to 85.0%). Lower performance was observed only for the BBC legitimate pretext (91.7% to 80.0%) and, to a lesser extent, the Pizza Hut legitimate pretext (95.2% to 92.5%). Overall, the differences between the study groups were therefore primarily associated with the Correct Rejection measure, while the performance for the Correct Acceptance measure remained largely comparable across the three study groups.

Finally, Table 6.4 summarizes the participants' performance on the two statement judgment tasks evaluating the *Malicious QR Code Scanner* and *Malicious QR Code Generator* attack techniques. For both statements, participants assigned to the QRNope group achieved the highest proportion of correct responses. The difference between the study groups was particularly pronounced for the

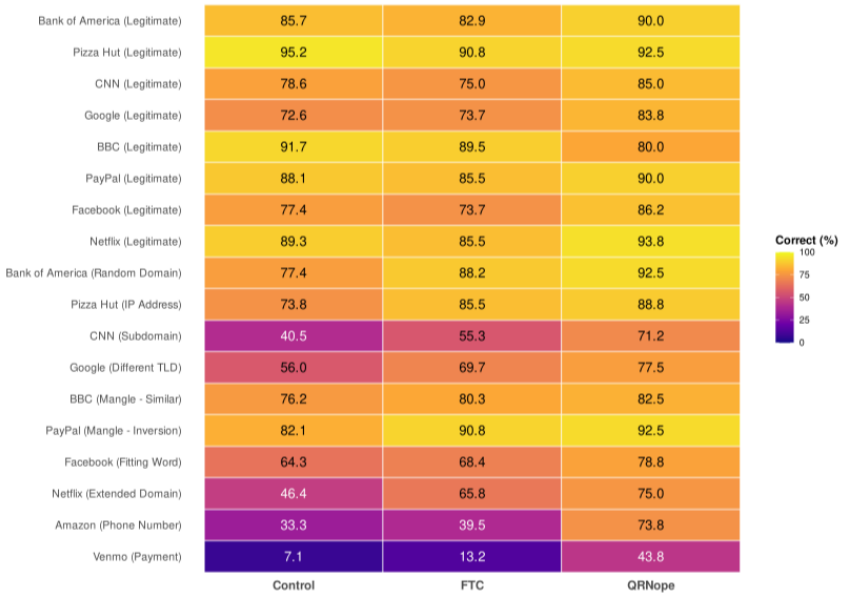


Figure 6.5: Heatmap of the participants' performance for each pretext across the three study groups. Higher percentages are represented by lighter colors.

Malicious QR Code Generator mitigation strategy, for which the QRNope group outperformed both the Control group and the FTC group by approximately ten percentage points.

Table 6.4: Percentage of participants that judge correctly the statements about the *Malicious QR Code Scanner* and *Malicious QR Code Generator* attack techniques.

Attack Technique	**Control**	**FTC**	**QRNope**
Malicious QR Code Scanner	89.3%	94.7%	95.0%
Malicious QR Code Generator	73.8%	67.1%	83.8%

6.3.2.2 Factors Influencing the Effectiveness

The first factors considered were the influence of the presentation order of the study blocks (Domain-block, Phone Number-block, Payment-block), followed by the influence of the presentation order of the pretexts within the Domain-block.

Regarding the influence of the presentation order of the study blocks, a small but statistically significant effect was observed for the position of the Domain-block ($F(1, 238) = 4.42, p = .037$). Specifically, participants achieved, on average, a 2.3% higher overall performance when the Domain-block was presented second rather than first, and a 4.7% higher overall performance when it was presented third rather than first. Similarly, the position of the Phone Number-block also showed a statistically significant effect ($F(1, 238) = 4.55, p = .034$). Participants achieved, on average, a 2.5% lower overall performance when the Phone Number-block was presented second rather than first, and a 4.9% lower overall performance when it was presented third rather than first. In contrast, no statistically significant effect was observed for the position of the Payment-block ($F(1, 238) = 0.003, p = .957$). However, it is important to note that, although the observed differences could reach approximately a 4.7% difference, the presentation order of the study blocks explained only a small proportion of the variance in the participants' overall performance ($R^2 \approx .02$).

Regarding the influence of the presentation order within the Domain-block, this was investigated through a generalized linear mixed-effects model. The model revealed a statistically significant effect of the pretext presentation order ($\beta = 0.035$, $z = 2.41$, $p = .016$), indicating an improvement in the participants' performance as they progressed through the Domain-block. The model also showed a significant effect of QR code type (i.e., legitimate or phishing; $\beta = -1.68$, $z = -4.83$, $p < .001$) and a significant interaction between the pretext presentation order and the QR code type ($\beta = 0.118$, $z = 5.82$, $p < .001$). This indicates that the improvement in performance across the Domain-block was primarily attributable to phishing domains, whereas the correct identification of legitimate domains remained largely unchanged.

Then, the analysis considered the influence of the participants' demographic factors, i.e., gender, age-group, IT knowledge, QR code scanning frequency, previous phishing knowledge, previous QRishing knowledge, and phishing training frequency. The descriptive statistics organized by the participants' demographic factors are summarized in Table 6.5.

Age-groups had a statistically significant effect on the participants' overall performance, with a medium effect size ($F(5, 234) = 3.10$, $p = .010$, $ges = .062$). Post-hoc Tukey's HSD tests showed that participants in the 65+ age-group achieved significantly lower overall performance than participants in the 25–34 age-group ($p = .015$) and the 45–54 age-group ($p = .007$), whereas no other pairwise differences achieved statistical significance. Participants between 18 and 64 years of age achieved comparable overall performance, indicating that the observed effect was primarily driven by the oldest age-group. These findings are consistent with previous research in

Table 6.5: Descriptive statistics of the participants' overall performance grouped by the participants' demographic data.

Factor	Level	N	Mean	SD
Gender	Female	116	72.8%	2.54
	Male	120	76.1%	2.68
	Non-binary	4	77.8%	1.83
Age-group	18–24	41	75.0%	2.31
	25–34	41	77.2%	2.42
	35–44	43	75.6%	2.52
	45–54	43	77.8%	2.33
	55–64	35	73.9%	3.06
	65+	37	66.7%	2.71
IT knowledge	Layman	4	75.0%	2.38
	Average	91	70.6%	2.47
	Experienced	113	76.1%	2.58
	Majored	32	78.3%	2.81
QR scan frequency	Daily	3	61.1%	4.36
	Multiple times/week	26	70.0%	2.39
	Around once/week	68	73.3%	2.70
	Around once/month	66	75.6%	2.61
	Less than once/month	72	76.7%	2.47
	Never	5	80.0%	2.51
Previous phishing knowledge	Yes	143	76.1%	2.60
	No	97	71.7%	2.56
Previous QRishing knowledge	Yes	117	76.1%	2.64
	No	123	72.8%	2.55
Training frequency	Never	127	74.4%	2.54
	Once	37	76.7%	2.75
	Every two years or less	23	73.9%	2.39
	Around once a year	33	74.4%	2.83
	About once a quarter	12	72.8%	2.68
	More frequently than once a quarter	8	66.7%	2.83

the email phishing context, which reported a lower phishing detection performance among older adults (e.g., as reported in both Grilli et al. (2020) and Lin et al. (2019)).
IT knowledge also showed a statistically significant effect on the participants' overall performance, although the observed effect size was small ($F(3, 236) = 3.45$, $p = .017$, $ges = .042$). Post-hoc Tukey's HSD tests showed that participants reporting average IT knowledge achieved significantly lower overall performance than both participants reporting experienced IT knowledge ($p = .034$)

and participants who had majored in Computer Science ($p = .047$). No other pairwise differences achieved statistical significance.

Previous phishing knowledge was associated with significantly higher overall performance, although the observed effect size was small ($F(1, 238) = 5.39, p = .021, ges = .022$). Participants who reported previous phishing knowledge achieved a mean overall performance of 76.1%, compared with 71.7% for participants without previous phishing knowledge.

No statistically significant effects were observed for gender ($F(2, 237) = 1.57$, $p = .210$, $ges = .013$), QR code scanning frequency ($F(5, 234) = 1.75$, $p = .125$, $ges = .036$), previous QRishing knowledge ($F(1, 238) = 3.43$, $p = .065$, $ges = .014$), or phishing training frequency ($F(5, 234) = 0.71$, $p = .620$, $ges = .015$).

Overall, these results suggest that age-group, IT knowledge, and previous phishing knowledge were associated with differences in the participants' overall performance. However, only age-group showed a medium effect size, whereas the effect sizes of IT knowledge and previous phishing knowledge were small. In contrast, gender, QR code scanning frequency, previous QRishing knowledge, and phishing training frequency did not show a statistically significant effect on the participants' overall performance.

6.3.3 Results for the Misconceptions (RQ3, RQ4)

To evaluate whether QRNope effectively addressed the misconceptions identified in Chapter 5, the responses to the nine misconception statements were analyzed. Each statement was designed to assess whether the corresponding misconception identified during the design of QRNope had been corrected by the intervention (see Section 5.2.4.3 and Section 5.2.4.4). For each misconception, the proportion of participants providing the correct response was computed separately for the Control, FTC, and QRNope groups. The descriptive results are summarized in Table 6.6.

As shown in Table 6.6, QRNope achieved the highest proportion of correct responses for five of the nine misconception statements (Misconceptions 6, 7, 10, 13, and 14). For two misconceptions (A and B), the three study groups achieved comparable performance, whereas for Misconceptions 1 and 15, the Control or FTC groups achieved descriptively higher performance than the QRNope group.

To determine whether the interventions influenced the participants' overall ability to correctly evaluate the misconception statements, a generalized linear mixed-effects model was fitted with intervention as a fixed effect and participant and misconception as random intercepts. Compared with the Control group, the model shows that the participants assigned to the QRNope group were more likely to correctly evaluate the misconception statements. However, the observed difference did not achieve statistical significance ($\beta = 0.410, OR = 1.51, 95\% CI = [0.96, 2.35]$, $p = .072$). Likewise, no statistically significant difference was observed between the FTC and Control groups ($\beta = 0.161, OR = 1.18, 95\% CI = [0.76, 1.82], p = .472$).

Table 6.6: Percentage of participants correctly evaluating each misconception statement.

Misconception	Control	FTC	QRNope	△ QRNope–Control	△ QRNope–FTC
Misconceptions on Attack Techniques					
A	95.2%	97.4%	96.2%	+1.0%	-1.1%
B	96.4%	93.4%	96.2%	-0.2%	+2.8%
6	81.0%	85.5%	87.5%	+6.5%	+2.0%
7	94.0%	92.1%	98.8%	+4.8%	+6.7%
Misconceptions on Mitigation Strategies					
1	81.0%	92.1%	81.2%	+0.2%	-10.9%
10	97.6%	90.8%	98.8%	+1.2%	+8.0%
13	61.9%	68.4%	73.8%	+11.9%	+5.4%
14	86.9%	89.5%	92.5%	+5.6%	+3.0%
15	91.7%	88.2%	88.8%	-2.9%	+0.6%

To directly compare QRNope with the FTC intervention, a second generalized linear mixed-effects model was fitted using the FTC group as the reference category. Compared with the FTC group, the model shows that the participants assigned to the QRNope group were more likely to correctly evaluate the misconception statements. However, this observed difference also did not achieve statistical significance ($\beta = 0.249$, $OR = 1.28$, $95\%\ CI = [0.81, 2.03]$, $p = .291$).
Consequently, **neither H3 nor H4 was supported**. Accordingly, **neither RQ3 nor RQ4 can be answered positively**.

6.4 Discussion

The evaluation demonstrates that QRNope is effective at improving users' ability to distinguish between legitimate and phishing QR codes. Participants assigned to the QRNope group achieved an overall performance of 82.2%, compared with 72.8% for the FTC intervention and 68.9% for the Control group, while also obtaining significantly higher overall performance than both comparison groups. More importantly, the observed improvement was not limited to a single attack technique. As shown in Figure 6.5, participants exposed to QRNope achieved the highest Correct Rejection rate among the three groups. These findings suggest that the development process described in Chapter 5 successfully produced an awareness intervention capable of addressing a broad range of QRishing attack techniques rather than improving performance only for a limited set of examples.
Although improvements were observed across all evaluated phishing pretexts, the payment scenario deserves particular attention. Whereas participants assigned to the QRNope group achieved Correct Rejection rates above 70% for the phone-number pretext (73.8%), the CNN subdomain

pretext (71.2%), the Google different-TLD pretext (77.5%), and the Netflix extended-domain pretext (75.0%), the corresponding Correct Rejection rate for the Venmo payment scenario reached only 43.8%. This value still represents a considerable improvement compared with both the Control group (7.1%) and the FTC intervention (13.2%), but it also indicates that payment requests remained the most difficult phishing scenario evaluated during the study.

Consequently, although Rule 4 successfully improved participants' performance, the evaluation suggests that further work is required before payment-related QRishing attack techniques can be considered to be addressed to the same extent as the remaining attack techniques. One possible explanation for the QRNope participants' performance on the Payment pretext is that payment requests fundamentally differ from the remaining scenarios considered throughout the evaluation. For the Domain-based pretexts, participants primarily needed to evaluate whether the domain presented by the iOS Camera emulation was legitimate or not. The mitigation strategy therefore focused on interpreting the information directly visualized by the QR code scanner. Payment requests, however, require users to perform an additional trust decision after the QR code has already been opened. Instead of evaluating only the information presented by the QR code scanner, users must also determine whether the payment recipient displayed by the payment application corresponds to the intended recipient.

Consequently, the security decision no longer depends exclusively on the awareness intervention, but also on the information exposed by the payment application's user interface. QRNope already attempts to address this challenge by introducing the concept of *impersonation-resistant identifiers* and explicitly recommending that users should not rely on names, profile pictures, logos, or similar graphical elements when authorizing a payment. Instead, Rule 4 recommends avoiding payments through apps that do not present such identifiers, as the Venmo pretext used was. Nevertheless, the present evaluation suggests that this recommendation should receive greater emphasis in future versions of QRNope. Unlike the phishing domains, which are accompanied by multiple visual examples illustrating both legitimate and phishing cases, the payment section relies primarily on visual examples of apps that do not have impersonation-resistant identifiers. Future versions of QRNope could therefore reinforce the existing recommendation by explicitly contrasting payment applications that show impersonation-resistant identifiers with applications that do not do so. Such an addition would not introduce a new mitigation strategy, but rather illustrate more clearly how the existing recommendation should be applied in practice.

Nevertheless, the results of the payment pretext also indicate that improving QRNope alone is unlikely to completely eliminate the remaining difficulties. Awareness interventions can only explain which information users should verify before authorizing a payment, but they cannot ensure that payment applications actually provide users with sufficient information to perform this verification. This issue became particularly apparent in the Venmo pretext, as the application displays a blue check mark at the bottom of the payment window. While this graphical element

may serve legitimate purposes within the application, it can also reasonably be interpreted by users as evidence that the recipient's identity has already been verified.
Consequently, users may incorrectly conclude that no additional verification is necessary, despite the application not exposing an impersonation-resistant identifier that allows the recipient's identity to be independently confirmed. This observation extends beyond the evaluation of QRNope and has broader implications for the design of payment applications. From a usable security perspective, awareness interventions should not compensate for interface designs that fail to provide users with the information required to make informed security decisions. Instead, both components should complement one another. Previous work by Berens et al. (2024b) already demonstrated that combining awareness interventions with tools produces better security outcomes than each of them on their own. The present results support this conclusion within the context of QRishing, as even after reading QRNope, participants remained dependent on the information presented by either the iOS Camera (for domains and phone numbers) or the payment application itself.
The results of Chapter 2 already highlighted that several participants reported difficulties interpreting the information visualized by the iOS Camera. The results from Section 6.3 suggest that also the payment app used as a pretext does not support users detecting security threats. Consequently, improving users' resilience against QRishing requires a broader effort from application developers to design interfaces that actively support secure decision making rather than relying exclusively on awareness interventions. Within the payment apps landscape, an example of such an interface is the Indian Unified Payments Interface (UPI) ecosystem, where payment applications display the bank-registered name associated with the destination account before the payment is authorized. Rather than relying exclusively on user-defined display names or graphical indicators, users can compare the expected recipient with the registered account holder before confirming the transaction. Although such approaches cannot completely prevent social engineering attacks, they provide users with impersonation-resistant identifiers that directly support the verification process recommended by QRNope. Future work should therefore investigate not only how awareness interventions can better communicate the importance of impersonation-resistant identifiers, but also how payment applications themselves can expose this information more effectively.
Besides the payment scenario, the misconception evaluation provides other insights for future versions of QRNope. One of the objectives during the development process was to explicitly identify misconceptions reported throughout the literature and address them within the awareness intervention. Yet, the descriptive results show that not all misconceptions had the same prevalence. For Misconceptions A, B, 7, 10, and 15, participants assigned to the Control group already achieved correct response rates ranging from 91.7% to 97.6%. Similarly, Misconception 1 was correctly evaluated by 81.0% of the Control group. These findings suggest that at least some misconceptions identified in Chapter 4.2 may no longer be widely held by users (or, at the very

least, by users from the US). Nevertheless, these misconceptions should not be removed, as they were still held by a subset of the participants.

In contrast, other misconceptions continue to represent an important challenge. The lowest performance across all three interventions was observed for Misconception 13, for which QRNope nevertheless achieved the highest proportion of correct responses (73.8%), compared with 68.4% for the FTC intervention and 61.9% for the Control group. Similar descriptive improvements were observed for Misconceptions 6, 7, and 14. However, despite these improvements, the generalized linear mixed-effects models did not identify statistically significant differences between QRNope and either comparison intervention. The results, however, do not necessarily suggest that additional misconception content is required, but rather that the existing misconception-related information should draw more attention during reading. Currently, misconception corrections within QRNope are generally embedded within the explanatory text or introduced through short *Note:* paragraphs. Although this presentation integrates naturally with the surrounding explanation, it may also reduce the likelihood that readers explicitly recognize these statements as corrections of common but incorrect mental models. Mayer (2019) argues that misconceptions should be explicitly identified through dedicated visual elements, drawing the readers' attention to them. Adopting a similar approach within future versions of QRNope may therefore improve the visibility of misconception-related information without substantially increasing either the reading time or the amount of content presented to users.

Finally, the results obtained for the FTC intervention highlight an interesting observation regarding the interpretation of awareness material. Based on the wording of the FTC intervention (shown in Section 6.2.1), it was initially expected that one of its recommendations could reinforce Misconception 1. However, the descriptive results do not support this expectation. Instead, participants assigned to the FTC group achieved the highest proportion of correct responses for this misconception (92.1%), outperforming both the Control group (81.0%) and the QRNope group (81.2%). Although the evaluation was not designed to determine the reasons behind the participants' interpretation of the FTC recommendation, the finding suggests that users do not necessarily interpret awareness material exactly as anticipated by its designers. Consequently, evaluating awareness interventions solely by examining whether their recommendations are technically correct may be insufficient, and qualitative data should be collected in the future to clarify the mental model of the participants.

7 Conclusion

This final chapter highlights and synthesizes the previous chapters by providing a summary of the results of each one (Section 7.1), and then exploring promising directions for future research building on the work presented in this thesis (Section 7.2). A brief final remark concludes the thesis.

7.1 Thesis Summary

The research presented in this thesis focuses on QRishing, users' awareness of this threat, how to address their low awareness, and whether the awareness intervention developed is effective.

7.1.1 QRishing Awareness Among QR Code Users

Chapter 2 investigated how users interact with QR codes in naturalistic settings. To this end, a field study was conducted in the metropolitan area of Chicago, USA, where pretexts with phishing QR codes were deployed in public locations and observed under real-world conditions. The study aimed at answering two questions: which factors attract and persuade users to scan QR codes, and what actions users take when scanning QR codes and accessing the URL encoded in them. A total of 42 participants voluntarily interacted with the QR codes and were subsequently asked to participate in a semi-structured interview. The interview data was then analyzed through an inductive qualitative coding approach.

Regarding the factors that led the participants to scan the QR codes, the results showed that the topic and the visual appearance of the pretext were the most attractive ones. This indicates that users determine whether to scan a QR code or not primarily based on non-security-relevant cues, such as the advertised content and the perceived legitimacy of the pretext. These observations are consistent with previous findings (e.g. Kersten et al. (2022)) on the influence of a believable and persuasive pretext on the success of a phishing attack.

Regarding the participants' behavior when accessing the content of a QR code, the results showed that the majority of the participants (67% of the codes) did not inspect the URL in their QR code scanner before accessing the related website, and that only 45.24% of the codes indicated that they inspected the URL within their browser. Furthermore, the interviews revealed that users often lacked a clear understanding of what even constitutes a security indicator, with 31% of the

codes referring to participants determining legitimacy based on the landing website overall look. This suggests that merely exposing users to URLs is insufficient without the knowledge required to interpret them, and that many users may simply have no awareness regarding what to base their decisions on.
More broadly, the findings suggested that users may be particularly susceptible to QRishing attacks. Compared to traditional phishing attacks such as emails, QR code interactions offer limited support for security-related decision making (e.g., default QR code scanners have no warnings). This issue appeared to be further exacerbated by the limited attention that QRishing has received in existing awareness interventions, which are primarily focused on email-based phishing attacks (as shown in, e.g., Mossano et al. (2020)). Taken together, the results highlighted two challenges in need of addressing: The first was the limited decision support provided by default QR code scanners. The second was the lack of awareness interventions specifically designed to address QRishing attack techniques and mitigation strategies. While recent research has started investigating technical approaches aimed at supporting users during QR code interactions, comparatively little work tackled the second challenge.
The findings of this chapter therefore established both the relevance of QRishing as a usable-security challenge and the need for dedicated awareness interventions. At the same time, the chapter raised the question of how such awareness interventions should be developed in a systematic, transparent, and replicable manner. Addressing this last challenge was the focus of the following chapter.

7.1.2 Framework for Developing Awareness Interventions

Chapter 3 addressed the problem of fragmentation within the awareness landscape. Previous research (e.g., Mossano et al. (2020) and Redmiles et al. (2020)) showed that awareness interventions often differ substantially in their content, both recommendation-wise and presentation-wise. These differences make it difficult to assess their quality, compare their results, and replicate their development. To address this issue, the chapter reviewed several frameworks for developing awareness interventions proposed in previous work, and then synthesized and expanded their contributions into the first version of a framework for developing awareness interventions (Framework 1.0).
Framework 1.0 integrated the elements proposed by several previous works (Yusoff (2010), Ghazvini and Shukur (2017), Jaeger (2018), Mayer (2019), Khader et al. (2021), Khando et al. (2021), Dialog für Cyber-Sicherheit (2022), and ENISA (2024c)) while expanding them through the inclusion of additional stages and explicit relations between them. The application of Framework 1.0 then highlighted several aspects requiring further refinement, such as the granularity of the stages and the representation of their relations. These refinements led to the second version of the framework for developing awareness interventions (Framework 2.0).

Framework 2.0 provided guidance regarding the identification of the attack techniques and mitigation strategies to cover, the consideration of misconceptions and security assumptions, the adaptation of awareness content to the target population, and the evaluation and revision of the resulting awareness intervention. Unlike several previous approaches, Framework 2.0 was not tied to a specific intervention format, domain, or security context, allowing it to be applied across a wide range of awareness intervention scenarios. Furthermore, it allowed developers to select the specific methods most appropriate for each stage of the process.
Framework 2.0 provided the methodological foundation for the remainder of this thesis and for the development of QRNope. However, applying Framework 2.0 required several inputs, including knowledge of the attack techniques present in the domain, common misconceptions regarding those attack techniques, and misconceptions regarding the corresponding mitigation strategies. The identification of these inputs was the focus of the following chapter.

7.1.3 Inputs Required by Framework 2.0: *All Attack Techniques in the Domain*, *Misconceptions on Attack Techniques*, and *Misconceptions on Mitigation Strategies*

Chapter 4 addressed the need for structured knowledge required to systematically develop QRishing awareness interventions using Framework 2.0. More specifically, the chapter focused on the stages *All attack techniques in the domain*, *Misconceptions on attack techniques*, and *Misconceptions on mitigation strategies*. To this end, the chapter presented the results of two systematic literature reviews and one systematic industry database review aimed at structuring existing knowledge on QRishing attack techniques and users' misconceptions about phishing and QRishing.
The first contribution of the chapter was the development of a taxonomy of QRishing attack techniques. By combining attack techniques from the academic literature with attack techniques collected from the MITRE ATT&CK® database aimed at iOS devices, the resulting taxonomy provided a structured representation of how QRishing attacks are delivered and which payloads they employ. The taxonomy allowed QRishing attacks to be categorized from three complementary perspectives: whether the QR code encountered was embedded or standalone, the vector used to deliver the payload, and the payloads themselves.
The second contribution of the chapter was the systematic identification and categorization of misconceptions about phishing. Building on an approach previously applied to password misconceptions in Mayer and Volkamer (2018), the review began by investigating misconceptions about phishing due to the broader research landscape compared to QRishing. The resulting analysis consolidated the misconceptions reported in these works into a structured list and identified which misconceptions were also relevant for QRishing. The results revealed misconceptions related to technical aspects of QRishing, URL analysis, security warnings, reporting procedures, and

technological defenses, while also highlighting broader themes such as users' lack of technical knowledge, confusion regarding security terminology, and excessive reliance on technological defenses.
Taken together, the taxonomy of QRishing attack techniques and the analysis of misconceptions provided the knowledge required to fulfill the corresponding stages of Framework 2.0. The systematic development of QRNope was then the focus of the following chapter.

7.1.4 Development of *QRNope*, a QRishing Awareness Intervention

Chapter 5 applied Framework 2.0 in Chapter 3 to develop *QRNope*, a QRishing awareness intervention aimed at lay users in the United States. QRNope focuses on QR code interactions in everyday-life scenarios and assumes the use of the default iOS Camera application on Apple mobile devices (e.g., iPhones). The goal of QRNope was to reach the Educational Element of the awareness maturity curve as interpreted in Schöni et al. (2024), i.e., QRNope provides readers with the knowledge necessary to identify the most relevant QRishing attack techniques and apply appropriate mitigation strategies. To maximize accessibility and reachability, the intervention was implemented as an accessible webpage.
The development process began by identifying the QRishing attack techniques relevant to both the target population and the context. To this end, the taxonomy developed in Chapter 4 was filtered according to the characteristics of the target population, the capabilities of iOS Camera, and the requirement that the intervention should be read and understood within approximately five minutes. Several attack techniques and mitigation strategies that were not actionable by the target population were excluded through explicit assumptions that delegated responsibility to developers, administrators, or online platforms. This process resulted in a set of QRishing attack techniques that could realistically be addressed through an awareness intervention understandable by lay users.
The development of QRNope also relied on the misconceptions identified in Chapter 4. These included misconceptions on both attack techniques and mitigation strategies relevant to the domain, context, and target population of QRNope. Together with the selected attack techniques and mitigation strategies, these misconceptions informed the content covered by the intervention. Building upon these inputs, QRNope was developed as a text-based awareness intervention structured around seven practical rules (eventually expanded to eight rules in the subsequent step). The intervention was designed using the Extended Parallel Process Model to balance threat communication with actionable mitigation strategies and thereby encourage protective behavior. The resulting content addresses the selected QRishing attack techniques, presents actionable mitigation strategies that address them, and corrects common misconceptions regarding both attack techniques and mitigation strategies. To support understanding and accessibility, the

intervention incorporates visual examples, employs simple language, and avoids unnecessary technical terminology.
The first version of QRNope was implemented as a self-contained static HTML webpage. Particular attention was devoted to accessibility requirements, resulting in an implementation compatible with assistive technologies and aligned with the Web Accessibility Guidelines version 2.2. Furthermore, a sanitization review was conducted before the feedback loops to ensure that the intervention itself would not expose readers to unnecessary risk.
The chapter then subjected QRNope to feedback from security experts, language experts, and members of the target population. The feedback loops resulted in several revisions of the presentation of the content, leading to QRNope v5.0. This version retained the content selected through the previous stages, while incorporating additional examples, clearer explanations, and revisions aimed at improving its understandability for the target population. The resulting intervention required approximately five minutes to read, achieved a Flesch Reading Ease Score of 71.5 and a Flesch-Kincaid Grade Level of 5.8, and was therefore suitable for readers without requiring specialized technical knowledge.
Overall, this chapter demonstrates how Framework 2.0 from Chapter 3 can be systematically applied to the development and refinement of a security awareness intervention. The evaluation of the final version of QRNope was the focus of the following chapter.

7.1.5 Evaluation of *QRNope*

Chapter 6 evaluated the final version of QRNope. Before conducting the final evaluation, two pilot studies were conducted to verify that the study implementation functioned as intended.
The final evaluation was conducted as an online between-subjects study with 240 participants from the US who use iOS devices. Namely, the study answered four research questions:

- **RQ1.** Are the readers of QRNope more effective than the readers of the placebo text in distinguishing phishing QR codes from legitimate ones?
- **RQ2.** Are the readers of QRNope more effective than the readers of the intervention from the FTC in distinguishing phishing QR codes from legitimate ones?
- **RQ3.** Is QRNope more effective than the placebo text at correcting the misconceptions on QRishing identified in the literature?
- **RQ4.** Is QRNope more effective than the intervention from the FTC at correcting the misconceptions on QRishing identified in the literature?

Participants were assigned to one of three groups: a Control group exposed to a placebo intervention, an FTC group exposed to existing QRishing recommendations published by the Federal Trade Commission, and a QRNope group exposed to QRNope. Following the intervention,

participants first evaluated 18 QR code pretexts, consisting of eight legitimate Domain-based pretexts, eight phishing Domain-based pretexts, one phone-number pretext, and one Venmo payment pretext. Subsequently, they judged a series of statements addressing misconceptions regarding QRishing attack techniques and mitigation strategies, including the mitigation strategy applicable to two attack techniques (*Malicious QR code Generator* and *Malicious QR code Scanner*), as these could not be evaluated through QR code pretexts. The study therefore evaluated whether QRNope improved participants' Overall Performance, Correct Acceptance, Correct Rejection, and their ability to correctly identify common misconceptions regarding QRishing.

The results showed that QRNope significantly improved participants' Overall Performance. Participants assigned to the QRNope group achieved an Overall Performance of 82.2%, compared with 72.8% in the FTC group and 68.9% in the Control group. No statistically significant difference was observed between the FTC group and the Control group. Consequently, both RQ1 and RQ2 could be answered positively.

The improvement was primarily associated with Correct Rejection. Participants assigned to the QRNope group achieved a Correct Rejection score of 77.6%, compared with 65.7% in the FTC group and 55.7% in the Control group. At the same time, participants assigned to the QRNope group also achieved the highest Correct Acceptance, with 81.9%, compared with 70.9% in the FTC group and 71.9% in the Control group. This indicates that QRNope supported the participants in correctly distinguishing phishing QR codes from legitimate ones.

The pretext-level analysis showed that QRNope had the highest Correct Rejection. The largest improvements compared with the Control group were observed for the phone-number pretext, the payment pretext, the subdomain pretext, the extended-domain pretext, and the different-TLD pretext. However, the payment pretext remained the lowest-performing phishing-related pretext in the QRNope group. This result suggests that payment-related QRishing attacks require additional support beyond the awareness intervention alone, because users also depend on the identity information exposed by the payment application's interface.

Regarding misconceptions, QRNope achieved the highest proportion of correct responses for five of the nine evaluated misconception statements. However, the generalized linear mixed-effects models did not show statistically significant differences between QRNope and either comparison group. Consequently, neither RQ3 nor RQ4 could be answered positively. For several misconceptions, the Control group already achieved high correct-response rates, suggesting that these misconceptions may not be equally prevalent within the evaluated population. For other misconceptions, especially Misconception 13, QRNope achieved descriptively higher performance than both comparison groups, but the difference did not reach statistical significance.

Overall, the chapter provides empirical evidence that QRNope is effective at improving users' ability to distinguish legitimate QR codes from phishing QR codes under the evaluated conditions. At the same time, the results identify two areas requiring further attention. First, payment-related QRishing attacks remain difficult because secure decision making depends not

only on user awareness, but also on whether payment applications expose impersonation-resistant identifiers. Second, misconception-related information may require a more visually salient presentation, as QRNope did not significantly outperform the comparison interventions in correcting misconceptions despite addressing them in its content. Together, these findings show that the framework-guided development process presented in Chapter 5 produced an effective awareness intervention, while also identifying concrete directions for improving both QRNope and the interfaces that users rely on when applying its recommendations.

7.2 Future Research

The research presented in this thesis opens several directions for future work.

Future work for the naturalistic study in Chapter 2 should investigate whether the observed behaviors generalize to other countries, cultures, and demographic groups. Previous research has already suggested that users from different populations may interact with QR codes differently (e.g., in Krombholz et al. (2015)). As such, understanding these differences may provide valuable insights into how QRishing awareness interventions should be adapted to different contexts.

Regarding Framework 2.0 in Chapter 3, future work should apply it to different security domains and investigate whether it consistently supports the creation of effective awareness interventions. Furthermore, comparing interventions developed through Framework 2.0 with interventions developed through alternative systematic methodologies may provide insight into the practical benefits of Framework 2.0's structured development processes.

As future work for the taxonomy presented in Section 4.1, additional platforms and deployment contexts could be used to expand it and provide a more comprehensive representation of QRishing attack techniques. In particular, the consideration of enterprise environments may expose users to attack techniques that differ from those commonly encountered by private users.

Similarly, the misconceptions review presented in Section 4.2 could be extended by systematically reviewing additional venues and databases, e.g., IEEE Xplore or Elsevier ScienceDirect. Such an expansion may identify further misconceptions and contribute to a more comprehensive understanding of users' misconceptions about phishing. Furthermore, future research may investigate misconceptions that are specific to QRishing and that do not emerge in the broader phishing literature.

Regarding QRNope, future work should first of all integrate the refinements mentioned in Section 6.4. Besides that, an interesting research direction would be the investigation of awareness interventions tailored to different target populations, technologies, and deployment scenarios. Namely, adapting QRNope to these different contexts, populations, and goals, could reveal aspects that were not initially considered because not present in the US. This would not only provide awareness interventions, or better, "QRNope adaptations" for those different situations, but also highlight potential peculiarities and differences that should be accounted for by other awareness

interventions as well. The evaluation in Chapter 6 could then be adapted to collect the data, providing a benchmark to compare these different QRNope adaptations.
Finally, the present evaluation focused on the immediate effectiveness of QRNope. Future work should therefore investigate the long-term retention of the acquired knowledge and evaluate whether the observed improvements persist over time and translate to real-world QR code interactions.

7.3 Final Remarks

Over the last decade, QR codes have evolved from a relatively niche technology into an integral component of everyday life. They are now routinely used for payments, authentication, information sharing, ticketing, advertising, and a wide variety of other activities. This widespread adoption has created considerable convenience for users and organizations alike, but at the same time, it has also created new opportunities for attackers. By concealing information behind a machine-readable representation, QR codes often require users to make security-relevant decisions while having access to fewer cues than those available in more traditional phishing contexts.
The research presented in this thesis was motivated by the observation that, despite the increasing prevalence of QRishing attacks, comparatively little attention has been devoted to understanding how users interact with QR codes, how QRishing attacks should be communicated to users, and how awareness interventions addressing QRishing should be developed. Existing awareness interventions have traditionally focused on email phishing (as shown in, e.g., Mossano et al. (2020)), while QRishing has often remained a secondary consideration. Furthermore, previous research has shown that awareness interventions frequently contain confusing recommendations that lack prioritization (as shown in, e.g., Redmiles et al. (2016)), potentially leading readers to security fatigue (as described in Stanton et al. (2016)).
To address these challenges, this thesis followed a systematic progression from problem identification to intervention development and evaluation. First, it investigated how users interact with QR codes in naturalistic settings and identified behaviors that may leave them exposed to successful QRishing attacks. Building upon these findings, the thesis then proposed a framework to support the systematic development, implementation, and evaluation of awareness interventions. The thesis then identified the knowledge required for the development of a QRishing awareness intervention through a taxonomy of QRishing attack techniques primarily aimed at iOS devices, and through a systematic review of misconceptions about phishing attack techniques and mitigation strategies. Finally, these results informed the development of QRNope, a QRishing awareness intervention specifically designed for lay users from the US who rely on the iOS Camera to scan QR codes. QRNope was then refined through a series of feedback loops with both experts and members of the target population, and evaluated in an online between-subjects study with 240 participants from the US. The evaluation showed that QRNope improved participants' overall

performance compared with both the FTC intervention and the placebo intervention. This indicates that users can be supported in following a more secure behavior during QR code interactions when QRishing-specific recommendations are selected, justified, and presented in a way that is actionable for the target population. At the same time, the evaluation also showed that awareness interventions do not remove the need for security-supportive interfaces, especially in situations where users must rely on the information exposed by QR code scanners or payment applications. Taken together, the results of this thesis contribute to a better understanding of QRishing from both a technical and a human-centered perspective. Beyond the specific intervention developed in this work, the thesis also demonstrates the value of approaching awareness intervention development as a systematic and transparent process. As QR codes continue to become increasingly integrated into everyday life, supporting users in making informed security decisions will remain an important challenge for both researchers and practitioners. It is hoped that the methods, results, and insights presented in this thesis will contribute to future efforts aimed at addressing this challenge and improving users' resilience against QRishing attacks.

A Appendix

A.1 Naturalistic Study – Verbal Consent Script

Dear Madam/Sir,

My name is Dr. Filipo Sharevski, I am a researcher from DePaul University. My contact number is 312-362-1075 and my email is fsharevs@depaul.edu. I am conducting a research study about the decision people to visit a website when interacting with QR codes and I noticed you just scanned a QR code in the nearby poster placed at [PLACE].
The purpose of the research is to learn more about how people interact with QR codes in general. I am recruiting only volunteers for this research study and you can participate on your own volition. You are eligible to participate if you are 18 years or above old, you are from United States, you are able to understand and converse in English language, you holds a internet capable smartphone and you have used a QR code to access a website at least once (which you just did).
I will ask you several open-ended questions and collect some personal information about you. The interview is anonymous and you are asked to participate on a voluntarily basis. If there is a question you do not want to answer, you may ask us to skip it. Your information will be kept confidential and stored in a secured computer under password protection and with encrypted files. The data will be kept de-identified. The participation will take between 5-10 minutes.
I have obtained an IRB protocol #IRB-2022-813approval and version date of the document. If you like to contact anyone from our institution regarding this research you may contact Jessica Bloom.

A.2 Naturalistic Study – Interview Script

1. What attracted you to scan this particular QR code? Please explain in your own words.
2. Once you scanned it, what were the next steps you took?
    a) **Clicked on the website**: Did you pay attention on any particular website elements?
    b) **Abandon it**: What were the reasons you decided not to proceed and visit the associated website?

3. **Inspection**. When you scan a QR code, do you inspect the domain URL in the displayed frame of the scanner (e.g. do you read the domain displayed on the linked website in the scanning app [yellow bar])?
   a) **Yes**: Do you also inspect the URL further when you open it in the browser?
   b) **No**: Do you perhaps inspect the URL when you open it in the browser?
4. Demographics: Age, gender, race/ethnicity, education, computer proficiency

A.3 Naturalistic Study – Codebook

- **Pretext.** Codes related to the pretext used to deliver the phishing attack.
  - **Topic.** The topic of the Ukrainian war attracted the user.
    - **Humanitarian.** The humanitarian opportunity attracted the user.
    - **Geopolitical.** The connection to the geopolitical situation attracted the user.
  - **General.** A generic curiosity to see what it was about attracted the user.
  - **Color.** The coloration of the pretext attracted the user.
  - **Text.** The text of the pretext attracted the user.
  - **Size.** The size of the pretext attracted the user.
  - **Affective.** The affective impression of the pretext attracted the user.
- **Behavior.** This category contains codes describing the general behavior of the users during their interactions with QR codes, the QR code vectors and the websites associated with them.
  - **URL inspected - App.** The user inspected the URL in the QR code scanner.
  - **URL inspected - Abandoned** The user inspected the URL in the QR code scanner and decided to abandon the interaction.
  - **URL not inspected - App.** The user did not inspect the URL in the QR code scanner.
  - **URL not inspected - UI.** The user did not inspect the URL in the QR code scanner due to inefficiencies with the UI.
  - **URL inspected - Browser** The user inspected the URL in the webaddress bar.
  - **URL not inspected - Website look.** The user did not inspect the URL in the QR code scanner or in the webaddress bar, but checked the website content.

    - ⁎ **Overall look.** The user checked the overall look of the website to determine legitimacy.
    - ⁎ **Read text.** The user read the text on the website to determine legitimacy.
- **Unknown actions.** The user did not disclose sufficient information to infer the outcome of their interaction with the QR code.

A.4 QRNope Evaluation - Instructions to the Participants

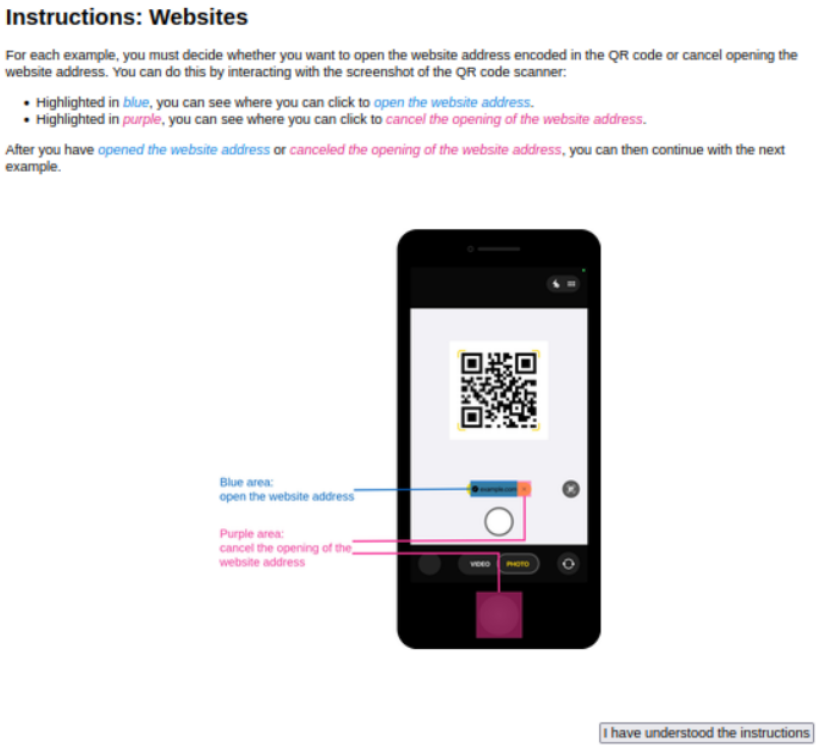


Figure A.1: Instructions on how to interact with the emulated iOS device in the URL-block part of QRNope evaluation.

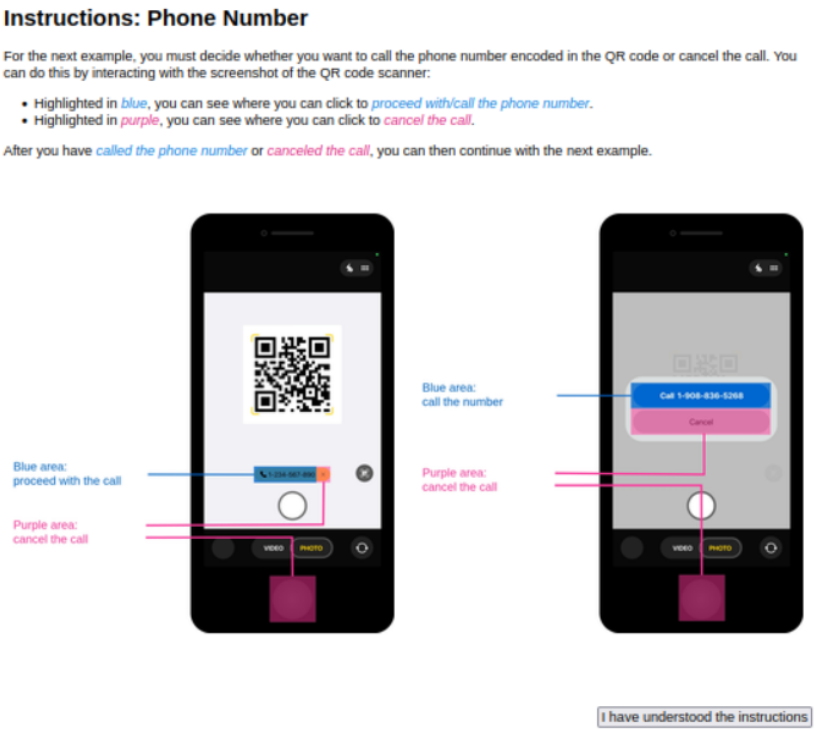


Figure A.2: Instructions on how to interact with the emulated iOS device in the Phone Number-block part of QRNope evaluation.

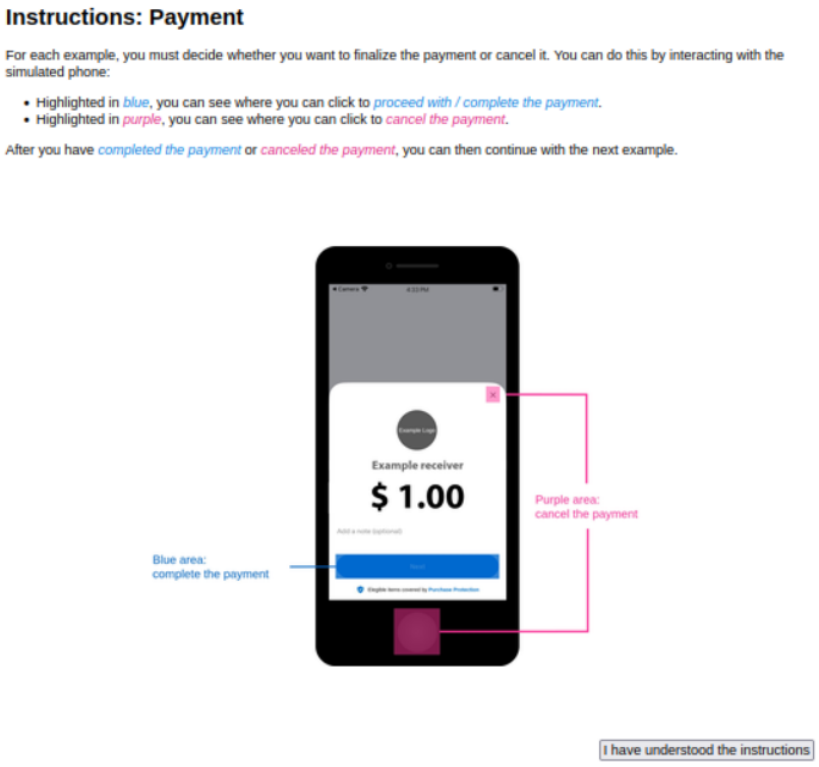


Figure A.3: Instructions on how to interact with the emulated iOS device in the Payment-block part of QRNope evaluation.

A.5 QRNope Evaluation - Demographics Tables

Table A.1: Sources from which participants reported having previously learned about phishing. Multiple categories could be assigned to a single response.

Source of previous phishing knowledge	Frequency
Workplace / employer / formal training	43
Videos (including YouTube and training videos)	28
General online sources	24
Formal education	11
News media	11
Self-directed observation / experience	8
Social media	6
Personal contacts	6
Professional / IT experience	5
Financial institutions / IT reminders	3
Other / unclear	4
Cannot remember / unspecified	1

Table A.2: Sources from which participants reported having previously heard about QRishing. Multiple categories could be assigned to a single response.

Source of previous QRishing knowledge	Frequency
News media	39
Workplace / employer / training	28
General online sources	21
Social media	18
Family, friends, or colleagues	13
Formal education	7
Cannot remember / unspecified	6
Professional experience	4
Financial institutions / security alerts	2
General awareness / everyday experience	2

List of Figures

2.1 Pretexts used in Vidas et al. (2013). From left to right: standalone QR code, standalone QR code with instructions how to scan it, QR code with a social network study deception, tear-off flyer with URL to survey. . . . 9
2.2 Posters used by Bekavac et al. (2024) in their study. For both sets, from left to right: tampered-looking QR code, QR code, SafeQR, tampered-looking SafeQR 11
2.3 Study design followed to conduct the naturalistic study. 12
2.4 Phishing poster embedding the legitimate-looking URL type. 14
2.5 Landing webpage as seen on a mobile phone. 15
2.6 Naturalistic Study – RQ2. Actions Around QR Codes: Results Per URL-Type 24
3.1 Ghazvini and Shukur (2017) framework on awareness interventions for the health sector. 29
3.2 Jaeger (2018) framework on awareness interventions development). 30
3.3 The framework presented in Mayer (2019). 30
3.4 Khader et al. (2021) framework for the academic context. 31
3.5 Khando et al. (2021) framework for private sector and public sector. 32
3.6 Representation of the process presented in Dialog für Cyber-Sicherheit (2022). 32
3.7 The framework security interventions from ENISA (2024c). 33
3.8 The version 1.0 of the framework for developing awareness interventions. 34
3.9 Version 2.0 of the Framework for Developing Information Security Awareness Interventions. 47
4.1 The PRISMA representation of the methodology from Athanzah et al. (2026). . . . 57
4.2 The attack techniques identified by Athanzah et al. (2026). 57
4.3 The process followed to collect the studies included in the systematic literature review. 59
4.4 The process followed to identify the MITRE attack techniques included. Modified from Page et al. (2021). 61
4.5 Visual representation of the taxonomy. 63
4.6 Structure of a URL and its parts. 67
4.7 The process followed to collect the studies included in the systematic literature review. 76

5.1 *Step 1* of the framework for developing awareness interventions. The *Security assumptions* stage is missing due to space. 92
5.2 Awareness maturity curve from Schöni et al. (2024). Highlighting in the original. QRNope's goal is to cover the full "Education Element". 96
5.3 *Step 2* of the framework for developing awareness interventions. The *Security assumptions* stage is missing due to space. 99
5.4 *Step 3* of the framework for developing awareness interventions. The *Security assumptions* stage is missing due to space. 104
5.5 *Step 4* of the framework for developing awareness interventions. 113
5.6 *Step 5* of the framework for developing awareness interventions. 116
5.7 The Extended Parallel Process Model from Witte (1992) (image from Gore and Bracken (2005)). 117
5.8 First version of the QRNope content. 124
5.9 *Step 6* of the framework for developing awareness interventions. 125
5.10 *Step 7* of the framework for developing awareness interventions. The evaluation part is omitted, because part of Chapter 6. 128
5.11 QRNope v5.0. 136
6.1 Placebo text used in the QRNope evaluation. 142
6.2 Awareness intervention from the FTC used in the QRNope evaluation. 144
6.3 Study design followed to evaluate QRNope. 145
6.4 Example of the interaction in the Domain-block part of the study. 150
6.5 Heatmap of the participants' performance for each pretext across the three study groups. Higher percentages are represented by lighter colors. 160
A.1 Instructions on how to interact with the emulated iOS device in the URL-block part of QRNope evaluation. 182
A.2 Instructions on how to interact with the emulated iOS device in the Phone Number-block part of QRNope evaluation. 183
A.3 Instructions on how to interact with the emulated iOS device in the Payment-block part of QRNope evaluation. 184

List of Tables

2.1 Naturalistic Study – RQ1. Pretext Persuasion Factors Analysis Results 22
2.2 Naturalistic Study – RQ2. Actions Around QR Codes: Overall Results 23
4.1 Attack techniques from Veit et al. (2026a), with brief description and example. . . . 68
4.2 Collected attack techniques in the domain with examples or descriptions. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*. 69
4.3 Collected attack techniques in the domain, their examples or descriptions, how iOS Camera show them, and if the URL visualization addresses them. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*. The last two columns show how the iOS Camera would show the various attack techniques in its interface. 71
4.4 The final list of attack techniques in the domain. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*. 72
4.5 Misconceptions classification between attack techniques and mitigation strategies, and rationale explaining why each belief is considered a misconception. 85
4.6 Misconceptions not relevant for the QRishing content, and the rationale why it is so. 86
4.7 Misconceptions relevant for the QRishing content, and the rationale why it is so. 87
4.8 Misconceptions relevant for the QRishing content, and the rationale why it is so. 87
5.1 All attack techniques in the domain with descriptions or examples. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*. 96
5.2 Attack techniques assumed to be addressed at the end of Step 1. 98
5.3 Attack techniques remaining at the end of *Step 1*. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*. 99
5.4 Attack techniques assumed to be addressed at the end of Step 2. 103
5.5 Attack techniques remaining at the end of *Step 2*. MQRC stands for *Malicious QR Code*, and LQRC stands for *Legitimate QR Code*. 103

5.6 Mitigation strategies found in the attack techniques sources. Type denotes whether the mitigation strategy was (1) directly identified in the literature, (2) synthesized from similar mitigation strategies, or (3) derived from the corresponding attack technique. 106
5.7 Actionable mitigation strategies. Highlighted in orange are those mitigation strategies conflicting with previous security assumptions, in magenta are those too technically complex for the target population, in cyan are the so-called "unreasonable mitigation strategies," and in blue the so called "interruption mitigation strategies". Interruption mitigation strategies marked with (∗) are treated differently due to the outcome being the same. 107
5.8 Attack techniques with actionable mitigation strategies remaining. 110
5.9 Attack techniques assumed to be addressed at the end of Step 3. 112
5.10 Misconceptions on attack techniques, whether they require coverage in QRNope, and rationale for those that do not. 115
6.1 List of the organizations used in the evaluation, their type, the pretext type, and the corresponding domains and subdomains. 148
6.2 Demographic characteristics of the participants. 157
6.3 Descriptive statistics of the participants' performance across the three study groups. The table also reports the difference between the Correct Acceptance and Correct Rejection. The overall performance is computed over all 18 pretexts, whereas the Correct Acceptance and Correct Rejection percentages are each computed over 10 pretexts. 158
6.4 Percentage of participants that judge correctly the statements about the *Malicious QR Code Scanner* and *Malicious QR Code Generator* attack techniques. 160
6.5 Descriptive statistics of the participants' overall performance grouped by the participants' demographic data. 162
6.6 Percentage of participants correctly evaluating each misconception statement. . . . 164
A.1 Sources from which participants reported having previously learned about phishing. Multiple categories could be assigned to a single response. 185
A.2 Sources from which participants reported having previously heard about QRishing. Multiple categories could be assigned to a single response. 185

List of Publications

Christopher Beckmann, Benjamin Berens, Niklas Kühl, Peter Mayer, Mattia Mossano, and Melanie Volkamer. Design and Evaluation of an Anti-phishing Artifact Based on Useful Transparency. In *Socio-Technical Aspects in Security*, pages 113–133, Cham, 2025. Springer Nature Switzerland. doi: 10.1007/978-3-031-83072-3_7.

Benjamin Berens, Mattia Mossano, and Melanie Volkamer. Phishing awareness and education – when to best remind? In *Proceedings of the 2022 Workshop on Usable Security and Privacy*, USEC '22, pages 1–15, San Diego, CA, USA, 2022. Internet Society. doi: 10.14722/usec.2022.23075.

Benjamin M. Berens, Mattia Mossano, and Melanie Volkamer. Taking 5 minutes protects you for 5 months: Evaluating an anti-phishing awareness video. *Computers & Security*, 137:103620, 2024a. doi: 10.1016/j.cose.2023.103620.

Benjamin Maximilian Berens, Florian Schaub, Mattia Mossano, and Melanie Volkamer. Better together: The interplay between a phishing awareness video and a link-centric phishing support tool. In *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI '24, pages 1–14, New York, NY, USA, 2024b. Association for Computing Machinery. doi: 10.1145/3544548.3581460.

Benjamin Maximilian Berens, Mattia Mossano, Maxime Veit, Anne Hennig, and Melanie Volkamer. Guidelines for Usable Security Interventions. *Festschrift in Honour of M. Angela Sasse*, pages 21–28, 2026. doi: 10.5445/IR/1000193085.

Mark Bohlender, Raphael Morisco, Mattia Mossano, Thorsten Schwarz, and Melanie Volkamer. SMILE4VIP: Intervention to Support Visually Impaired Users Against Phishing. In *Workshop on Accessible Security & Privacy*, volume 00 of *WASP 2024*, pages 650–657, New York, NY, US, 2024. IEEE. doi: 10.1109/eurospw61312.2024.00079.

Anne Hennig, Mattia Mossano, Lukas Aldag, Maxime Veit, and Melanie Volkamer. "It sounds like automation" – Towards Users' Understanding of Adaptive Systems. In *Proceedings of the Mensch Und Computer 2025*, MuC '25, New York, NY, US, 2025. ACM. doi: 10.5445/IR/1000184818.

Mattia Mossano and Melanie Volkamer. Literature Review: Misconceptions About Phishing. In *Symposium on Human Aspects of Information Security and Assurance*, HAISA '24, pages 215–228, Cham, 2025. Springer. doi: 10.1007/978-3-031-72559-3_15.

Mattia Mossano, Kami Vaniea, Lukas Aldag, Reyhan Düzgün, Peter Mayer, and Melanie Volkamer. Analysis of publicly available anti-phishing webpages: contradicting information, lack of concrete advice and very narrow attack vector. In *European Symposium on Usable Security*, EuroUSEC 2020, pages 130–139, New York, NY, US, 2020. IEEE. doi: 10.1109/eurospw51379.2020.00026.

Mattia Mossano, Benjamin Berens, Philip Heller, Christopher Beckmann, Lukas Aldag, Peter Mayer, and Melanie Volkamer. Smile - Smart Email Link Domain Extractor. In *Proceedings of the 2021 Wokshop on Security, Privacy, Organizations, and Systems Engineering*, SPOSE 2021, pages 403–412, Cham, CH, 2022. Springer. doi: 10.1007/978-3-030-95484-0_23.

Mattia Mossano, Oksana Kulyk, Benjamin Maximillian Berens, Elena Marie Häußler, and Melanie Volkamer. Influence of URL Formatting on Users' Phishing URL Detection. In *European Symposium on Usable Security*, EuroUSEC 2023, pages 318–333, New York, NY, USA, 2023. ACM. doi: 10.1145/3617072.3617111.

Mattia Mossano, Anne Hennig, Fabian Lukas Ballreich, Benjamin Maximillian Berens, Filipo Sharevski, Angela Martina Sasse, and Melanie Volkamer. A framework for developing information security awareness measures. In *Symposoium on usable privacy and security – poster session*, SOUPS 2025, Berkeley, CA, US, 2025. USENIX. URL `https://www.usenix.org/conference/soups2025/presentation/mossano-poster`.

Mattia Mossano, Jennifer Vander Loop, and Filipo Sharevski. Like phishing in a pond: Preliminary results of a realistic hybrid qrishing simulation campaign in a large business organization. In *Symposoium on usable privacy and security – poster session*, SOUPS 2026, Berkeley, CA, US, 2026a. USENIX.

Mattia Mossano, Maxime Fabian Veit, Tobias Länge, Benjamin Maximillian Berens, Filipo Sharevski, and Melanie Volkamer. Development, Evaluation, and Implementation of SEQR – a Usable Secure QR code Scanner. In *Conference on Human Factors in Computing Systems*, CHI 2026, pages 1–33, New York, NY, US, 2026b. ACM. doi: https://doi.org/10.1145/3772318.3793213.

Benjamin Reinheimer, Lukas Aldag, Peter Mayer, Mattia Mossano, Reyhan Duezguen, Bettina Lofthouse, Tatiana von Landesberger, and Melanie Volkamer. An investigation of phishing awareness and education over time: When and how to best remind users. In *Proceedings of the 2020 Symposium on Usable Privacy and Security*, SOUPS '20, pages 259–284, Berkeley,

CA, USA, 2020. USENIX Association. URL https://www.usenix.org/conference/soups2020/presentation/reinheimer.

Filipo Sharevski, Mattia Mossano, Maxime Veit, Gunther Schiefer, and Melanie Volkamer. Exploring phishing threats through qr codes in naturalistic settings. In *Symposium on Usable Security*, USEC 2024, page 17, Reston, VA, US, 2024. Internet Society. doi: 10.14722/usec.2024.23050.

Maxime Veit, Mattia Mossano, Tobias Länge, and Melanie Volkamer. Comprehensive List of User Deception Techniques in Emails, 2026a.

Maxime Fabian Veit, Mattia Mossano, Tobias Länge, and Melanie Volkamer. Anatomy of a phish – a comprehensive list of user deception techniques in emails. In *Symposoium on usable privacy and security – poster session*, SOUPS 2026, Berkeley, CA, US, 2026b. USENIX.

Bibliography

Rahmad Abdillah, Zarina Shukur, Masnizah Mohd, and Mohd Zamri Murah. Phishing classification techniques: A systematic literature review. *IEEE Access*, 10(1):41574–41591, 2022. doi: 10.1109/ACCESS.2022.3166474.

Rufai Ahmad and Sotirios Terzis. Understanding Phishing in Mobile Instant Messaging: A Study into User Behaviour Toward Shared Links. In *Human Aspects of Information Security and Assurance*, HAISA '22, pages 197–206, Cham, 2022. Springer. doi: 10.1007/978-3-031-12172-2_15.

Rana Alabdan. Phishing attacks survey: Types, vectors, and technical approaches. *Future Internet*, 12(10):168, 2020. doi: 10.3390/fi12100168.

Sara Albakry, Kami Vaniea, and Maria K. Wolters. What is this url's destination? empirical evaluation of users' url reading. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–12, New York, NY, US, 2020. ACM. doi: 10.1145/3313831.3376168.

Ahmed Aleroud and Lina Zhou. Phishing environments, techniques, and countermeasures: A survey. *Computers and Security*, 68(1):160–196, 2017. doi: 10.1016/j.cose.2017.04.006.

Mohamed Alsharnouby, Furkan Alaca, and Sonia Chiasson. Why phishing still works: User strategies for combating phishing attacks. *International Journal of Human-Computer Studies*, 82(1):69–82, 2015. doi: 10.1016/j.ijhcs.2015.05.005.

Amnesty International. Ukraine crisis: Help protect civilians. https://donate.amnestyUS.org/page/122555/donate/1?ea.tracking.id=W23XXADEVR0P&en_og_source=W23XXADEVR0P&supporter.appealCode=W23XXADEVR0P&gad_source=1&gclid=CjwKCAiA0syqBhBxEiwAeNx9NxTVyadhgV4YqhHz4T6r0SjBGPo-q50sa0XccRimHbjfqdHD69qvGhoChAkQAvD_BwE&gclsrc=aw.ds, 2026. Accessed on 08.05.2026.

Anti-Phishing Working Group. Phishing Activity Trends Report - 1st quarter 2026. Technical Report 4th quarter 2025, APWG, 2026.

Apple. Qr code recognition on ios 11 - tech talks - videos, 2026. URL `https://developer.apple.com/videos/play/tech-talks/206/`.

Arvinsani Athanzah, Ryo Pratama Putra, Michael Yohannes Turisno, and Dedy Syamsuar. Risk of QR Code Phishing: A Systematic Literature Review on Common Attacks, Awareness, and Future Prevention Methods. In *The International Conference on Information Systems and Intelligent Technologies*, volume 1785, pages 52–62. Springer Nature Switzerland, Cham, 2026. doi: 10.1007/978-3-032-14750-9_5.

Andrey Averin and Natalya Zyulyarkina. Malicious qr-code threats and vulnerability of blockchain. In *Proceedings of the 2020 Global Smart Industry Conference*, GloSIC '20, pages 82–86, New York, NY, US, 2020. IEEE. doi: 10.1109/GloSIC50886.2020.9267840.

Ruth Korede Ayeni, Ayodele Ariyo Adebiyi, Julius Olatunji Okesola, and Ennmanuel Igbekele. Phishing Attacks and Detection Techniques: A Systematic Review. In *2024 International Conference on Science, Engineering and Business for Driving Sustainable Development Goals (SEB4SDG)*, pages 1–17, 2024. doi: 10.1109/SEB4SDG60871.2024.10630203.

Wenying Bao, Wenbin Yao, Ming Zong, and Dongbin Wang. Cross-site scripting attacks on android hybrid applications. In *Proceedings of the 2017 Conference on Cryptography, Security and Privacy*, CSP '17, pages 56–61, New York, NY, US, 2017. ACM. doi: 10.1145/3058060.3058076.

Luke Barr. FBI warns criminals are using fake QR codes to scam users, 2022. URL `https://abcnews.go.com/Politics/fbi-warns-criminals-fake-qr-codes-scam-users/story?id=82371866`. Accessed: 2026.06.22.

Irving Ricardo Barron and Gaurav Sharma. Quashing Quishing Attacks Using Self-Authenticating Dual-Modulated QR Codes. *IEEE Security & Privacy*, 24(1):26–34, 2026. doi: 10.1109/MSEC.2025.3530487.

Luka Jure Lars Bekavac, Simon Mayer, and Jannis Strecker. Qr-code integrity by design. In *Extended Abstracts of the 2024 CHI Conference on Human Factors in Computing Systems*, CHI EA '24, pages 1–9, New York, NY, US, 2024. ACM. doi: 10.1145/3613905.3651006.

Debalina Bera, Obi Ogbanufe, and Dan J. Kim. Towards a thematic dimensional framework of online fraud: An exploration of fraudulent email attack tactics and intentions. *Decision Support Systems*, 171(1):113977, 2023. doi: 10.1016/j.dss.2023.113977.

Tim Berners-Lee, Roy T. Fielding, and Larry M. Masinter. Uniform Resource Identifier (URI): Generic Syntax. Request for Comments RFC 3986, Internet Engineering Task Force, 2005.

Marcus Beyer, Sarah Ahmed, Katja Doerlemann, Simon Arnell, Simon Parkin, Angela Sasse, and Neil Passingham. Awareness is Only the First Step. Technical report, Hewlett Packard, 2015.

Akashdeep Bhardwaj, Varun Sapra, Aman Kumar, Naman Kumar, and S Arthi. Why is phishing still successful? *Computer Fraud & Security*, 2020(9):15–19, 2020. doi: 10.1016/s1361-3723(20)30098-1.

Akashdeep Bhardwaj, Fadi Al-Turjman, Varun Sapra, Manoj Kumar, and Thompson Stephan. Privacy-aware detection framework to mitigate new-age phishing attacks. *Computers & Electrical Engineering*, 96(1):107546, 2021. doi: 10.1016/j.compeleceng.2021.107546.

Leyla Bilge, Thorsten Strufe, Davide Balzarotti, and Engin Kirda. All your contacts are belong to us: Automated identity theft attacks on social networks. In *Conference on World Wide Web*, WWW '09, pages 551–560, Madrid, Spain, 2009. ACM. doi: 10.1145/1526709.1526784.

Eric Blancaflor, Mark Brendon Calida, Matthew Chan, Nicole La Ysico, and Goten Sup An Keith. Impact analysis and attack simulation on quishing (a qc code phishing) using qrljacker. In *Proceedings of the 2024 International Conference on Electrical, Computer and Energy Technologies*, ICECET '24, pages 1–5, New York, NY, US, 2024. IEEE. doi: 10.1109/icecet61485.2024.10698628.

Marc Brysbaert. How many words do we read per minute? A review and meta-analysis of reading rate. *Journal of Memory and Language*, 109:104047, December 2019. doi: 10.1016/j.jml.2019.104047.

Alastair Campbell, Chuck Adams, Rachael Bradley Montgomery, Michael Cooper, and Andrew Kirkpatrick. Web Content Accessibility Guidelines (WCAG) 2.2, 2024. URL `https://www.w3.org/TR/WCAG22/`.

Keegan Cargrill, Tamirat Abegaz, Luis Cueva Parra, and Richelle DaSouza. Scan Me: QR Codes as Emerging Malware Delivery Mechanism. In *Proceedings of the Future Technologies Conference (FTC) 2023, Volume 2*, pages 611–617, Cham, 2023. Springer Nature Switzerland. doi: 10.1007/978-3-031-47451-4_44.

Matthieu Chartier. Standard Poster Size for Academic Conferences: Specs, Layout and Submission Rules, 2025. URL `https://fourwaves.com/blog/standard-poster-size-for-academic-conference/`. Accessed: 2025.05.07.

Kuo-Chien Chou and Ran-Zan Wang. The nested qr code. *IEEE Signal Processing Letters*, 27 (1):1230–1234, 2020. doi: 10.1109/lsp.2020.3006375.

Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1):37–46, 1960. doi: 10.1177/001316446002000104.

CORE. ICORE Conference Portal | International Conference on Human Factors in Computing Systems, 2026. URL https://portal.core.edu.au/conf-ranks/1053/.

CounterPoint. US Smartphone Market Share. Technical report, CounterPoint, 2023. URL https://www.counterpointresearch.com/us-smartphone-shipments-decline-in-q1-2023-amid-high-inflation-inventory-correction-apple-share-up/. Accessed on 25.06.2026.

Adrian Dabrowski, Katharina Krombholz, Johanna Ullrich, and Edgar R. Weippl. Qr inception. In *Proceedings of the 2014 Workshop on Security and Privacy in Smartphones & Mobile Devices*, SPSM '14, pages 3–10, New York, NY, US, 2014. ACM. doi: 10.1145/2666620.2666624.

Denso Wave Incorporated. Qr code development history, 2025. URL https://www.denso-wave.com/en/technology/vol1.html.

Rachna Dhamija, J. Doug Tygar, and Marti Hearst. Why phishing works. In *Proceedings of the 2006 CHI Conference on Human Factors in Computing Systems*, CHI '06, pages 581–590, New York, NY, US, 2006. ACM. doi: 10.1145/1124772.1124861.

Dialog für Cyber-Sicherheit. Leitfaden des Workstreams „Effektive IT-Security-Awareness: Wirksam ein Bewusstsein für Risiken schaffen". Technical report, Bundesamt für Sicherheit in der Informationstechnik, 2022. URL https://www.dialog-cybersicherheit.de/storage/uploads/ws21-endprodukte/WS4%20-%20Effektive%20IT-Security%20Awareness.pdf.

Joanmarie Diggs, James Nurthen, Michael Cooper, and Carolyn MacLeod. Accessible Rich Internet Applications (WAI-ARIA) 1.2, 2023. URL https://www.w3.org/TR/wai-aria-1.2.

Verena Distler. The Influence of Context on Response to Spear-Phishing Attacks: an In-Situ Deception Study. In *Conference on Human Factors in Computing Systems*, CHI '23, pages 1–18, New York, NY, US, 2023. ACM. doi: 10.1145/3544548.3581170.

Rishabh Dudheria. Evaluating features and effectiveness of secure qr code scanners. In *Proceedings of the 2017 Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery*, CyberC '17, pages 40–49, New York, NY, US, 2017. ACM. doi: 10.1109/cyberc.2017.23.

Marc Dupuis and Karen Renaud. Scoping the ethical principles of cybersecurity fear appeals. *Ethics and Information Technology*, 23(3):265–284, 2021. doi: 10.1007/s10676-020-09560-0.

Gregory W. Edwards, Michael J. Gonzales, and Marc A. Sullivan. Robocalling: STIRRED AND SHAKEN! - An Investigation of Calling Displays on Trust and Answer Rates. In *Conference*

on Human Factors in Computing Systems, CHI '20, pages 1–12, New York, NY, US, 2020. ACM. doi: 10.1145/3313831.3376679.

Ayman El Aassal and Rakesh Verma. Spears Against Shields: Are Defenders Winning the Phishing War? In *Proceedings of the ACM International Workshop on Security and Privacy Analytics - IWSPA '19*, pages 15–24, Richardson, Texas, US, 2019. ACM Press. doi: 10.1145/3309182.3309191.

Diana Elagina. US smartphone market share by vendor 2025, 2025. URL https://www.statista.com/statistics/620805/smartphone-sales-market-share-in-the-us-by-vendor/.

ENISA. Best Practices Cyber Crisis Management. Technical report, European Union Agency for Cybersecurity, 2024a. URL https://data.europa.eu/doi/10.2824/767828. Accessed on 11.05.2026.

ENISA. Cybersecurity Education Maturity Assessment. Technical report, European Union Agency for Cybersecurity, 2024b. URL https://op.europa.eu/en/publication-detail/-/publication/3728a51a-2487-11ef-a195-01aa75ed71a1/language-en. Accessed on 11.05.2026.

ENISA. Guideline on Security Measures under the EECC- 4th edition. Technical report, European Union Agency for Cybersecurity, 2024c. URL https://doi.org/10.2824/44013.

European Committee for Standardization. EN 301549:2021. Accessibility requirements for ICT products and services - AccessibleEU, 2021. URL https://accessible-eu-centre.ec.europa.eu/content-corner/digital-library/en-3015492021-accessibility-requirements-ict-products-and-services_en.

European Parliament and Council. Directive (EU) 2018/1972 of the European Parliament and of the Council of 11 December 2018 establishing the European Electronic Communications Code (Recast) (Text with EEA relevance). http://data.europa.eu/eli/dir/2018/1972/2024-10-18, 2024. Accessed 14.06.2026.

FCC. TRACED Act Implementation, 2024. URL https://www.fcc.gov/TRACEDAct.

Federal Bureau of Investigation. Unsolicited Packages Containing QR Codes Used to Initiate Fraud Schemes, 2025. URL https://www.fbi.gov/investigate/cyber/alerts/2025/unsolicited-packages-containing-qr-codes-used-to-initiate-fraud-schemes.

Federal Bureau of Investigation. 2025 Internet Crime Report. Technical report, FBI, 2026.

Ana Ferreira, Lynne Coventry, and Gabriele Lenzini. Principles of persuasion in social engineering and their use in phishing. In *Proceedings of the 2015 Conference on Human Aspects of Information Security, Privacy, and Trust*, HAS '15, pages 36–47, Cham, CH, 2015. Springer. doi: 10.1007/978-3-319-20376-8_4.

Andy Field. *Discovering Statistics Using IBM SPSS Statistics*. Sage Publications, ThoUSnd Oaks, CA, US, 2013.

Rudolf Flesch. How to write plain english: Let's start with the formula. *University of Canterbury*, 1979.

Riccardo Focardi, Flaminia L. Luccio, and Heider A. M. Wahsheh. Usable cryptographic QR codes. In *Proceedings of the 2018 International Conference on Industrial Technology*, ICIT '18, pages 1664–1669, New York, NY, US, 2018. IEEE. doi: 10.1109/icit.2018.8352431.

Riccardo Focardi, Flaminia L. Luccio, and Heider A. M. Wahsheh. Usable security for qr code. *Journal of Information Security and Applications*, 48(1):102369, 2019. doi: 10.1016/j.jisa.2019.102369.

Jason Ford and Hala Strohmier Berry. Feasibility of machine learning-enhanced detection for QR code images in email-based threats. In *Proceedings of the 2024 Cyber Awareness and Research Symposium*, CARS '24, pages 1–9, New York, NY, US, 2024. IEEE. doi: 10.1109/cars61786.2024.10778732.

Anjuli Franz, Verena Zimmermann, Gregor Albrecht, and Katrin Heatwik. Sok: Still plenty of phish in the sea – a taxonomy. In *Proceedings of the 2021 Symposium on Usable Privacy and Security*, SOUPS '21, page 21334, Berkeley, CA, US, 2021. USENIX Association. doi: 10.5555/3563572.3563590.

Alisa Frik, Juliann Kim, Joshua Rafael Sanchez, and Joanne Ma. Users' Expectations About and Use of Smartphone Privacy and Security Settings. In *Conference on Human Factors in Computing Systems*, CHI '22, pages 1–24, New York, NY, US, 2022. ACM. doi: 10.1145/3491102.3517504.

Michael Froehlich, Philipp Hulm, and Florian Alt. Under pressure. a user-centered threat model for cryptocurrency owners. In *Proceedings of the 2021 Conference on Blockchain Technology and Applications*, BTA '21, pages 39–50, New York, NY, US, 2021. ACM. doi: 10.1145/3510487.3510494.

FTC. Scammers hide harmful links in QR codes to steal your information, 2023. URL `https://consumer.ftc.gov/consumer-alerts/2023/12/scammers-hide-harmful-links-qr-codes-steal-your-information`.

FTC. Report Fraud, 2026. URL https://reportfraud.ftc.gov.

Evan Fuller, Jeffrey M. Rabin, and Guershon Harel. INTELLECTUAL NEED AND PROBLEM-FREE ACTIVITY IN THE MATHEMATICS CLASSROOM. *Jornal Internacional de Estudos em Educação Matemática*, 4(1):35, 2011. doi: 10.17921/2176-5634.2011v4n1p%p.

Arash Ghazvini and Zarina Shukur. A Framework for an Effective Information Security Awareness Program in Healthcare. *International Journal of Advanced Computer Science and Applications*, 8(2):193–205, 2017. doi: 10.14569/ijacsa.2017.080226.

Kate Gibson. Want to help people in Ukraine? Here are ways to donate, 2023. URL https://www.cbsnews.com/news/russia-ukraine-refugees-donations-charity/. Accessed: 2025.05.07.

Google. Avoid & report phishing emails - Gmail Help, 2026. URL https://support.google.com/mail/answer/8253?hl=en&utm_source=chatgpt.com.

Mahesh Gopinath and Sibi Chakkaravarthy Sethuraman. A comprehensive survey on deep learning based malware detection techniques. *Computer Science Review*, 47(1):100529, 2023. doi: 10.1016/j.cosrev.2022.100529.

Thomas D. Gore and Cheryl Campanella Bracken. Testing the Theoretical Design of a Health Risk Message: Reexamining the Major Tenets of the Extended Parallel Process Model. *Health Education & Behavior*, 32(1):27–41, 2005. doi: 10.1177/1090198104266901.

Slade E. Griffin and Casey C. Rackley. Vishing. In *Proceedings of the 5th Annual Conference on Information Security Curriculum Development*, InfoSecCD '08, pages 33–35, New York, NY, US, 2008. ACM. doi: 10.1145/1456625.1456635.

Matthew D Grilli, Katelyn S McVeigh, Ziad M Hakim, Aubrey A Wank, Sarah J Getz, Bonnie E Levin, Natalie C Ebner, and Robert C Wilson. Is This Phishing? Older Age Is Associated With Greater Difficulty Discriminating Between Safe and Malicious Emails. *The Journals of Gerontology Series B: Psychological Sciences and Social Sciences*, 76(9):1711–1715, 2020. doi: 10.1093/geronb/gbaa228.

Yuqing Guan and Andrea Tick. Literature review on security of personal information in electronic payments. In *Proceedings of the 2024 Symposium on Applied Computational Intelligence and Informatics*, SACI '24, pages 533–540, Southampton, UK, 2024. WIT Press. doi: 10.1109/saci60582.2024.10619864.

Dong Guo, Jian Cao, Xiaoqi Wang, Qiang Fu, and Qiang Li. Combating qr-code-based compromised accounts in mobile social networks. *Sensors*, 16(9):1522, 2016. doi: 10.3390/s16091522.

Frank Offei Gyimah, Ernest Ofori-Mensah, Henrietta Boowuo, and Sakhi Aggrawal. Empowering shared mobility vehicle riders, stopping scams: A cyber kill chain and awareness approach to qrishing on college campuses. In *Proceedings of the 2024 Cyber Awareness and Research Symposium*, CARS '24, pages 1–7, New York, NY, US, 2024. IEEE. doi: 10.1109/cars61786.2024.10778789.

Hilda Hadan, Nicolas Serrano, and L Jean Camp. A holistic analysis of web-based public key infrastructure failures: comparing experts' perceptions and real-world incidents. *Journal of Cybersecurity*, 7(1):tyab025, 2021. doi: 10.1093/cybsec/tyab025.

C.B. Haley, R.C. Laney, J.D. Moffett, and B. Nuseibeh. The effect of trust assumptions on the elaboration of security requirements. In *Proceedings. 12th IEEE International Requirements Engineering Conference, 2004.*, pages 102–111, 2004. doi: 10.1109/ICRE.2004.1335668.

Charles B. Haley, Robin C. Laney, Jonathan D. Moffett, and Bashar Nuseibeh. Using trust assumptions with security requirements. *Requirements Engineering*, 11(2):138–151, 2006. doi: 10.1007/s00766-005-0023-4.

Julie M. Haney and Wayne G. Lutters. "It's Scary. . . It's Confusing. . . It's Dull": How Cybersecurity Advocates Overcome Negative Perceptions of Security. In *Symposium on Usable Privacy and Security*, SOUPS 2018, pages 411–425, Berkeley, CA, US, 2018. USENIX Association. URL `https://www.usenix.org/conference/soups2018/presentation/haney-perceptions`.

Ryan Heartfield and George Loukas. Detecting semantic social engineering attacks with the weakest link: Implementation and empirical evaluation of a human-as-a-security-sensor framework. *Computers & Security*, 76:101–127, 2018. doi: 10.1016/j.cose.2018.02.020.

Franziska Herbert, Steffen Becker, Leonie Schaewitz, Jonas Hielscher, Marvin Kowalewski, Angela Sasse, Yasemin Acar, and Markus Dürmuth. A World Full of Privacy and Security (Mis)conceptions? Findings of a Representative Survey in 12 Countries. In *Conference on Human Factors in Computing Systems*, CHI '23, pages 1–23, New York, NY, US, 2023. ACM. doi: 10.1145/3544548.3581410.

Cormac Herley. So Long, and No Thanks for the Externalities: The Rational Rejection of Security Advice by Users. In *Proceedings of the 2009 Workshop on New Security Paradigms Workshop*, NSPW '09, pages 133–144, Oxford, United Kingdom, 2009. ACM. doi: 10.1145/1719030.1719050.

Jonas Hielscher, Annette Kluge, Uta Menges, and M. Angela Sasse. "taking out the trash": Why security behavior change requires intentional forgetting. In *New Security Paradigms Workshop*, NSPW 2021, pages 108–122, New York, NY, US, 2021. ACM. doi: 10.1145/3498891.3498902.

Simon Hill. How to scan a qr code on your smartphone, 2021. URL https://www.wired.com/story/how-to-scan-a-qr-code/.

Grant Ho, Ariana Mirian, Elisa Luo, Khang Tong, Euyhyun Lee, Lin Liu, Christopher A. Longhurst, Christian Dameff, Stefan Savage, and Geoffrey M. Voelker. Understanding the Efficacy of Phishing Training in Practice. In *2025 IEEE Symposium on Security and Privacy (SP)*, pages 37–54, 2025. doi: 10.1109/SP61157.2025.00076.

Nora Hutchinson, Grayson L. Baird, and Megha Garg. Examining the Reading Level of Internet Medical Information for Common Internal Medicine Diagnoses. *The American Journal of Medicine*, 129(6):637–639, 2016. doi: 10.1016/j.amjmed.2016.01.008.

International Organization for Standardization. ISO/IEC 40500:2025, 2025. URL https://www.iso.org/standard/91029.html.

Safwati Ismail, Mohammed Hazim Alkawaz, and Alvin Ebenazer Kumar. Quick response code validation and phishing detection tool. In *Proceedings of the 2021 International Symposium on Computer Applications and Industrial Electronics*, ISCAIE '21, pages 261–266, New York, NY, US, 2021. IEEE. doi: 10.1109/iscaie51753.2021.9431807.

Lennart Jaeger. Information Security Awareness: Literature Review and Integrative Framework. In *Hawaii International Conference on System Sciences*, HICSS 51, pages 4703–4712, Hawaii, US, 2018. AIS. doi: 10.24251/hicss.2018.593.

Jurjen Jansen and Rutger Leukfeldt. How people help fraudsters steal their money: an analysis of 600 online banking fraud cases. In *Proceedings of the 2015 Workshop on Socio-Technical Aspects in Security and Trust*, STAST '15, pages 24–31, New York, NY, US, 2015. IEEE. doi: 10.1109/stast.2015.12.

Jurjen Jansen and Paul Van Schaik. The design and evaluation of a theory-based intervention to promote security behaviour against phishing. *International Journal of Human-Computer Studies*, 123:40–55, 2019. doi: 10.1016/j.ijhcs.2018.10.004.

Rebecca Jeong and Sonia Chiasson. 'Lime', 'Open Lock', and 'Blocked': Children's Perception of Colors, Symbols, and Words in Cybersecurity Warnings. In *Conference on Human Factors in Computing Systems*, CHI '20, pages 1–13, New York, NY, US, 2020. ACM. doi: 10.1145/3313831.3376611.

Xing Jin, Xunchao Hu, Kailiang Ying, Wenliang Du, Heng Yin, and Gautam Nagesh Peri. Code injection attacks on html5-based mobile apps. In *Proceedings of the 2014 Conference on Computer and Communications Security*, CCS '14, pages 66–77, New York, NY, US, 2014. ACM. doi: 10.1145/2660267.2660275.

Mohammed Alhassan Jubur. Comparative technical analysis of qr code and nfc in contactless payments. In *Proceedings of the 2024 International Conference on Computer Technology Applications*, ICCTA '24, pages 242–246, New York, NY, US, 2024. ACM. doi: 10.1145/3674558.3674593.

Hanna Kallio, Anna-Maija Pietilä, Martin Johnson, and Mari Kangasniemi. Systematic methodological review: Developing a framework for a qualitative semi-structured interview guide. *Journal of Advanced Nursing*, 72(12):2954–2965, 2016. doi: 10.1111/jan.13031.

Tai-Wei Kan, Chin-Hung Teng, and Mike Y Chen. QR Code Based Augmented Reality Applications. In *Handbook of Augmented Reality*, pages 339–354. Springer, New York, NY, US, 2011. doi: 10.1007/978-1-4614-0064-6_16.

Leon Kersten, Pavlo Burda, Luca Allodi, and Nicola Zannone. Investigating the effect of phishing believability on phishing reporting. In *European Symposium on Security and Privacy Workshops*, EuroS&PW '22, pages 117–128, New York, NY, US, 2022. IEEE. doi: 10.1109/EuroSPW55150.2022.00018.

Mohammed Khader, Marcel Karam, and Hanna Fares. Cybersecurity Awareness Framework for Academia. *Information*, 12(10):417, 2021. doi: 10.3390/info12100417.

Khando Khando, Shang Gao, Sirajul M. Islam, and Ali Salman. Enhancing employees information security awareness in private and public organisations: A systematic literature review. *Computers & Security*, 106:102267, 2021. doi: 10.1016/j.cose.2021.102267.

Amin Kharraz, Engin Kirda, William Robertson, Davide Balzarotti, and Aurélien Francillon. Optical delusions: A study of malicious QR codes in the wild. In *Proceedings of the 2014 Conference on Dependable Systems and Networks*, DSN '14, pages 192–203, New York, NY, US, 2014. IEEE. doi: 10.1109/dsn.2014.103.

Peter Kieseberg, Manuel Leithner, Martin Mulazzani, Lindsay Munroe, Sebastian Schrittwieser, Mayank Sinha, and Edgar Weippl. QR code security. In *Proceedings of the 2010 Conference on Advances in Mobile Computing and Multimedia*, MoMM '10, pages 430–435, New York, NY, US, 2010. ACM. doi: 10.1145/1971519.1971593.

Peter Kieseberg, Sebastian Schrittwieser, Manuel Leithner, Martin Mulazzani, Edgar Weippl, Lindsay Munroe, and Mayank Sinha. Malicious pixels – using QR codes as attack vector. In *Trustworthy Ubiquitous Computing*, pages 21–38. Atlantis Press, Paris, FR, 2012. doi: 10.2991/978-94-91216-71-8_2.

J. Peter Kincaid, Robert P. Fishburne, R ichard L. Rogers, and Brad S. Chissom. Derivation of New Readability Formulas for Navy Enlisted Personnel. Technical report, Naval Air Station Memphis, 1975.

Barbara Kitchenham. Procedures for Performing Systematic Reviews. Technical report, Keele University, Keele, GB, 2004.

Katharina Krombholz, Peter Frühwirt, Peter Kieseberg, Ioannis Kapsalis, Markus Huber, and Edgar Weippl. QR code security: A survey of attacks and challenges for usable security. In *Human Aspects of Information Security, Privacy, and Trust*, pages 79–90. Springer, Berlin, DE, 2014. doi: 10.1007/978-3-319-07620-1_8.

Katharina Krombholz, Peter Frühwirt, Thomas Rieder, Ioannis Kapsalis, Johanna Ullrich, and Edgar Weippl. QR code security: How secure and usable apps can protect users against malicious QR codes. In *Proceedings of the 2015 Conference on Availability, Reliability and Security*, ARES '15, pages 230–237, New York, NY, US, 2015. IEEE. doi: 10.1109/ares.2015.84.

Katharina Krombholz, Karoline Busse, Matthew Smith, and Emanuel von Zezschwitz. "If HTTPS Were Secure, I Wouldn't Need 2FA"-End User and Administrator Mental Models of HTTPS. In *IEEE Symposium on Security and Privacy*, IEEE S&P '19, pages 246–263, New York, NY, US, 2019. IEEE. doi: 10.1109/sp.2019.00060.

Kerstin Kunz and Ekaterina Lapshinova-Koltunski. English vs. German from a textual perspective: Looking inside chain intersection. *Bergen Language and Linguistics Studies*, 9(1), April 2018. doi: 10.15845/bells.v9i1.1520.

K.T.A.U Lakmal, L.M.C Perera, S.P.K Padmika, S.P.A De Silva, Dinithi Pandithage, and Deemantha Siriwardana. Email armour: A multi-layered email defense solution. In *Proceedings of the 2024 International Conference on Information Technology Research*, ICITR '24, pages 1–6, New York, NY, US, 2024. IEEE. doi: 10.1109/icitr64794.2024.10857760.

J. Richard Landis and Gary G. Koch. The Measurement of Observer Agreement for Categorical Data. *Biometrics*, 33(1):159, 1977. doi: 10.2307/2529310.

Adam Lerner, Alisha Saxena, Kirk Ouimet, Ben Turley, Anthony Vance, Tadayoshi Kohno, and Franziska Roesner. Analyzing the use of quick response codes in the wild. In *Proceedings of the 2015 Conference on Mobile Systems, Applications, and Services*, MobiSys '15, pages 359–374, New York, NY, US, 2015. ACM. doi: 10.1145/2742647.2742650.

H Leventhal. Fear appeals and persuasion: The differentiation of a motivational construct. *American Journal of Public Health*, 61(6):1208–1224, 1971. doi: 10.2105/AJPH.61.6.1208.

Tie Li, Gang Kou, and Yi Peng. Improving malicious urls detection via feature engineering: Linear and nonlinear space transformation methods. *Information Systems*, 91(1):101494, 2020. doi: 10.1016/j.is.2020.101494.

Tian Lin, Daniel E. Capecci, Donovan M. Ellis, Harold A. Rocha, Sandeep Dommaraju, Daniela S. Oliveira, and Natalie C. Ebner. Susceptibility to spear-phishing emails. *ACM Transactions on Computer-Human Interaction*, 26(5):1–28, 2019. doi: 10.1145/3336141.

Wei-Han Lin, Guan-Yan Yang, and Kuo-Hui Yeh. Integrating fido authentication with new digital identity in taiwan. In *Proceedings of the 2022 Global Conference on Consumer Electronics*, GCCE '22, pages 311–312, New York, NY, US, 2022. IEEE. doi: 10.1109/gcce56475.2022.10014031.

Enze Liu, George Kappos, Eric Mugnier, Luca Invernizzi, Stefan Savage, David Tao, Kurt Thomas, Geoffrey M. Voelker, and Sarah Meiklejohn. Give and take: An end-to-end investigation of giveaway scam conversion rates. In *Proceedings of the 2024 Internet Measurement Conference*, IMC '24, pages 704–712, New York, NY, US, 2024. ACM. doi: 10.1145/3646547.3689005.

Neelanjan Manna. Encrypted message traversal using QR codes. *International Journal of Research Publication and Reviews*, 3(10):1073–1075, 2022. URL `https://ijrpr.com/uploads/V3ISSUE10/IJRPR7472.pdf`.

Ioana Andreea Marin, Pavlo Burda, Nicola Zannone, and Luca Allodi. The Influence of Human Factors on the Intention to Report Phishing Emails. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, CHI '23, pages 1–18, New York, NY, US, 2023. ACM. doi: 10.1145/3544548.3580985.

Market.us. Amazon Statistics and Facts, 2025. URL `https://market.us/statistics/e-commerce-websites/amazon/`. Accessed on 29.06.2026.

Peter Mayer. *Secure and Usable User Authentication*. PhD thesis, Karlsruhe Institute of Technology, 2019.

Peter Mayer and Melanie Volkamer. Addressing Misconceptions About Password Security Effectively. In *Workshop on Socio-Technical Aspects in Security and Trust*, STAST '17, pages 16–27, New York, NY, US, 2018. ACM. doi: 10.1145/3167996.3167998.

Shiona McCallum. Deplorable scam emails fake fundraising for ukraine, 2022. URL `https://www.bbc.com/news/technology-60836962`. Accessed: 2025.05.07.

Richard McPherson, Suman Jana, and Vitaly Shmatikov. No escape from reality: Security and privacy of augmented reality browsers. In *Proceedings of the 2015 Conference on World Wide Web*, WWW '15, pages 743–753, Geneva, CH, 2015. International World Wide Web Conferences Steering Committee. doi: 10.1145/2736277.2741657.

Meik Michalke, Earl Brown, Alberto Mirisola, Alexandre Brulet, and Laura Hauser. koRpus: Text Analysis with Emphasis on POS Tagging, Readability, and Lexical Diversity, May 2021.

Microsoft. What is the mitre att&ck framework? | microsoft security, 2025. URL https://www.microsoft.com/en-us/security/business/security-101/what-is-mitre-attack-framework.

Microsoft. Advanced Outlook.com security for Microsoft 365 subscribers, 2026. URL https://support.microsoft.com/en-us/outlook/advanced-outlook-com-security-for-microsoft-365-subscribers.

Mimecast. Human Risk Management & Advanced Email Security, 2026. URL https://www.mimecast.com/.

Sandhya Mishra and Devpriya Soni. SMS Phishing and Mitigation Approaches. In *2019 Twelfth International Conference on Contemporary Computing (IC3)*, pages 1–5, 2019. doi: 10.1109/IC3.2019.8844920.

MITRE. Adversary-in-the-Middle, Technique T1557, 2026a. URL https://attack.mitre.org/techniques/T1557/.

MITRE. Audio Capture, Technique T1429, 2026b. URL https://attack.mitre.org/techniques/T1429/.

MITRE. Command and Scripting Interpreter, Technique T1623, 2026c. URL https://attack.mitre.org/versions/v19/techniques/T1623/.

MITRE. Command and Scripting Interpreter: Unix Shell, Sub-technique T1623.001, 2026d. URL https://attack.mitre.org/versions/v19/techniques/T1623/001/.

MITRE. Compromise Client Software Binary, Technique T1645, 2026e. URL https://attack.mitre.org/versions/v19/techniques/T1645/.

MITRE. Drive-By Compromise, Technique T1456, 2026f. URL https://attack.mitre.org/versions/v19/techniques/T1456/.

MITRE. Encrypted Channel: SSL Pinning, Sub-technique T1521.003, 2026g. URL https://attack.mitre.org/versions/v19/techniques/T1521/003/.

MITRE. FAQ | MITRE ATT&CK®, 2026h. URL https://attack.mitre.org/resources/faq/.

MITRE. Indicator Removal on Host, Technique T1630, 2026i. URL https://attack.mitre.org/versions/v19/techniques/T1630/.

MITRE. Input Capture, Technique T1417, 2026j. URL https://attack.mitre.org/versions/v19/techniques/T1417/.

MITRE. Input Capture: GUI Input Capture, Sub-technique T1417.002, 2026k. URL https://attack.mitre.org/versions/v19/techniques/T1417/002/.

MITRE. Input Capture: Keylogging, Sub-technique T1417.001, 2026l. URL https://attack.mitre.org/versions/v19/techniques/T1417/001/.

MITRE. Lateral Movement, Tactic TA0008, 2026m. URL https://attack.mitre.org/tactics/TA0008/.

MITRE. Linked Devices, Technique T1676, 2026n. URL https://attack.mitre.org/versions/v19/techniques/T1676/.

MITRE. Masquerading, Technique T1655, 2026o. URL https://attack.mitre.org/versions/v19/techniques/T1655/.

MITRE. Masquerading: Match Legitimate Name or Location, Sub-technique T1655.001, 2026p. URL https://attack.mitre.org/versions/v19/techniques/T1655/001/.

MITRE. Phishing, Technique T1660, 2026q. URL https://attack.mitre.org/versions/v19/techniques/T1660/.

MITRE. Protected User Data, Technique T1636, 2026r. URL https://attack.mitre.org/versions/v19/techniques/T1636/.

MITRE. Protected User Data: Accounts, Sub-technique T1636.005, 2026s. URL https://attack.mitre.org/versions/v19/techniques/T1636/005/.

MITRE. Protected User Data: Calendar Entries, Sub-technique T1636.001, 2026t. URL https://attack.mitre.org/versions/v19/techniques/T1636/001/.

MITRE. Protected User Data: Call Log, Sub-technique T1636.002, 2026u. URL https://attack.mitre.org/versions/v19/techniques/T1636/002/.

MITRE. Protected User Data: Contact List, Sub-technique T1636.003, 2026v. URL https://attack.mitre.org/versions/v19/techniques/T1636/003/.

MITRE. Protected User Data: SMS Messages, Sub-technique T1636.004, 2026w. URL https://attack.mitre.org/versions/v19/techniques/T1636/004/.

MITRE. Subvert Trust Controls, Technique T1632, 2026x. URL https://attack.mitre.org/techniques/T1632/.

MITRE. Subvert Trust Controls: Code Signing Policy Modification, Sub-technique T1632.001, 2026y. URL https://attack.mitre.org/techniques/T1632/001/.

MITRE. Version History, 2026z. URL https://attack.mitre.org/resources/versions/.

Tyler Moore and Benjamin Edelman. Measuring the Perpetrators and Funders of Typosquatting. In *Financial Cryptography and Data Security*, pages 175–191, Berlin, Heidelberg, 2010. Springer. doi: 10.1007/978-3-642-14577-3_15.

Francois Mouton, Louise Leenen, and H. S. Venter. Social engineering attack examples, templates and scenarios. *Computers & Security*, 59(1):186–209, 2016. doi: 10.1016/j.cose.2016.03.004.

Sean Murphy. Laravel QR code generator infected with malware, 2021. URL https://www.kernelmode.blog/laravel-qr-code-generator-infected-with-malware/. 2026.06.22.

Hossen Mustafa, Wenyuan Xu, Ahmad Reza Sadeghi, and Steffen Schulz. You Can Call but You Can't Hide: Detecting Caller ID Spoofing Attacks. In *Conference on Dependable Systems and Networks*, IEEE DNS '14, pages 168–179, London, United Kingdom, 2014. IEEE. doi: 10.1109/DSN.2014.102.

Lorenzo C. Neil, Charlotte Healy, and Julie Haney. "A five-year-old could understand it" versus "This is way too confusing": Exploring Non-expert Understandings and Perceptions of Cybersecurity Definitions. In *Proceedings of the 2025 CHI Conference on Human Factors in Computing Systems*, pages 1–19, Yokohama Japan, 2025. ACM. doi: 10.1145/3706598.3713820.

NIST. Ransomware Protection and Response | CSRC | CSRC, 2021. URL https://csrc.nist.gov/projects/ransomware-protection-and-response.

NIST. Phishing - Glossary | CSRC, 2026a. URL https://csrc.nist.gov/glossary/term/phishing.

NIST. SPAM - Glossary | CSRC, 2026b. URL https://csrc.nist.gov/glossary/term/spam.

Xinyu Niu, Jiandao Zhao, and Bo Tian. The security threat and precautionary measures of qr code of internet of things technology. *Advances in Engineering Technology Research*, 11(1): 775, 2024. doi: 10.56028/aetr.11.1.775.2024.

Norton. What is Pegasus spyware, and how to detect and remove it, 2025. URL https://us.norton.com/blog/emerging-threats/pegasus-spyware.

Mark Nottingham. URI Design and Ownership. Request for Comments RFC 8820, Internet Engineering Task Force, 2020.

James Nurthen, Michael Cooper, and Shawn Lawton Henry. WAI-ARIA Overview, 2025. URL https://www.w3.org/WAI/standards-guidelines/aria/.

Office of Cybersecurity. Beware of ukraine-themed phishing scams. https://web.archive.org/web/20230603072020/https://it.wisc.edu/scams/beware-of-ukraine-themed-phishing-scams//, 2022. No longer available; Only version retrievable from the Wayback Machine.

Tetsuya Okuda, Naoko Chiba, Mitsuaki Akiyama, Toshinori Fukunaga, Ryohei Suzuki, and Masayuki Kanda. Brand Validation: Security Indicator to Better Indicate Website Identity. In *HCI for Cybersecurity, Privacy and Trust*, HCI-CPT '21, pages 432–447, 2021. doi: 10.1007/978-3-030-77392-2_28.

Bridget Opazo, Don Whitteker, and Chen-Chi Shing. Email trouble: Secrets of spoofing, the dangers of social engineering, and how we can help. In *Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, ICNC-FSKD '17, pages 2812–2817, 2017. doi: 10.1109/FSKD.2017.8393226.

Jillian A. Tullis Owen, C. Ellis, and E. Goffman. Naturalistic Inquiry. In *The SAGE Encyclopedia of Qualitative Research Methods*. SAGE Publications, Inc., 2008. doi: 10.4135/9781412963909.

Matthew J Page, Joanne E McKenzie, Patrick M Bossuyt, Isabelle Boutron, Tammy C Hoffmann, Cynthia D Mulrow, Larissa Shamseer, Jennifer M Tetzlaff, Elie A Akl, Sue E Brennan, Roger Chou, Julie Glanville, Jeremy M Grimshaw, Asbjørn Hróbjartsson, Manoj M Lalu, Tianjing Li, Elizabeth W Loder, Evan Mayo-Wilson, Steve McDonald, Luke A McGuinness, Lesley A Stewart, James Thomas, Andrea C Tricco, Vivian A Welch, Penny Whiting, and David Moher. The prisma 2020 statement: an updated guideline for reporting systematic reviews. *BMJ*, 372 (1):n71, 2021. doi: 10.1136/bmj.n71.

Atharva Pawar, Chirag Fatnani, Rajani Sonavane, Riya Waghmare, and Sarang Saoji. Secure qr code scanner to detect malicious url using machine learning. In *Proceedings of the 2022 Asian Conference on Innovation in Technology*, ASIANCON '22, pages 1–8, New York, NY, US, 2022. IEEE. doi: 10.1109/asiancon55314.2022.9908759.

PayPal. Venmo for Business: Drive Social Commerce Growth in 2025, 2025. URL https://www.paypal.com/us/brc/article/get-growing-with-social-payments. Accessed on 29.06.2026.

Pennian Bank. Quishing - the newest threat to your financial security, 2025. URL https://pennian.bank/blog/watch-out-for-quishing-the-newest-threat-to-your-financial-security/.

Krassie Petrova, Adriana Romaniello, B Dawn Medlin, and Sandra A Vannoy. QR Codes Advantages and Dangers. In *Conference on E-Business and Telecommunications*, ICETE '16, pages 112–115, Online, 2016. SciTePress. doi: 10.5220/0005993101120115.

Pizza Hut. Pizza Hut Near Me - Store Locations | Pizza Hut, 2026. URL https://locations.pizzahut.com/. Accessed on 29.06.2026.

Emilee Rader and Anjali Munasinghe. "Wait, Do I Know This Person?": Understanding Misdirected Email. In *Conference on Human Factors in Computing Systems*, CHI '19, pages 1–13, New York, NY, US, 2019. ACM. doi: 10.1145/3290605.3300520.

Ahmad Sahban Rafsanjani, Norshaliza Binti Kamaruddin, Hazlifah Mohd Rusli, and Mohammad Dabbagh. Qsecr: Secure qr code scanner according to a novel malicious url detection framework. *IEEE Access*, 11(1):92523–92539, 2023. doi: 10.1109/access.2023.3291811.

Elissa M. Redmiles, Sean Kross, and Michelle L. Mazurek. How I Learned to be Secure: a Census-Representative Survey of Security Advice Sources and Behavior. In *Conference on Computer and Communications Security*, CCS 2016, page 666–677, New York, NY, US, 2016. ACM. doi: 10.1145/2976749.2978307.

Elissa M Redmiles, Noel Warford, Amritha Jayanti, and Aravind Koneru. A Comprehensive Quality Evaluation of Security and Privacy Advice on the Web. In *USENIX Security Symposium*, USENIX Security '20, pages 89–108, Berkeley, CA, US, 2020. USENIX.

Robert W. Reeder, Adrienne Porter Felt, Sunny Consolvo, Nathan Malkin, Christopher Thompson, and Serge Egelman. An Experience Sampling Study of User Reactions to Browser Warnings in the Field. In *Conference on Human Factors in Computing Systems*, CHI '18, pages 1–13, New York, NY, US, 2018. ACM. doi: 10.1145/3173574.3174086.

Koceilah Rekouche. Early Phishing. *arXiv:1106.4692 [cs]*, 2011.

Karen Renaud and Marc Dupuis. Cyber security fear appeals: Unexpectedly complicated. In *Proceedings of the New Security Paradigms Workshop*, pages 42–56, San Carlos Costa Rica, 2019. ACM. doi: 10.1145/3368860.3368864.

Report Fraud. UK's Home for Reporting Cyber Crime & Fraud - Report Fraud, 2026. URL https://www.reportfraud.police.uk/.

Joshua Reynolds, Deepak Kumar, Zane Ma, Rohan Subramanian, Meishan Wu, Martin Shelton, Joshua Mason, Emily Stark, and Michael Bailey. Measuring identity confusion with uniform resource locators. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, CHI '20, pages 1–12, New York, NY, US, 2020. ACM. doi: 10.1145/3313831.3376298.

E. Ricson. 80+ qr code statistics & trends 2026 full report, 2026. URL https://www.qrcode-tiger.com/qr-code-statistics-2022-q1.

German Rodriguez, Jenny Torres, Pamela Flores, Eduardo Benavides, and Daniel Nuéz Agurto. Xsstudent: Proposal to avoid cross-site scripting (xss) attacks in universities. In *Proceedings of the 2019 Cyber Security in Networking Conference*, CSNet '19, pages 142–149, New York, NY, US, 2019. IEEE. doi: 10.1109/csnet47905.2019.9108965.

Ronald W. Rogers. A Protection Motivation Theory of Fear Appeals and Attitude Change. *The Journal of Psychology*, 91(1):93–114, 1975. doi: 10.1080/00223980.1975.9915803.

Christoph Rottermanner, Peter Kieseberg, Markus Huber, Martin Schmiedecker, and Sebastian Schrittwieser. Privacy and data protection in smartphone messengers. In *Proceedings of the 2015 Conference on Information Integration and Web-Based Applications Services*, iiWAS '15, pages 1–10, New York, NY, US, 2015. ACM. doi: 10.1145/2837185.2837202.

Sayak Saha Roy, Poojitha Thota, Krishna Vamsi Naragam, and Shirin Nilizadeh. From chatbots to phishbots?: Phishing scam generation in commercial large language models. In *Proceedings of the 2024 IEEE Symposium on Security and Privacy*, SP '24, pages 36–54, New York, NY, US, 2024. IEEE. doi: 10.1109/sp54263.2024.00182.

Scott Ruoti, Tyler Monson, Justin Wu, Daniel Zappala, and Kent Seamons. Weighing Context and Trade-offs: How Suburban Adults Selected Their Online Security Posture. In *Symposium on Usable Privacy and Security*, SOUPS '17, Berkeley, CA, US, 2017. USENIX. doi: 10.5555/3235924.3235942. URL `https://www.usenix.org/system/files/conference/soups2017/soups2017-ruoti.pdf`.

Dale Rutherford and Ningning Wu. Cybersecurity risks in the deployment and use of digital business cards: Implications for organizations and end-users. In *Proceedings of the 2023 Conference on Computational Science and Computational Intelligence*, CSCI '23, pages 765–770, New York, NY, US, 2023. IEEE. doi: 10.1109/csci62032.2023.00130.

Nur Sahira Aziyan Mohd Sabri, Noorhayati Mohamed Noor, and Zolidah Kasiran. Secured qr scanner (sqr) based on query method. In *Proceedings of the 2023 International Conference on Recent Advances and Innovations in Engineering*, ICRAIE '23, pages 1–6, Singapore, SG, 2023. Springer. doi: 10.1109/ICRAIE57573.2023.10012345.

Johnny Saldaña. *The coding manual for qualitative researchers*. SAGE, Los Angeles, CA, US, 2013.

Gargi Sarkar and Sandeep K. Shukla. Behavioral analysis of cybercrime: Paving the way for effective policing strategies. *Journal of Economic Criminology*, 2(1):100034, 2023. doi: 10.1016/j.jeconc.2023.100034.

Olivia Sarker, Asangi Jayatilaka, Shrif Haggag, Chelsea Liu, and M. ALi babar. A Multi-Vocal Literature Review on Challenges and Critical Success Factors of Phishing Education, Training and Awareness. *Journal of Systems and Software*, 208(111899):1–25, 2024. doi: 10.1016/j.jss.2023.111899.

Yerkezhan Sartayeva and Henry C. B. Chan. A survey on indoor positioning security and privacy. *Computers & Security*, 131(1):103293, 2023. doi: 10.1016/j.cose.2023.103293.

Riccardo Scandariato, Kim Wuyts, and Wouter Joosen. A descriptive study of Microsoft's threat modeling technique. *Requirements Engineering*, 20(2):163–180, 2015. doi: 10.1007/s00766-013-0195-2.

Lorin Schöni, Victor Carles, Martin Strohmeier, Peter Mayer, and Verena Zimmermann. You know what? - evaluation of a personalised phishing training based on users' phishing knowledge and detection skills. In *European Symposium on Usable Security*, EuroUSEC 2024, pages 1–14, New York, NY, US, 2024. ACM. doi: 10.1145/3688459.3688460.

Kent Seamons. *Privacy-Enhancing Technologies*, pages 149–170. Springer, Cham, 2022. doi: 10.1007/978-3-030-82786-1_8.

Jan Seeburger. No cure for curiosity: Linking physical and digital urban layers. In *Proceedings of the 2012 Nordic Conference on Human-Computer Interaction: Making Sense Through Design*, NordiCHI '12, pages 247–256, New York, NY, US, 2012. ACM. doi: 10.1145/2399016.2399054.

Brooke Seipel. What Is Quishing? How QR Code Scams Work and How to Avoid Them, 2026. URL `https://www.mcafee.com/blogs/tips-tricks/what-is-quishing-how-qr-code-scams-work-and-how-to-avoid-them/`. Accessed on 3.7.2026.

Filipo Sharevski, Amy Devine, Emma Pieroni, and Peter Jachim. Phishing with Malicious QR Codes. In *European Symposium on Usable Security*, EuroUSEC '22, pages 160–171, New York, NY, US, 2022. ACM. doi: 10.1145/3549015.3554172.

Mahmood Sharif, Kevin A. Roundy, Matteo Dell'Amico, Christopher Gates, Daniel Kats, Lujo Bauer, and Nicolas Christin. A Field Study of Computer-Security Perceptions Using Anti-Virus Customer-Support Chats. In *Conference on Human Factors in Computing Systems*, CHI '19, pages 1–12, New York, NY, US, 2019. ACM. doi: 10.1145/3290605.3300308.

Vishrut Sharma. A study of malicious qr codes. *International Journal of Computational Intelligence and Information Security*, 3(5):1–6, 2012. URL `https://www.academia.edu/1864785/A_Study_of_Malicious_QR_Codes`.

Richard Shay, Iulia Ion, Robert W. Reeder, and Sunny Consolvo. "My religious aunt asked why i was trying to sell her viagra": experiences with account hijacking. In *Conference on Human Factors in Computing Systems*, CHI '14, pages 2657–2666, New York, NY, US, 2014. ACM. doi: 10.1145/2556288.2557330.

Shu Shen, Zhao-Qing Wei, Li-Juan Sun, Yang-Qing Su, Ru-Chuan Wang, and Han-Ming Jiang. The shared bicycle and its network—internet of shared bicycle (iosb): A review and survey. *Sensors*, 18(8):2581, 2018. doi: 10.3390/s18082581.

Dong-Hee Shin, Jaemin Jung, and Byeng-Hee Chang. The psychology behind qr codes: User experience perspective. *Computers in Human Behavior*, 28(4):1417–1426, 2012.

Cathrine Shoichet. Scammers are targeting sponsors who are trying to help ukrainians reach the us. https://edition.cnn.com/2022/08/03/us/uniting-for-ukraine-phishing-scam-cec/index.html, 2022. Accessed: 2025.05.07.

Nadiia Shulzhenko and Snizhana Romashkin. Internet fraud and transnational organized crime. *Juridical Tribune*, 10(1):162–172, 2020. URL `https://www.tribunajuridica.eu/arhiva/An10v1/11.%20Romashkin_Shulzhenko.pdf`.

Jun Song, Kun Gao, Xinyang Shen, Xiaotian Qi, Rui Liu, and Kim-Kwang Raymond Choo. Qrfence: A flexible and scalable qr link security detection framework for android devices. *Future Generation Computer Systems*, 88(1):663–674, 2018. doi: 10.1016/j.future.2018.05.082.

spamloco. Demo Dirty use of USSD Codes in Cellular Network en Ekoparty 2012, 2012. URL `http://www.youtube.com/watch?v=Q2-0B04HPhs`. Accessed last on 27.05.2026.

S S Sravan, Susmita Mandal, P J A Alphonse, and P L Ramesh. A partial offline payment system for connecting the unconnected using internet of things: A survey. *Comput. Surveys*, 57(2):1–35, 2024. doi: 10.1145/3687132.

Brian Stanton, Mary F. Theofanos, Sandra Spickard Prettyman, and SUSnne Furman. Security Fatigue. *IT Professional*, 18(5):26–32, 2016. doi: 10.1109/MITP.2016.84.

Statista. Netflix: Number of subscribers worldwide 2025, 2025. URL `https://www.statista.com/statistics/250934/quarterly-number-of-netflix-streaming-subscribers-worldwide/`. Accessed on 29.06.2026.

Ashley Stewart. Microsoft lost billions of dollars on hololens, and its huge ivas military contract is in trouble, 2024. URL `https://africa.businessinsider.com/news/microsoft-lost-billions-of-dollars-on-hololens-and-its-huge-ivas-military-contract-is/symf9vy`.

Sikiru Subairu, John Alhassan, Shafii Abdulhamid, and Joseph Ojeniyi. A review of detection methodologies for quick response code phishing attacks. In *Proceedings of the 2020 International Conference on Computer and Information Sciences*, ICCIS '20, pages 1–5, New York, NY, US, 2020. IEEE. doi: 10.1109/iccis49240.2020.9257687.

A. Subash, C. S. Arvin Danny, and M. Vijayalakshmi. IoT-based secure luggage storage kiosk. *International Journal of Information Technology*, 15(4):1911–1918, 2023. doi: 10.1007/s41870-023-01229-3.

Naeem Firdous Syed, Syed W Shah, Rolando Trujillo-Rasua, and Robin Doss. Traceability in supply chains: A cyber security analysis. *Computers & Security*, 112(1):102536, 2022. doi: 10.1016/j.cose.2021.102536.

Divyanshu Thakur, Srijan Sah, Priyansh Ailsinghani, and Virender Ranga. A comparative study of machine learning models for network traffic classification in the iot landscape. In *Proceedings of the 2024 International Conference on Applied Artificial Intelligence and Computing*, ICAAIC '24, pages 670–676, New York, NY, US, 2024. IEEE. doi: 10.1109/icaaic60222.2024.10575268.

David R. Thomas. A General Inductive Approach for Analyzing Qualitative Evaluation Data. *American Journal of Evaluation*, 27(2):237–246, 2006. doi: 10.1177/1098214005283748.

SUSnne Trauzettel-Klosinski, Klaus Dietz, and the IReST Study Group. Standardized Assessment of Reading Performance: The New International Reading Speed Texts IReST. *Investigative Ophthalmology & Visual Science*, 53(9):5452–5461, August 2012. doi: 10.1167/iovs.11-8284.

Yu-Ju Tu, Wei Zhou, and Selwyn Piramuthu. Critical risk considerations in auto-id security: Barcode vs. rfid. *Decision Support Systems*, 142(1):113471, 2021. doi: 10.1016/j.dss.2020.113471.

UNICEF. Unicef in ukraine. https://www.unicefUS.org/what-unicef-does/where-unicef-works/europe/ukraine?gad_source=1&gclid=CjwKCAiA0syqBhBxEiwAeNx9N25fr-Em7DZW7AfT30GsVAUYi4H8xG7gNhMXoXAtBXHz58eWhzA11hoCm3wQAvD_BwE, 2026. Accessed on 08.05.2026.

U.S. Access Board. U.S. Access Board - Revised 508 Standards and 255 Guidelines, 2018. URL `https://www.access-board.gov/ict/`.

US Environmental Protection Agency. Readability for Developing and Pretesting Concepts, Messages, Materials, and Activities, 2020. URL `https://www.epa.gov/choose-fish-and-shellfish-wisely/readability-developing-and-pretesting-concepts-messages-materials`.

Dimitri Van Landuyt and Wouter Joosen. A descriptive study of assumptions in STRIDE security threat modeling. *Software and Systems Modeling*, 21(6):2311–2328, December 2022. doi: 10.1007/s10270-021-00941-7.

Gaurav Varshney, Manoj Misra, and Pradeep Atrey. Secure authentication scheme to thwart rt mitm, cr mitm and malicious browser extension based phishing attacks. *Journal of Information Security and Applications*, 42(1):1–17, 2018. doi: 10.1016/j.jisa.2018.07.001.

Gaurav Varshney, Padmavathi Iyer, Swati Goel, and Tarun Kumar Singh. Upiρ: An envisioned policy-based upi architecture for secure transactions. In *Proceedings of the 2024 Conference on Mobile Ad-Hoc and Smart Systems*, MASS '24, pages 658–663, New York, NY, US, 2024a. IEEE. doi: 10.1109/mass62177.2024.00105.

Gaurav Varshney, Rahul Kumawat, Vijay Varadharajan, Uday Tupakula, and Chandranshu Gupta. Anti-phishing: A comprehensive perspective. *Expert Systems with Applications*, 238(1): 122199, 2024b. doi: 10.1016/j.eswa.2023.122199.

Reddy B Varun and S Rashmi. Prevalent cyber attacks and defense. In *Proceedings of the 2023 International Conference on Integrated Circuits and Communication Systems*, ICICACS '23, pages 1–6, New York, NY, US, 2023. IEEE. doi: 10.1109/icicacs57338.2023.10099516.

Maxime Veit, Oliver Wiese, Fabian Lucas Ballreich, Melanie Volkamer, Douglas Engels, and Peter Mayer. Sok: The past decade of user deception in emails and today's email clients' susceptibility to phishing techniques. *Computers & Security*, 150(1):104197, 2024. doi: 10.1016/j.cose.2024.104197.

Timothy Vidas, Emmanuel Owusu, Shuai Wang, Cheng Zeng, Lorrie Faith Cranor, and Nicolas Christin. Qrishing: The susceptibility of smartphoneusers to qr code phishing attacks. In *Proceedings of the 2013 Confernece on Financial Cryptography and Data Security*, volume 7862 of *FC '13*, pages 52–69, Cham, CH, 2013. Springer. doi: 10.1007/978-3-642-41320-9_4.

Melanie Volkamer, Karen Renaud, and Paul Gerber. Spot the phish by checking the pruned url. *Information and Computer Security*, 24(4):372–385, 2016. doi: 10.1108/ICS-07-2015-0032.

Heider A. M. Wahsheh and Mohammed S. Al-Zahrani. Secure Real-Time Computational Intelligence System Against Malicious QR Code Links. *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS & CONTROL*, 16(3), 2021.

Heider AM Wahsheh and Flaminia L Luccio. Security and privacy of qr code applications: A comprehensive study, general guidelines and solutions. *Information*, 11(4):217, 2020. doi: 10.3390/info11040217.

Paul Keng Fai Wan and Shanshan Jiang. Enabling a dynamic information flow in digital product passports during product use phase: A literature review and proposed framework. *Sustainable Production and Consumption*, 54(1):362–374, 2025. doi: 10.1016/j.spc.2025.01.014.

Yifei Wang. A survey of phishing detection: from an intelligent countermeasures view. In *Proceedings of the 2022 Conference on Telecommunications, Optics and Computer Science*, TOCS '22, pages 761–769, New York, NY, US, 2022. IEEE. doi: 10.1109/tocs56154.2022.10016193.

Yunjia Wang and Ishbel Duncan. A novel method to prevent phishing by using ocr technology. In *Proceedings of the 2019 Conference on Cyber Security and Protection of Digital Services*, Cyber Security '19, pages 1–5, New York, NY, US, 2019. IEEE Computer Society. doi: 10.1109/cybersecpods.2019.8885101.

Kim Witte. Putting the fear back into fear appeals: The extended parallel process model. *Communication Monographs*, 59(4):329–349, 1992. doi: 10.1080/03637759209376276.

World Wide Web Consortium. Web Standards, 2026. URL `https://www.w3.org/standards/`.

Quan Xiao. Understanding the asymmetric perceptions of smartphone security from security feature perspective: A comparative study. *Telematics and Informatics*, 58(1):101535, 2021. doi: 10.1016/j.tele.2020.101535.

Huiping Yao and Dongwan Shin. Towards preventing qr code based attacks on android phone using security warnings. In *Proceedings of the 2013 Asian Symposium on Information, Computer and Communications Security*, ASIACCS '13, pages 341–346, New York, NY, US, 2013. ACM. doi: 10.1145/2484313.2484357.

Kelvin S. C. Yong, Kang Leng Chiew, and Choon Lin Tan. A survey of the QR code phishing: The current attacks and countermeasures. In *Conference on Smart Computing & Communications*, ICSCC '19, pages 1–5, New York, NY, US, 2019. IEEE. doi: 10.1109/icscc.2019.8843688.

Amri Yusoff. *A Conceptual Framework for Serious Games and Its Validation*. PhD thesis, University of Southampton, 2010.

Mohd Imran Md Yusop, Nazhatul Hafizah Kamarudin, Nur Hanis Sabrina Suhaimi, and Mohammad Kamrul Hasan. Advancing passwordless authentication: A systematic review of methods, challenges, and future directions for secure user identity. *IEEE Access*, 13(1):13919–13943, 2025. doi: 10.1109/access.2025.3528960.

Xiao Zhang, Griffin Klevering, Xinyu Lei, Yiwen Hu, Li Xiao, and Guan-Hua Tu. The security in optical wireless communication: A survey. *ACM Computing Surveys*, 55(14s):1–36, 2023. doi: 10.1145/3594718.

Rui Zhao. Campass: a secure camera-based password manager for kiosk browsing. In *Proceedings of the 2023 Conference on Trust, Security and Privacy in Computing and Communications*, TrustCom '23, pages 1580–1585, New York, NY, US, 2023. IEEE. doi: 10.1109/trustcom 60117.2023.00215.

Sarah Zheng and Ingolf Becker. Presenting suspicious details in user-facing e-mail headers does not improve phishing detection. In *Proceedings of the 2022 Symposium on Usable Privacy and Security*, SOUPS '22, pages 1–12, Berkeley, CA, US, 2022. USENIX Association. URL `https://www.usenix.org/conference/soups2022/presentation/zheng`.

Anfu Zhou, Guangyuan Su, Shilin Zhu, and HuaDong Ma. Invisible qr code hijacking using smart led. *ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 3(3):1–23, 2019. doi: 10.1145/3351284.

Yixin Zou, Kevin Roundy, Acar Tamersoy, Saurabh Shintre, Johann Roturier, and Florian Schaub. Examining the Adoption and Abandonment of Security, Privacy, and Identity Theft Protection Practices. In *Conference on Human Factors in Computing Systems*, CHI '20, pages 1–15, New York, NY, US, 2020. ACM. doi: 10.1145/3313831.3376570.

Zakiah Zulkefli, Manmeet Mahinderjit Singh, Azizul Rahman Mohd Shariff, and Azman Samsudin. Typosquat Cyber Crime Attack Detection via Smartphone. *Procedia Computer Science*, 124:664–671, 2017. doi: 10.1016/j.procs.2017.12.203.