

# A Survey on Multi-Factor Authentication for Online Banking in the Wild

Federico Sinigaglia<sup>a,b</sup>, Roberto Carbone<sup>b</sup>, Gabriele Costa<sup>c</sup>, Nicola Zannone<sup>d</sup>

<sup>a</sup>*DIBRIS, Università degli Studi di Genova, Genova, Italy*

<sup>b</sup>*Security & Trust Research Unit, Fondazione Bruno Kessler, Trento, Italy*

<sup>c</sup>*SysMA Unit, IMT School for Advanced Studies, Lucca, Italy*

<sup>d</sup>*Eindhoven University of Technology, Eindhoven, Netherlands*

---

## Abstract

In recent years, the usage of online banking services has considerably increased. To protect the sensitive resources managed by these services against attackers, banks have started adopting Multi-Factor Authentication (MFA). To date, a variety of MFA solutions have been implemented by banks, leveraging different designs and features and providing a non-homogeneous level of security and user experience. Public and private authorities have defined laws and guidelines to guide the design of more secure and usable MFA solutions, but their influence on existing MFA implementations remains unclear. In this work, we present a latitudinal study on the adoption of MFA and the design choices made by banks operating in different countries. In particular, we evaluate the MFA solutions currently adopted in the banking sector in terms of *(i)* compliance with laws and best practices, *(ii)* robustness against attacks and *(iii)* complexity. We also investigate possible correlations between these criteria. Based on this study, we identify a number of lessons learned and open challenges.

**Keywords:** Multi-Factor Authentication, Online Banking, Mobile Banking, Remote Payments, Legal Compliance, Threat Models, Field Study

---

---

*Email addresses:* federico.sinigaglia.ge@gmail.com (Federico Sinigaglia), carbone@fbk.eu (Roberto Carbone), gabriele.costa@imtlucca.it (Gabriele Costa), n.zannone@tue.nl (Nicola Zannone)

## 1. Introduction

Over the last decade, the shift towards online business has gained momentum. A sector in which online services are becoming predominant is the banking sector, where most banks have started offering their services online. Online banking services allow customers to remotely access their bank accounts and financial data as well as to perform online payments and other financial transactions. These services are becoming increasingly popular among customers. According to Eurostat [1], the number of European citizens using online banking services has doubled since 2007 and currently more than half of the European population use an online banking service daily.

Although online banking services provides evident benefits to both banks and customers, they introduce new security and privacy issues. Resources managed by online banking services are sensitive and, thus, they should be properly protected against theft and other attacks. A fundamental security measure for the protection of online resources is the employment of reliable (digital) authentication mechanisms, i.e., procedures that verify the digital identity of users and check their legitimacy. In this context, users have to exhibit an *identity proof* that can only be provided by the users themselves, thus deterring attackers from breaching their online resources. The most common identity proof consists of user credentials, i.e., username and password. However, they are often considered insufficient to achieve an adequate level of security and their use exposes users to several threats [2].

To tackle this problem, banks have started adopting Multi-Factor Authentication (MFA). MFA is based on a security protocol, called MFA protocol, that integrates the use of credentials with additional identity proofs (the so-called *authentication factors*). Authentication factors are based on either *knowledge*, *possession* or *inherence*. During the execution of an MFA protocol, authentication factors are provided through specific objects, called *authenticators*. Therefore, an attacker stealing user credential cannot execute an MFA protocol without also controlling the necessary authenticators.

When properly designed and implemented, MFA protocols provide strong security guarantees. Clearly, such guarantees can decay in case of a poor design. Designing security protocols is error-prone and many protocols implemented and deployed in real

applications have been found flawed years later [3]. The design of an MFA protocol, especially if compared to the one of “standard” authentication protocols, is particularly challenging. Indeed, the type, number and order of employed authenticators, the associated authenticator factors and the employed communication channels have a significant impact on the security properties of MFA protocols. In addition, MFA protocols can be used to perform operations either from desktop computers (called *Internet Payments* - IP) or from mobile devices (called *Mobile Payments* - MP), requiring specific designs for tackling the different security assumptions underlying these endpoints. The ease-of-use of an MFA protocol is also of paramount importance to assess its efficacy. As shown in [4, 5, 6], the use of multiple authenticators in the execution of an MFA protocol can negatively affect user experience, which can have an impact on its security. On top of that, the preliminary phases of MFA, i.e., the registration of a new customer (called *enrollment*) and the *binding* of authenticators to users, require special attention to properly establish identity proofs and the associated user identity.

In order to regulate the design and adoption of MFA protocols and improve their security, a number of initiatives like FIDO [7] and OATH [8] have proposed to standardize MFA protocols. Moreover, public and private authorities have introduced regulations, directives and guidelines to steer their development and usage. For instance, the European Banking Authority (EBA) acknowledged the importance of MFA in the online banking context and, in 2013, issued directives and recommendations for online payment service [9, 10]. More recent payment service directives [11] and related regulatory technical standard [12] strongly bound online banking with MFA, explicitly stating the features that MFA protocols should support to be legitimately used for online banking. Similarly, other standardization bodies like the National Institute of Standards and Technology (NIST) [13] and Payment Card Industry (PCI) [14] have proposed a set of guidelines concerning the digital identity management through MFA. Similar initiatives are also carried out by private companies, which have started releasing their own guidelines [15, 16, 17].

In principle, these initiatives aim to guide the design of more secure and usable MFA protocols. However, the actual security and effectiveness of MFA remain uncertain. The main reason lies in the lack of a standardized approach in the adoption of MFA and in the consequent large number and heterogeneity of proprietary MFA protocols that

emerged over the last years. The goal of this work is to understand the state of affairs in the adoption of MFA in the context of online banking services.

*Our Contribution.* This paper presents a latitudinal study on the adoption of MFA and the design choices made by banks operating in different countries. In particular, we evaluate the MFA solutions currently adopted in the banking sector in terms of (i) compliance with laws and best practices, (ii) robustness against attacks and (iii) complexity. We also investigate possible correlations between these criteria. Our study mainly focuses on online banking in the European Union (EU) and is grounded on the EU legal framework. Nonetheless, it also analyzes the adoption of MFA by non-EU banks to provide a comparative benchmark and to obtain a more global view on state of affairs in the adoption of MFA in the banking sector.

For our study, we select 21 EU banks among those based in the first 7 countries for gross domestic product. As a reference with other important markets, we also select other 9 banks that are based in relevant countries (for the banking sector) but not subject to the EU legal framework, i.e., China, USA and Switzerland. For all banks, we review publicly available information (provided by the banks themselves) and collect data on the MFA protocols and authenticators as well as on the enrollment and binding procedures employed by each bank. The obtained dataset is used to investigate how MFA has been currently adopted by banks and evaluate their performances in terms of compliance with laws and best practices, resistance to attacker models and ease of use.

To evaluate the compliance of banks with laws and guidelines, we extract (i) relevant *legal requirements* from the EU regulations and directives concerning MFA (including recommendations for the security of Internet payments [9], those for mobile payments [10], the Payments Service Directive 2 [11] and the associated Regulatory Technical Standard [12]) and (ii) *best practices* from various documents, guidelines and white papers provided by NIST [13] and other relevant institutions in the online banking context [15, 16, 17].

The security of MFA protocols is evaluated by assessing their resistance against relevant *attacker models*. In particular, we adopt a classification of attacker models inspired to the classification proposed by NIST [13] and define an algebraic approach to

verify if an attacker model is able to compromise a given MFA protocol. To evaluate the ease-of-use of MFA protocols, we introduce a novel metric to assess the *complexity* of MFA protocols, i.e., the efforts required by users for their execution.

Moreover, we hypothesize that these criteria might not be independent from each other. To this end, we investigate whether these criteria are correlated. In particular, we investigate possible correlations between (i) the compliance with requirements and the complexity of MFA protocols, (ii) the compliance with requirements and the resistance of MFA protocols against attacks and (iii) the complexity of MFA protocols and their resistance against attacks.

Our study leads to several important insights. The analyzed banks tend to offer multiple MFA protocols to their customers, based on very different designs and employing different authenticators. However, the potential of authenticators and their security properties seem to be not fully understood yet. This has resulted in many complex MFA protocols that do not provide high security guarantees against attacks. In particular, the robustness of the analyzed MFA protocols against attacker models is, in general, lower than expected. However, we expect that the compliance with RTS [12], which will become in force in mid-2019, will improve the security level offered by MFA protocols.

*Related Work.* MFA is attracting increasingly attention in the banking sector and this resulted in the design of several MFA protocols for online banking, which are summarized in a few surveys. These surveys usually provide a classification and a comparison of MFA protocols and implementations. Choubey et al. [18] analyze the authentication mechanisms for IP employed by banks of 7 countries. In particular, the authors provide a classification of the adopted authenticators and emphasize the lack of a standardization in the design of MFA protocols. Kiljan et al. [19] review the authentication and communications protocols for online banking adopted by 80 banks worldwide. This study provides an analysis of the temporal evolution of MFA protocols adopted by banks, together with a classification of the used authentication factors and MFA protocols for both IP and MP. The security of MFA protocols for IP is evaluated by analyzing the implementation of the underlying TLS/SSL mechanisms whereas the security of MFA protocols for MPs is not analyzed. Dmitrienko et al. [20] analyze the

		Choubey et al. [18]	Kiljan et al. [19]	Dmitrienko et al. [20]		Krol et al. [5]	Althobaiti [21]	Our survey
	Number of Banks	60	80	4	10	–	–	30
	Geographical Distribution	UK,CA,IN,IRL US,RSA,AU	World	DE	UK	SA,UK	DE,UK,FR,IT,ES, NL,SW,CN,US,CH	
<b>Dataset</b>	Number of MFA Protocols	–	80*	6	9	3	61 (153)	
	Endpoints	IP	IP, MP†	MP	IP	IP	IP, MP	
	Temporal evolution	–	–	–	–	–	–	
	Authentication factors/authenticators	✓	✓	–	✓	✓	–	✓
	Enrollment and Binding	–	–	–	–	–	–	✓
<b>Compliance</b>	Regulations	–	–	–	–	–	–	✓
	Best Practices	–	–	–	–	–	–	✓
<b>Security</b>	Security of TLS/SSL implementation	–	✓	✓	–	–	–	–
	Perceived security	–	–	–	✓	✓	–	–
	Resistance of MFA protocols against attacker models	–	–	✓	–	–	–	✓
<b>Usability</b>	Perceived usability	–	–	–	✓	✓	–	–
	Complexity	–	–	–	–	–	–	✓
<b>Correlations</b>	Exemptions and Complexity	–	–	–	–	–	–	✓
	Compliance with security requirements and resistance to attackers	–	–	–	–	–	–	✓
	Complexity and resistance to attackers	–	–	–	–	–	–	✓
		* No reference to unique MFA protocols.						
		† Only classification of authenticator factors.						

Table 1: Comparison with related work.

security of 6 commonly used MFA protocols for MP. In particular, they identify the main weaknesses of these MFA protocols in terms of potential implementation errors and resistance to attacker models. Krol et al. [5] analyze the usability and perceived security of the authentication mechanisms employed by 10 UK banks (for a total of 9 MFA protocols) through user studies. Similarly, Althobaiti [21] evaluates the security and usability of MFA protocols based on questionnaires and field tests. Finally, it is worth mentioning that this work substantially extends [22], in which the authors pose the basis of the methodology used in this study.

The aforementioned studies differ from each other for the analyzed features and scope. Table 1 summarizes the main differences between those studies and our study. A primary difference is in the analyzed dataset and, in particular, in the number of banks and MFA protocols considered.

All surveys provide an analysis of MFA protocols along with the used authenticators, with the exception of the work in [18], which only provides a classification of authenticators without analyzing the protocols in which they are used. However, most surveys only analyze MFA protocols specific to one endpoint, with the majority considering protocols for IP. Our survey considers both protocols for IP and MP, since they might provide different security levels and user experience. Existing surveys also do not consider user enrollment and the binding of authenticators. Nevertheless, these phases can affect the overall security of an MFA protocol. Moreover, none of the previous works assesses the compliance of MFA solutions with laws and best practices. We claim that this aspect is also relevant, since often laws and best practices define a baseline for the security guarantees that an MFA protocol must provide.

Security aspects of MFA protocols are considered by most surveys, but at a different level compared to our work. For instance, some surveys [19, 20] analyze weaknesses in MFA implementations, whereas others [5, 21] focus on the security of MFA protocols perceived by users. In contrast to these studies, our work evaluates the security of MFA protocols by assessing their robustness against some attacker models. This analysis aims to compare MFA protocols in terms of resistance to well defined attack scenarios. At the best of our knowledge, the only other proposal considering an attacker model for MFA protocols is [20]. However, their attacker model only considers the MP context.

Moreover, only a few surveys [5, 21] evaluate the usability of MFA protocols. However, differently from those surveys that evaluate perceived usability and user satisfaction of MFA protocols through user studies, we focus on the efficiency of MFA protocols and propose an “objective” measurement of the complexity of MFA protocols, which can be computed from the dataset at hand. Finally, our survey is the only one that aim to discover correlations between compliance with laws and best practices, security and usability aspects. Leveraging this investigation, we are able to verify how the different features of MFA protocols and their compliance with laws and best practices are realized along with their effects.

*Structure of the paper.* The remainder of the paper is structured as follows. Section 2 provides background knowledge on MFA. Section 3 presents the requirements and best prac-

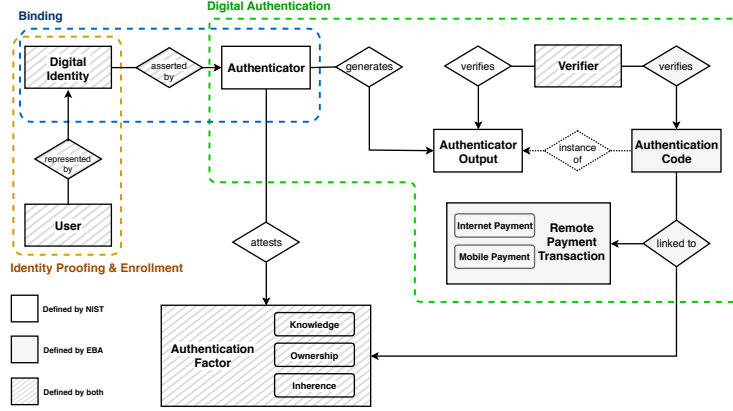


Figure 1: MFA Conceptual Model.

tices extracted from directives and regulations. Section 4 presents our methodology. In particular, we present a description of our dataset along with the selected features and the research questions along with the evaluation criteria. Section 5 presents and discusses the obtained results, with a specific focus on the compliance of banks with requirements and best practices, along with a security and usability evaluation of MFA protocols. Section 6 discusses potential threats that may have undermined the obtained results. Finally, Section 7 presents lessons learned and open challenges and Section 8 concludes the paper.

## 2. Background

In this section, we introduce the main concepts related to Multi-Factor Authentication (MFA) in payment services. Our study of the literature has shown the lack of a common and consistent terminology in the field. Among the others [23, 24, 25, 26, 27, 28], we have identified two main authoritative bodies, namely the *National Institute of Standards and Technology* (NIST) [13] and the *European Banking Authority* (EBA) [9, 10, 11, 12, 29]. These authorities target different aspects of MFA, i.e., the application of MFA for strong user authentication and for online payment services respectively. Here, we revisit and align the concepts from those sources to build a common terminology. The identified concepts and their relationships are depicted in Figure 1.

Authentication is typically employed in information systems to verify users' identity,



thus constituting a prerequisite to allowing access to resources. A user's *digital identity* is defined in [13] as a set of attributes that uniquely describe a user in a specific context (e.g., a payment service). In practice, the verification of a user's digital identity is performed through a so-called *authentication protocol*. An authentication protocol is a sequence of actions that allow the *digital authentication* of a user by verifying the possession and control of specific categories of credentials called *authentication factors* (AF) (by the NIST in [13]) or *authentication elements* (by EBA in [9, 10, 11, 12]). An *authentication factor* can be of three different types: (i) something the user knows (*knowledge factors*); (ii) something the user possesses (*ownership factors*); or (iii) something the user is (*inherence factors*). When an authentication protocol leverages more than one authentication factor, it is referred to as MFA protocol. The user starts the MFA protocol from her *endpoint*, e.g., a web browser or a mobile phone, and she authenticates by means of her authenticator factors.

The possession and control of authentication factors is attested through some specific objects called *authenticators*. An authenticator is “something the user possesses and controls (typically a cryptographic module or password) that is used to authenticate the user's identity” [13]. There exist a variety of authenticators, each providing different factors and/or having different features. Classic examples of authenticators are memorized secrets, look-up secrets, out-of-band devices, one time password (OTP) devices (see Section 4.2.2 for a detailed overview). An authenticator can attest more than one authentication factor in which case it is referred to as *multi-factor authenticator*.

Every authenticator can *generate* an output value on demand, called *authenticator output*. The ability to generate valid authenticator outputs proves that the user possesses and controls the authenticator (and thus the corresponding authentication factors). Nevertheless, the relationship between an authenticator and its output depends on the nature of the authenticator itself. In the case of a knowledge factor, for instance, an authenticator and the corresponding output are the same entity (e.g., the password or the secret code itself). On the other hand, in the case of ownership factors, the authenticator is the object used to generate the output, for instance, an OTP, which is the authenticator output.

It is worth noting that EBA introduces a refined notion of authenticator output, called *authentication code* [12]. An authentication code is a unique code generated by a

cryptography-based authenticator and *dynamically linked* to a specific *remote payment transaction*. With remote payments transaction, EBA indicates two kinds of transactions: *Internet payments* and *mobile payments*. Internet payment refers to any card payment, credit transfer and transfer of electronic money via the Internet [9]. *Mobile payment* indicates any transaction for which payment data and instructions are transmitted or confirmed via a dedicated mobile application [10]. In this work, we distinguish these types of transactions since it allows differentiating digital authentication procedures based on the platform on which transactions are executed (desktop computers or mobile phones), allowing for a more fine grained analysis.

The verification of authenticator outputs and, thus, of authentication factors is performed by the so-called *verifiers*. A verifier is “an entity that *verifies* the user’s identity by verifying the user’s possession and control of the authenticators” [13]. The type of verifier depends on the nature of the corresponding authenticator. In the case of a memorized secret (e.g., a password), for instance, the verifier has “only” to check if the provided secret is correct. In the case of an OTP device, instead, the verifier has to generate an expected OTP and compare it with the received one in order to validate it.

To run an MFA protocol, users must be previously registered to the system and obtained the necessary authenticators. The registration phase, called *enrollment*, encompasses a preliminary process of user identification, called *identity proofing*. Through this process, a service provider collects, validates and verifies information about an individual. Once this process is performed, the service provider is able to recognize the identity of the individual with an adequate level of assurance. Authenticators are handed over users through the so-called *binding* phase (referred to as *delivery of credentials, authentication devices and software* by EBA in [11, 12]). The binding phase consists of three steps: *request*, *delivery* and *activation*. During the request step the user informs the bank that she wants to activate a certain AF. The second step concerns the delivery of authenticators to the user. The activation step aims to guarantee that AFs are properly delivered. Binding can be executed by a human operator or remotely, e.g., over the Internet or via registered mail, potentially leveraging an MFA protocol employing previously bound authenticators. Note that additional binding operations can be performed in a separate occasion, i.e., whenever a user wants to associate a

new authenticator to her identity. Hereafter, we refer to the design choices of a bank concerning the adoption of MFA protocols, authenticators, enrollment and binding procedures as MFA implementation.

### 3. Requirements and Best Practices

Several public and private stakeholders have defined requirements and best practices for the implementation of MFA systems. In this section, we identify and list the ones that are relevant for this study. A summary of the identified requirements and best practices is presented in Table 2 and Table 3, respectively. There, we use ●, ◐ and ○ to denote whether a statement is fully, partially or not defined by a certain source. The requirements and best practices will drive our review process of MFA implementations.

#### 3.1. Requirements

The main source of requirements for MFA-based e-payment systems are EU regulations.<sup>1</sup> We group requirements according to their scope: (i) *authenticator* requirements refer to specific features that authenticators must comply with; (ii) *digital authentication* requirements refer to properties of the MFA protocols employed for digital authentication; (iii) *identity proofing* and *binding* requirements refer to properties of enrollment and binding phases, respectively. Below we discuss those requirements, which are summarized in Table 2.

*Authenticator requirements.* RTS [12] requires that authenticators are tamper-proof. This, however, is a hard requirement to meet, in case of software authenticators that run on multi-purpose, e.g., mobile devices (this will be detailed in Section 4). As highlighted in [32], software authenticators are relatively easy to compromise if the platform running

---

<sup>1</sup>We also considered the Payment Card Industry (PCI) Data Security Standard [30, 31] (becoming effective in 2018). As a matter of fact, PCI applies to any service that stores and manages payment card data, which includes e-banking services. Nevertheless, the requirements specified in the Data Security Standard are either too generic (e.g., provide documentation to the user for an informed usage of MFA) or too specific for the single implementation (that we are not able to test). For this reason, these requirements have not been considered in our survey.

		RSIP [9]	RSMP [10]	PSD2 [11]	RTS [12]
<b>Authenticators</b>					
<b>RL1</b>	If a software authenticator or an authentication code is used through a multi-purpose device, the integrity of the device must be checked	○	○	○	●
<b>Digital Authentication</b>					
<b>RL2</b>	MFA protocols must be always employed when the user performs risky operations	●	●	●	●
<b>RL3</b>	Every MFA protocol must employ at least two different types of AFs	●	●	●	●
<b>RL4</b>	Every MFA protocol must employ at least two independent AFs	●	●	●	●
<b>RL5</b>	Every MFA protocol must result in the generation of an authentication code that is unique, dynamically linked to a specific operation and accepted only once.	○	○	●	●
<b>RL6</b>	Every MFA protocol must make the user aware of crucial information on the operation she is going to authorize	○	○	○	●
<b>Enrollment and Binding</b>					
<b>RL7</b>	Identity proofing must be performed with a high level of confidence	○	○	○	●
<b>RL8</b>	The binding procedure for every authenticator must be executed in a secure manner	●	●	●	●
<b>RL9</b>	Every remotely delivered authenticator must be activated before its usage	○	○	○	●

Table 2: Key Requirements in EU regulations and directives.

them is compromised. Therefore, according to [12], when a software authenticator or an authentication code is used through a multi-purpose device, such a device must be checked against alterations (**RL1**). The aim of this requirement is hence to limit the aforementioned weakness as much as possible, by imposing banks to equip their software authenticators and applications with mechanisms for ensuring the integrity of mobile devices and potentially blocking the execution of payments from an insecure endpoint.

*Digital Authentication requirements.* The specifications concerning digital authentication based on MFA have evolved over time. While some initial requirements on MFA were given in RSIP [9], more precise and stringent requirements were defined

in later regulations and directives. For instance, PSD2 [11] introduces the concept of authentication code and dynamic linking (further refined in RTS [12]).

EBA defines specific situations in which MFA must be adopted (**RL2**). In particular, PSD2 and RTS require the adoption of MFA when (i) accessing a personal account for the first time in 90 days, (ii) initiating a payment transaction and (iii) executing any operation that may imply a risk of fraud or other abuses. Hereafter, we refer to such operations as *risky operations*. Intuitively, a risky operation is an operation that may lead to a leakage of sensitive data or to an economical damage. The operations to be considered as *risky* were identified by EBA after public consultations with several stakeholders within the online banking context. Specifically, the identification process took into account the outcome of empirical research and risk analysis procedures that are normally adopted in the field.

EBA also explicitly lists a number of situations in which MFA may not be employed. These exemptions for the MFA employment are regulated in [12]. In particular, operations such as (i) checking the account balance, (ii) paying a trusted beneficiary, (iii) executing a recurring transaction and (iv) executing a bank transfer between user's own accounts may not require MFA. This requirement has an impact on both the security and usability of MFA protocols. In particular, **RL2** helps in identifying the situations in which the security level provided by an MFA protocol should be reasonably high. The exemptions, instead, helps identifying those situations in which the user can be relieved from MFA, increasing the usability and perceived ease of use.

Further requirements are provided for the design of the MFA protocol itself. A first design requirement, defined already in [9], concerns the variety of AFs employed in a MFA protocol. In particular, EBA requires that AFs are *distinct*, i.e., at least two distinct types of AFs (e.g., one knowledge and one ownership factor) should be employed in the verification process (**RL3**). This requirement is crucial for MFA protocols, since the variety of AFs can strongly increase their security. The differentiation of AFs, indeed, forces a potential malicious agent to adopt different techniques in order to compromise an identity proof, hence making it difficult for the attacker to execute an MFA protocol on behalf of the user.

Another fundamental design requirement for MFA protocols is the *independence*

of the employed AFs. Specifically, after compromising one AF, an attacker must have no advantage for compromising the others. This concept is formalized in [9, 11, 12], where it is stated that AFs adopted by a MFA protocol must be independent (**RL4**). Reasonably, to compromise an AF, the attackers must control the authenticator that attests it. Therefore, an attacker has to control at least two different authenticators to compromise two AFs. Furthermore, EBA requires that an MFA protocol “shall result in the generation of an authentication code” [12]. EBA also states that the authentication code is the information used to authorize a specific payment transaction. For this reason, an authentication code should be OTP-generated (through a device or other cryptographic facility) and must be uniquely linked to a specific transaction (**RL5**). This requirement aims to improve the security level of MFA by strictly linking an authentication code to its context of use; accordingly, even if intercepted by a malicious agent, it can only be used for an operation that has been previously confirmed.

Finally, EBA requires in [12] that the user is made aware of (the crucial information about) the ongoing transaction (**RL6**). The information provided to the user should include (i) the amount of the transaction, (ii) the payee and (iii) the generated codes. Intuitively, this requirement aims to reduce the risk for the user to be induced to perform unwanted actions.

*Enrollment and Binding requirements.* Regulations and directives also define requirements both on the enrollment and binding procedures. For the former, EBA requires that users are identified with a high level of assurance (**RL7**). This is because issues in this phase can affect the reliability of the entire MFA process. In particular, EBA requires that user identification is carried out by a trained person [12]. Additional constraints are imposed by eIDAS regulations [33], which require user identification to be based on highly reliable identification proofs. For the binding process instead, two main requirements are defined by the EBA. The first addresses the delivery of authenticators [9, 10]. In particular, each authenticator must be delivered exactly to the intended user and in a secure manner (**RL8**). Accordingly, authenticators should be delivered to users personally after an in-person (*de visu*) identification. In case of “remote binding”, authenticators should be delivered only after the user has been identified through MFA

(by means of previously bounded authenticators).

An additional (and more specific) requirement concerning the binding phase requires an activation procedure for those authenticators that have been remotely delivered (**RL9**). Leveraging an activation procedure, indeed, a service provider can provide users with a unique code to be inserted in the delivered authenticator in order to uniquely bind it to their identity.

The aforementioned requirements (**RL7**, **RL8** and **RL9**) are clearly intended to limit the possible risks in the operations that are preliminary for the MFA usage. The security guarantees offered by an MFA protocol do indeed depend on the aforementioned operations: if an attacker manages to obtain an authenticator, the overall security of any MFA protocol depending on that authenticator is compromised.

### 3.2. *Best Practices*

Best practices typically target the technical details of an MFA implementation. In this case, the main sources are both private and public authorities involved in the security review process. Here, we discuss the guidelines released by the NIST [13] and PCI-SSC (MFA Guidelines, [14]) as well as by three independent vendors of digital identity security systems, i.e., Gemalto [16], PingIdentity [17] and Centrify [15]. We categorize best practices following the same criteria used for the requirements.

*Authenticator best practices.* Best practices on authenticators put a major emphasis on mobile devices, e.g., smartphones and tablets. For instance, Ping Identity [17] states that, if a service requiring MFA employs a mobile banking application, the capabilities of the authenticator should be integrated in it (**BP1**). That is, the service provider should not use a separate authentication application. This best practice aims to improve the usability of MFA protocols by reducing the amount of applications that the user has to interact with. **BP1** can somehow be related to **RL1**: joining two software applications into one might help in the security measures effectiveness, since security functionalities do not have to be implemented twice.

*Digital authentication best practices.* These best practices deal with the digital authentication process with a particular emphasis on MFA protocols and their features. For

		NIST [13]	PCL-SSC [14]	CENTRIFY [15]	GEMALTO [16]	PING IDENTITY [17]
<b>Authenticators</b>						
<b>BP1</b>	A software authenticator should be integrated in the mobile banking application (if any)	○	○	○	○	●
<b>Digital Authentication</b>						
<b>BP2</b>	MFA protocols should rely on standard solutions	○	◐	●	●	○
<b>BP3</b>	Step-up authentication should be adopted	○	◐	●	●	●
<b>BP4</b>	MFA protocols should limit SMS reception as much as possible	●	●	○	○	●
<b>Enrollment and Binding</b>						
<b>BP5</b>	Identity proofing should be executed with high level of confidence	●	◐	○	○	○
<b>BP6</b>	The binding procedure should be executed in a secure manner	●	●	○	●	●
<b>BP7</b>	Two authenticators attesting ownership factors should be bound after the enrollment	●	○	○	○	○
<b>BP8</b>	The user should be offered with multiple authenticators of different types	●	○	●	◐	●

Table 3: Selected Best Practices.

example, Centrify [15] and Gemalto [16] recommend to implement MFA protocols using standard solutions (**BP2**). This because, unlike proprietary algorithms, standardized algorithms go through public scrutiny by industry and security experts, which reduces the chance of any inherent weakness or vulnerability. This best practice adds some constraints to the design requirements related to MFA protocols (i.e., **RL3** to **RL6**) by encouraging the employment of solutions that comply with the aforementioned requirements and by limiting the adoption of ad-hoc solutions that may introduce security issues.

Moreover, best practices in [15, 16, 17] provide recommendation on the situations in which MFA should be adopted. In particular, they recommend to implement adaptive or



“step-up” authentication procedures. This means that MFA should be avoided when not strictly necessary (**BP3**). A way to implement this is to require an increasing number of authenticators depending on the actions to be performed by the user. This best practice aims to improve the perceived ease-of-use during the access to remote data. Step-up authentication relieves users from the burden to achieve an unnecessary high security level by requiring MFA only in situations where it is needed. It is worth noting that this best practice exemplifies the exemptions introduced in **RL2** by suggesting practical solutions to be adopted in low-risk scenarios.

Another best practice targets the adoption of SMS services, therefore adding constraints to MFA protocol requirements. In particular, both the NIST and PCI-CSS (in [13] and [14], respectively) deprecate the usage of out-of-band authentication via SMS (**BP4**). This is because the reliability of this kind of authenticator has been recently questioned: a large amount of malware has specifically targeted this authentication method to obtain sensitive data for MFA [34, 35]. Moreover, the advantage of using alternative channels (compared to classic HTTPS connection) may be nullified if an SMS is received on the same phone from which a payment operation is started [14]. According to [36], the adoption of SMS can lead to situations in which the mobile phone of the user constitutes a single point of failure whose exploitation can compromise the security of all communication channels.

*Enrollment and binding best practices.* The NIST provides a detailed explanation in [13] on how enrollment and, in particular, identity proofing should be performed. In our context, it can be summarized by asserting that identity proofing should be performed with high level of confidence (**BP5**). This can be performed by an in-person identification and/or based on documents and certifications issued by a so-called qualified entity (e.g., a national authority). Regarding the binding process, the best practices released by many public and private authorities [13, 14, 16, 17] require service providers to bind authenticators to the user’s identity in a secure manner (**BP6**). This because “an authentication mechanism is only as strong as the binding process that issued the credentials” [17]. It is worth noting that **BP5** and **BP6** match requirements **RL7** and **RL8**, respectively. This underlines the paramount importance of the identity

proofing and binding processes in the security of MFA systems.

According to [13], at least two physical authenticators should be bound to the user's identity immediately after enrollment (**BP7**). This best practice is related to **RL9** and permits users to be immediately able to provide ownership factors for digital authentication or for future remote bindings. Moreover, a number of authorities [13, 15, 17] recommend service providers to support flexible authentication procedures (**BP8**). This reduces to allowing the user to select among multiple, alternative authenticators of different types. Customizing the MFA experience could, indeed, increase the perceived usability of an MFA implementation. Binding multiple authenticators also makes it possible to recover from the loss or theft of other user's authenticators.

The best practices above have been extracted from guidelines and white papers concerning MFA applied in a generic context. Nevertheless, they can be easily mapped to one or more requirements defined in EBA documents. Differently from what one may expect, the best practices related to the security aspects of MFA are rarely more strict than the related requirements (except for the one concerning the SMS usage). On the other hand, it is noticeable how their adoption can increase the ease-of-use of MFA protocols.

#### **4. Methodology**

In this section we present the methodology that we adopted for the analysis of MFA solutions adopted by banks.

##### *4.1. Research Questions*

The aim of this work is to understand the status, in terms of adoption, security and complexity, of MFA implementations employed by banks. To this end, we put forward five research questions. We advocate that these questions are both relevant and addressable with the available data. We consider them relevant as they characterize critical aspects of the adoption of MFA protocols within online banking. As a result of our research, we provide the answers to these questions in Section 5.

*RQ1: What is the landscape (demography) of the MFA implementations adopted by banks?* This question aims to understand the design choices made by banks with respect to their MFA implementations. Specifically, we investigate whether multiple choices of AFs and MFA protocols are given, which authenticators are employed, how users are required to enroll to the bank and how they can bind authenticators to their identity. On the top of this, we perform an analysis of common practices and potential trends in the adoption of MFA in the banking sector.

*RQ2: Do MFA implementations adopted by banks comply with requirements and best practices?* We look for evidences indicating whether MFA implementations currently adopted by banks comply with the requirements and best practices we identified in Section 3. On the one hand, assessing the compliance of a digital authentication solution with the regulations and directives in force provides an indication of its correctness and legitimacy. On the other hand, best practices are related both to the mitigation of security risks and improvement of the ease-of-use: ignoring them might expose the system to security threats or require the user to execute complex authentication procedures.

*RQ3: How do the MFA protocols adopted by banks behave w.r.t. the relevant attacker models?* Our goal is to identify the attacker models that are more likely to succeed against the MFA protocols currently employed by banks. Here, we consider (a slight extension of) the attacker models described in [13]. In particular, for the MFA protocols employed by each bank, we identify which (groups of) attackers can compromise them. Through this evaluation, we aim to assess the robustness of MFA protocols against typical attack scenarios.

*RQ4: How complex are the MFA protocols adopted by banks?* Our goal is to evaluate the ease-of-use of the MFA protocols adopted by banks. The ease-of-use of an MFA protocol mainly depends on the authenticators it uses. In fact, authenticators might significantly differ in terms, e.g., of interaction with the users, input and output data. To answer this question, we define in Section 4.3 a metric to compute the complexity of an MFA protocol based on the resources (i.e., memory, manual operations and devices) required to execute the protocol. The analysis of the complexity of the MFA protocols

adopted by banks provides us with an indication of how much importance is given to usability in the design of MFA protocols in the banking sector.

*RQ5: Are there significant correlations between compliance of MFA implementations (with requirements and best practices), robustness against security threats and complexity of the MFA protocols adopted by banks?* Although interesting per se, the compliance (with requirements and best practices), robustness against security threats and complexity of the MFA protocols adopted by banks might exhibit some correlation between each other. For instance, one might wonder whether exemptions (RQ2) are more common among banks adopting complex authentication procedures (RQ4). This research question aims to investigate possible correlations between various aspects of MFA implementations. In particular, we hypothesize possible correlations (presented in Section 4.4) and verify the validity of our hypotheses. This analysis provides us with additional insights into MFA implementations adopted in the banking sector and allows us to evaluate the effects of the design choices made by banks.

#### 4.2. Dataset

Our analysis relies on a dataset collected by analyzing the resources made publicly available by banks. All information has been gathered from the official documentation, including web pages, handbooks and manuals, FAQ pages, security guidelines and more. Moreover, when available, we took into account interactive demos and video tutorials. Finally, we took advantage of unofficial, side data source as well. Among them, it was extremely effective to extract information from the provided Android applications. For instance, an analysis of the relevant permissions (e.g., `READ_SMS`), together with a manual inspection of the mobile application code and resources, can clarify whether an otp can be automatically read by a mobile application or not. The data had been collected and updated until June 30th 2018, and it is available for online consultation at <https://mfa-team.github.io>.

During data collection, we noticed that in some cases different sources (for the same bank) presented conflicting information. In such cases, we gave more importance to written documentation than to multimedia tutorials and we preferred more recent sources

rather than older ones. Moreover, in few occasions, the available documentation does not provide sufficient details of the MFA implementation. In these cases, we assumed the worst case scenario. A detailed description of the critical aspects and limitations of our dataset is provided in Section 6.

#### *4.2.1. Bank selection*

We carried out a systematic review of the digital authentication solutions adopted by 30 important international banks based in 10 different countries (3 banks per country). In particular, a first group includes 21 banks chosen among those based in the first seven countries in the European Union<sup>2</sup> for gross domestic product<sup>3</sup>, i.e., Germany, United Kingdom, France, Italy, Spain, Netherlands and Sweden. For each country, we selected the three largest banks (offering online banking services to individuals) in terms of assets (according to the Standard & Poor’s 2017 classification<sup>4</sup>). The second group of banks are based in relevant countries (for the banking sector) but not subject to the EU legal framework, i.e., China, USA and Switzerland. Again, for these countries we considered the three largest banks in terms of assets. The geographic distribution of the selected banks is depicted in Figure 2. This allows us to enlarge our perspective on the banking sector by investigating how non-EU banks behave with respect to MFA adoption. The validity and the potential limitations of the described choice of banks will be discussed in Section 6.

#### *4.2.2. Selected features*

In this section we describe the key features of the digital authentication and related aspects adopted by each bank in our dataset based on the aforementioned documentation. As done in Section 3, we group them according to their scope. Table 4 summarizes the notation introduced in this section.

#### *Authenticators*

For our analysis, we have identified the categories of authenticators offered by banks to

---

<sup>2</sup>The data collection started before Brexit.

<sup>3</sup><https://goo.gl/ZctkLc>

<sup>4</sup><https://goo.gl/HR3HDQ>

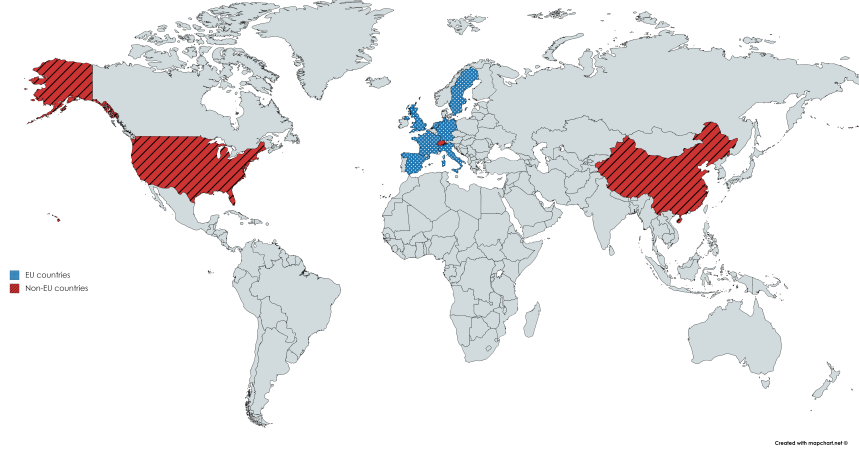


Figure 2: Geographic distribution of the analyzed banks. EU countries (filled with a blue pattern) are Spain, France, UK, Netherlands, Italy, Germany and Sweden. Non-EU countries (filled with a red pattern) are USA, Switzerland and China.

AUTHENTICATORS		
Data items	Data channels	
$\epsilon$ empty data item	$\gg_h$ manual copy	$\gg_i$ inter-process communication
$opid$ operation identifier	$\gg_o$ optical code scan	$\gg_m$ mobile telephony network
$otp$ one-time password	$\gg_n$ network packet	
EXEMPTIONS		
$\times$ no exemptions	$\checkmark$ personal data visualization	$\checkmark\checkmark$ low risk payments
ENROLLMENT AND BINDING		
$\text{III}$ the user goes to a local branch	$\text{🌐}$ the user establishes a remote session	
$\text{E}^\dagger$ the user runs a MFA protocol	$\text{E}^\dagger$ the user operates during the enrollment	
( $^\dagger$ binding only)		

Table 4: Notation for data I/O, enrollment & binding and exemptions.

their clients. We follow a classification of authenticators similar to the one proposed in [13] with few, minor modifications. In particular, here we propose a more rigorous categorization based on their distinguishing features. Beside the authenticator type, these features include (i) input/output data and channels, (ii) authentication factors and

(iii) interaction with the user.

Input/output data consist of two elements, i.e., *data item* and *channel*. A data item represents the content of an input/output value while a channel indicates the medium used to transmit it. Beside the empty data  $\epsilon$  (that we omit when clear from the context), we consider two data types, i.e., operation identifiers (*opid*) and one-time-passwords (*otp*) that contribute to the security of MFA protocols (other data types that do not actually contribute to the security of an MFA protocols are not explicitly modeled and treated as  $\epsilon$ ). Briefly, *opid* is a pseudo-random code that uniquely identifies a specific operation and *otp* is a special kind of authenticator output (see Section 2) that is pseudo-randomly generated by an authenticator and used only once. A data channel can be:  $\gg_h$  (i.e., the user manually copies a data item),  $\gg_o$  (i.e., a camera scans an optical code, e.g., a QR code),  $\gg_n$  (i.e., a packet is sent over the network<sup>5</sup>),  $\gg_i$  (i.e., a signal is sent through an inter-process communications mechanism) and  $\gg_m$  (i.e., the data item is sent via the mobile telephony network). For instance, *opid* $\gg_h$  indicates that an operation identifier (*opid*) is passed as an input by the user ( $\gg_h$ ), whereas  $\gg_n$  *otp* indicates that an authenticator sends a one-time password (*otp*) over the Internet ( $\gg_n$ ).

Authenticators can attest one or more AFs (see below). We indicate among brackets [...] the type of AF that is attested by the authenticator. In particular, we use K, O and I for knowledge, ownership and inherence, respectively. Finally, we label an authenticator with ? to denote that it shows the ongoing operation details and asks for the user's confirmation.

Below we list the categories of authenticators that we inherit from [13]. A schematic representation of these categories is depicted in Figure 3.

*Memorized secret* **Q**. A piece of information (sometimes called memorable information) that a user shares with the bank and that she has to recall. It attests a knowledge factor. For instance, passwords, PIN numbers, pass-phrases and answers to a secret question are examples of memorized secrets. It is worth noting that, with our representation, the output of this authenticator is omitted, since the authenticator output is the authenticator

---

<sup>5</sup>If not explicitly stated, we assume network communications to use secure, e.g., HTTPS, connections.

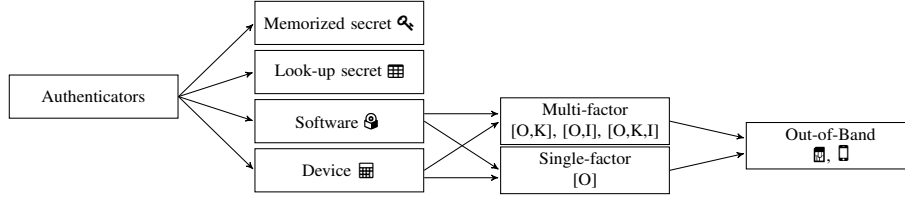


Figure 3: Authenticators classification.

itself and it is always manually copied by the user.

*Look-up secret*  $\text{grid}$ . A (physical or electronic) record that stores a set of secrets shared between the user and the bank. It attests an ownership factor. To generate an output, the user is requested to provide the secret associated to a certain position in the record. Also in this case, we omit the output of the authenticator, which is a code derived from the entries of the look-up secret<sup>6</sup> and it is always manually copied by the user on the endpoint.

*Authenticator device*  $\text{grid}$ . A piece of hardware used to generate an authenticator output. It usually attests an ownership factor. In [13] authenticator devices are grouped in two classes: *single-factor* and *multi-factor* devices. A single-factor device attests a single AF, that is the ownership of the device itself. A typical example of single-factor device is the so-called *time-based otp key*, or *token*. This type of device periodically displays a new OTP code that the user should manually copy to successfully complete the execution of an MFA protocol. With our notation, this can be written as  $\text{grid}[O] \gg_h \text{otp}$ . Instead, a multi-factor device attests more than one AF.<sup>7</sup> A common example of multi-factor device authenticator is a device that generates a unique *otp* from an alphanumeric string and that requires a PIN to be activated, thus attesting both an ownership and a knowledge factor. This authenticator can be represented as  $\text{opid} \gg_h \text{grid}[O,K] \gg_h \text{otp}$ .

It is worth noting that, in this work, authenticators leveraging a smart card for executing operations are included in this category of authenticators, hence being represented with symbol  $\text{grid}$ . This is because authenticator devices relying on a smart card usually

<sup>6</sup>Note that the output of this authenticator is not pseudo-randomly generated, hence it cannot be indicated using *otp*.

<sup>7</sup>Note that an authenticator device can rely on multiple AF of the same type, e.g., two or more ownership factors. However, this case is immaterial for our analysis. For this reason we avoid writing, e.g.,  $\text{grid}[O,O]$ .



do not attest an ownership factor (i.e., they are equal for every customer of the same bank). Hereafter, we represent those authenticators as  $\text{[O]}$  if, together with the reader, they only require a smart card, and  $\text{[O,K]}$  if a PIN is also required.

A further category of authenticator devices defined in [13] consists of *out-of-band* devices. These authenticators are uniquely addressable and communicate over a distinct, namely *secondary*, channel, i.e. a different channel from the user endpoint. Thus, at least one between the input and output channels of any out-of-band authenticator is labeled with  $n$  or  $m$ . Since this type of authenticator usually relies on a SIM card, i.e., a single-factor device attesting an ownership factor that establishes a secondary channel through the mobile phone network, we indicate it with  $\text{[O]}$  (rather than  $\text{[O]}$ ). For instance,  $\text{otp} \gg_m \text{[O]} \gg_m \text{otp}$  is an out-of-band device that receives an  $\text{otp}$  through the phone network. Note that, in this case, we assume that the received  $\text{otp}$  has been properly generated by the server and is dynamically linked to the specific ongoing transaction.

*Software Authenticator*  $\text{[O]}$ . A program that generates authenticator outputs. These authenticators are the software counterpart of authenticator devices. Thus, the same categories apply in this case. In particular, we distinguish between single-factor and multi-factor software authenticator. Moreover, a software authenticator can be out-of-band under the same condition of an out-of-band device. In this case, we use  $\text{[O]}$  to denote it (as out-of-band software commonly consists of a mobile application running on a mobile phone).

The notation introduced above allows us to concisely specify the MFA protocols adopted by banks. For instance, consider an MFA protocol that first requires users to provide a memorized secret and then to use an authenticator device to generate, from an  $\text{opid}$  displayed on her browser, an  $\text{otp}$  that is manually copied. This protocol can be represented as  $\text{[O]} ; \text{opid} \gg_h \text{[O]} \gg_h \text{otp}$ , where symbol “;” is used to separate the steps of the protocol.

### Digital authentication

In addition to the MFA protocols adopted by banks (see above), we are interested in two features of digital authentication, namely the user endpoint and the presence of any exemption. To gather this information, we took advantage of introductory pages,

handbooks and tutorials (more than 50% of the considered banks integrate an interactive tutorial in their mobile banking application).

*Endpoint.* We distinguish between digital authentication for *Internet Payments* (IP) and *Mobile Payments* (MP), as proposed in [9, 10].<sup>8</sup> Briefly, the user endpoint is a web browser for IP and a mobile banking application for MP.<sup>9</sup> The main motivation for differentiating between IP and MP is that they behave differently w.r.t. the attacker models introduced in Section 4.3.2.

*Exemptions.* For each bank, we checked whether exemptions are used for a certain kind of operations. This means that, in certain circumstances, banks allow their users to authenticate using a single-factor authentication protocol. This fact is usually reported on handbooks and tutorials. For instance, certain banks allow exemptions only for non-payment operations, e.g. balance check, while others also allow exemptions for low risk payments, e.g. payments toward trusted parties. Thus, we distinguish three types of exemptions: no exemptions (✕), exemptions for personal data visualization (✓) and exemption for low risk payment operations (✓✓).

#### *Enrollment and binding*





Both user enrollment and binding procedures require checking the user's identity. The official documentation provided by each bank includes a detailed description of how customers can identify themselves. For our analysis, we are interested in the modalities used by banks to verify user identity during enrollment, i.e., identity proofing, and binding phases.










*Enrollment.* In this phase users are identified by the bank and associated to their digital identity. Identity proofing can be performed either in person or remotely. Specifically, we identified two modalities for user identification, namely (🏠) in which users should go to a local branch and be identified in person and (📞) in which users are identified by interacting remotely, e.g., through a web portal or a call service.



---

<sup>8</sup>Interestingly, the most recent EU directive concerning MFA in the online banking services (i.e., PSD2 [11]) does not adopt the same distinction.

<sup>9</sup>Although in many cases, a user can run the IP authentication using a mobile phone browser.

*Binding.* Binding is a procedure that users should perform to associate a new authenticator to their digital identity. As described in Section 2, it consists of three steps, namely *request*, *delivery* and *activation*. Each of these steps may involve user identification. Clearly, identity checks during the binding phase can be carried out in the same way as for identity proofing (see above). Nevertheless, since the user has been previously enrolled, she already has a digital identity and, possibly, a previously bound authenticator. This enables one additional identification modalities, i.e., through an MFA protocol. We denote it with . In terms of level of assurance for the identification, we claim that  is better than  and  is better than both. Moreover, a number of banks allows the binding of one or more authenticators just after the enrollment phase. When this happens, we denote the *request* of these authenticators using symbol **E**.

Hereafter, we use a triple to represent the three steps of the binding procedure. Each element of the triple either contains one of the aforementioned symbols, i.e., , , , or “–” if the step requires no action (the first element of the triple – *request* – can also contain symbol **E**). For instance, (, , ) indicates a binding procedure in which the user requests an authenticator in person () , receives it through a remote delivery system () and activates it by running an MFA protocol () .

Notice that some banks offer alternatives to carry out these steps. When this occurs, we only consider the operation with lower level of assurance. Also, it may happen that two operations are needed to be carried out concurrently, e.g., an activation may require both  and . In this case, we only consider the operation with higher level of assurance as an attacker has to compromise both in order to gain control of the authenticator.

#### 4.3. Evaluation criteria

In this section we put forward a list of evaluation criteria for the selected banks in terms of compliance with requirements (Section 3.1) and best practices (Section 3.2), resistance to attacker models and complexity of employed MFA protocols. These criteria form the baseline for our analysis (Section 5).





ID	★	☆	☆	Criteria
<b>RL1</b>	Every	Some	No	banking application (including software authenticators) provides a device integrity check
<b>RL2</b>	No	n.a.	Some	risky operation is performed without MFA
<b>RL3</b>	Every	Some	No	MFA protocol relies on at least two AF
<b>RL4</b>	Every	Some	No	MFA protocol relies on at least two authenticators
<b>RL5</b>	Every	Some	No	MFA protocol contains $opid \gg \cdot \gg otp$ or $otp \gg \cdot \gg otp$
<b>RL6</b>	Every	Some	No	MFA protocol uses at least one of $\text{⌘}^?[\dots]$ and $\text{⌘}^?[\dots]$
<b>RL7</b>			n.a.	used for the enrollment
<b>RL8</b>	Every	Some	No	binding includes  or 
<b>RL9</b>	Every	Some	No	binding is $(\cdot, \text{⌘}, \cdot)$ , $(\cdot, \cdot, \text{⌘})$ or $(\cdot, \cdot, \text{⌘})$

Table 5: Summary of the evaluation criteria for requirements.

#### 4.3.1. Requirements and best practices evaluation

In this section we define the criteria to assess the fulfillment of the requirements and best practices presented in Section 3. It is worth noting that some requirements and best practices are defined at the level of banking application, authenticator and MFA protocol (e.g., **RL1**, **RL3** to **RL6**, **RL8** and **RL9**) and others at the level of bank. In our analysis, we evaluate the former group of requirements and best practices also at the level of bank. In this case, we consider three possible levels of fulfillment defined through the quantification over the banking application, authenticator and MFA protocols adopted by a bank: *fulfilled* (★) denotes that all adopted target elements (banking applications, authenticators or MFA protocols) satisfy the requirement; *partially/possibly*<sup>10</sup> *violated* (☆) denotes that some (but not all) adopted target elements satisfy the requirement; and *violated* (☆) denotes that none of the target elements satisfies the requirement. The criteria for assessing the compliance with requirements and best practices are summarized in Table 5 and Table 6, respectively.

#### Requirements:

<sup>10</sup>The compliance of a bank with some requirements (e.g., **RL7**) depends on the specific implementation adopted by the bank. In this cases, ★ indicates that the implementation adopted by a bank can potentially violate the requirement.

**RL1.** To determine whether a bank meets this requirement, we verify their software authenticators and mobile banking applications.<sup>11</sup> In particular, we inspect the code, looking for the use of root detection mechanisms. Thus, we state that **RL1** is fulfilled when the code of every application includes these checks, partially violated when only some applications include these checks, and violated otherwise.

**RL2.** This requirements focuses on the protection of risky operations (see Section 3.1) with MFA. Trivially, **RL2** is satisfied if the bank does not allow the execution of risky operations without MFA. Otherwise, **RL2** is violated.

**RL3.** This requirement is fulfilled by a bank if and only if all employed MFA protocols involve at least two AFs of different type. For instance, a protocol in which  $\mathcal{Q}_k$  is combined with  $\mathbb{A}[O]$  complies with **RL3** (since  $\mathcal{Q}_k$  subsumes a knowledge factor), while a protocol using authenticators  $\mathbb{A}$  and  $\mathbb{A}[O]$  does not (as both authenticators assert an ownership factor). If only a subset of the MFA protocols offered by a bank meets this criterion, the requirement is considered partially violated by the bank. Finally, if none of the offered MFA protocols involve at least two AFs of different type, **RL3** is violated.

**RL4.** As stated in Section 3.1, we assume that two AFs are independent when an adversary has to control at least two authenticators to compromise both. For instance, this is the case for  $\mathcal{Q}_k; \mathbb{A}$ . Conversely, two AFs are not independent when they are attested by a single, multi-factor authenticator, e.g.,  $\mathbb{A}[O, K]$ . Under this interpretation, **RL4** requires that every MFA protocol adopted by a bank employs at least two distinct authenticators. Hence, a bank partially violates **RL4** if some MFA protocols do not use more than one authenticator; if none of the MFA protocols uses more than one authenticator, we consider **RL4** violated.

**RL5.** This requirement is satisfied by a bank if and only if all employed MFA protocols result in an authenticator output that is uniquely associated to the ongoing operation. In symbols, we require that an MFA protocol includes at least one authenticator with

---

<sup>11</sup>We only considered the Android version of banking applications.

the form  $\text{opid} \gg \cdot \gg \text{otp}$ <sup>12</sup> or  $\text{otp} \gg \cdot \gg \text{otp}$ . For example, an MFA protocol relying on  $\text{opid} \gg_i \text{[O,K]} \gg_i \text{otp}$  does link the output to the ongoing operation, while one only relying on  $\text{[O,K]} \gg_h \text{otp}$  does not. If only some of the protocols satisfy it, **RL5** is considered partially violated and, if none of the protocols returns the desired authenticator output, **RL5** is violated.

**RL6.** This requirement is satisfied by a bank if and only if all employed MFA protocols employ at least one authenticator labeled with **?**, i.e., in the case of  $\text{[O,K]}^?[\dots]$  and  $\text{[O,K]}^?[\dots]$ . In contrast, if such an authenticator is only used in some of the MFA protocols, **RL6** is partially violated; if none of the provided MFA protocols uses it, **RL6** is violated.

**RL7.** The required level of assurance is clearly achieved when enrollment is carried out in person (**III**). Otherwise, i.e., when enrollment is performed remotely (**IV**), we claim that **RL7** is possibly violated.

**RL8.** As for **RL7**, the binding procedure of an authenticator provides the required level of assurance only if one step is done at the bank (**III**) or through an MFA protocol (**V**). Note that an MFA protocol executed in a binding step should comply at least with **RL3** and **RL4** and the authenticators it employs should have been bound in compliance with **RL8** and **RL9**. If the request step includes **E**, it is necessary to consider the modality in which the enrollment is executed (for that bank). Thus, a bank satisfies **RL8** if the binding procedure for all its authenticators provide the required level of assurance. Instead, the requirement is partially violated when only some binding procedures rely on in-person identification or MFA. Finally, if no binding procedure requires in-person identification or MFA, **RL8** is violated.

**RL9.** This requirement is satisfied under two circumstances: either the delivery phase consists of **III** or there is a secure activation step (i.e., **III** or **V**). On the other hand, we consider **RL9** possibly violated if these conditions are met by a bank only for some authenticators. In all other cases, the bank violates **RL9**.

---

<sup>12</sup>We assume that if an *opid* is received as an input, it is actually used to generate the output, which is therefore assumed unique. Unfortunately, we cannot verify if the output is accepted only once by the server.

ID	★	☆	☆	Criteria
<b>BP1</b>	Every	n.a.	No	🔑 and/or 📱 is integrated in MP endpoint
<b>BP2</b>	Every	Some	No	relevant (cfr. Table 7) API is used by banking applications
<b>BP3</b>	✓ or ✎	n.a.	✗	exemptions are adopted by the bank
<b>BP4</b>	No	Some	Every	MFA protocol relies on 📱
<b>BP5</b>	–	–	–	same as <b>RL7</b>
<b>BP6</b>	–	–	–	same as <b>RL8</b>
<b>BP7</b>	2+	1	0	among 📱[O,⋯], 🔑[O,⋯] and 📱 have binding procedure of the form (E,⋯)
<b>BP8</b>	2+	n.a.	1	type of authenticator is provided

Table 6: Summary of the evaluation criteria for best practices.

*Best practices:*




**BP1.** This best practice is fulfilled whenever a bank offering software authenticators integrates their functionality in the mobile banking application. Otherwise, the best practice is violated. In the case a bank does not provide users with any software authenticator (🔑 and 📱), **BP1** is considered to be fulfilled.





**BP2.** In general, checking the adoption of standard technologies requires disassembling and inspecting the target object. Since we cannot do this for remote services and authenticator devices, here we only focus on software authenticators (used both for IP and MP) and mobile banking applications. In particular, we decompiled the Android applications (both software authenticators and banking applications) offered by each bank and inspected the code looking for standard APIs (according to [37, 38, 39]). Table 7 defines the relationship between the (elements of the) MFA protocols and the standard APIs that it should use, namely the relevant APIs. Thus, we claim that a bank satisfies **BP2** when all its Android applications use the standard APIs that are relevant for the MFA protocols they are involved in. For instance, for the MFA protocol  $\text{opid} \gg_n \square[\text{O}] \gg_n \text{otp}$  we expect to find the APIs `com.google.android.gms.safetynet.*` (always), `javax.net.ssl.*` (due to  $\text{opid} \gg_n$  and  $\gg_n \text{otp}$ ), `firebase.messaging.*` (due to  $\square$ ) and `android.security.keystore.*` (due to  $[\text{O}]$ ). If only some applications use those API, we say that **BP2** is possibly violated and, when no application uses them, we say that the best practice is violated. Note that this evaluation criterion adheres as

Element	Relevant API	Used for
always	com.google.android.gms.safetynet.*	Integrity
I	android.hardware.fingerprint.*	Fingerprint
O	android.security.keystore.*	Secure storage
K	android.widget.EditText ("textPassword" mode)	Password field
☎	android.telephony.TelephonyManager	Telephony network
📧	firebase.messaging.*	Push notification
» <sub>i</sub>	android.content.Intent (permission protected)	Android IPC
» <sub>o</sub>	gms.vision.barcode.* or firebase.ml.vision.*	Optical code
» <sub>n</sub>	javax.net.ssl.*	HTTPS & SSL

Table 7: Relationship between MFA protocol elements and relevant Android APIs.

much as possible to **BP2**. Nonetheless, we stress that the usage of commercial APIs – different from the ones listed above – does not necessarily mean that a less secure digital authentication is provided. However, a security analysis of APIs is out of the scope of this work; the implications of this criterion on our analysis are discussed in Section 6.

**BP3.** This best practice is fulfilled when a bank adopts a step-up authentication driven by the risk level of the operation to be performed. In this context, the fact that a bank defines multiple risk levels is indicated by the exemptions it adopts. Thus, we state that a bank satisfies **BP3** if it adopts some exemption, i.e., either  or . We consider **BP3** violated by banks that do not use exemptions, i.e., .



**BP4.** This best practices concerns the usage of SMS messages (received through ) in MFA protocols. Clearly, if none of the MFA protocols employed by a bank uses , the bank satisfies **BP4**. Otherwise, if some (but not all) of its MFA protocols leverage , the bank partially violates **BP4**; if all MFA protocols make use of , the bank violates **BP4**.

**BP5.** This best practice requires users to exhibit an official identification document to a bank clerk. The best practice is hence subsumed by **RL7**. Therefore, we evaluate **BF5** using the same criterion defined for **RL7**.



**BP6.** This best practice is subsumed by **RL8**. Accordingly, its satisfaction is assessed using the criterion defined for **RL8**.

**BP7.** This best practice states that the user should receive at least two authenticators devices (or one authenticator device and a look-up secret) immediately after the enrollment phase. Thus, a bank satisfies **BP7** if the binding procedure for at least two authenticator devices (or look-up secrets) is performed during enrollment (**E**), i.e., the procedure has the form (**E**, ·, ·). The best practice is possibly violated if this happens for only one authenticator. Otherwise, **BP7** is violated.

**BP8.** This best practice is fulfilled by a bank whenever the bank provides its users with at least two different types of authenticators, e.g.,  and . Otherwise, if at most one type is provided, **BP8** is violated.

#### 4.3.2. Attacker models and applicability

The robustness of MFA protocols against attacks is a critical aspect in the evaluation of their security. The NIST [13] defines several attacker models. Here, we follow the same approach but with few, minor refinements (see below). Briefly, these refinements are necessary to apply the attacker models to our definition of MFA protocol.

Each attacker model is characterized by an application condition, i.e., a set of capabilities in terms of which authenticators an attacker can compromise. To precisely define the application conditions and effects of each attacker, we follow an algebraic approach. We use symbol  $\mathbf{1}$  to denote the unit element w.r.t. “;” (the sequence operator), i.e.,  $\mathbf{1};S = S; \mathbf{1} = S$  for any sequence  $S$ . Let  $A$  be the set of all authenticators (as defined in Section 4.2.2), we define an attacker as a total function  $f : A \cup \{\mathbf{1}\} \rightarrow A \cup \{\mathbf{1}\}$ . Intuitively,  $f(X) = Y$  means that the capabilities of the attacker allow her to treat the authenticator  $X$  as if it was  $Y$ , i.e.,  $X$  and  $Y$  provide an equivalent protection against  $f$ . When  $f(X) = \mathbf{1}$  we say that attacker  $f$  neutralized authenticator  $X$ .<sup>13</sup> Instead, when  $f(X) = Y$  and  $X \neq Y \neq \mathbf{1}$  we say that  $f$  partially compromises  $X$ .<sup>14</sup> Finally, when

<sup>13</sup>Also we require that  $f(\mathbf{1}) = \mathbf{1}$  for every  $f$ .

<sup>14</sup>For the sake of presentation, we write  $F$  (with  $F \in \{O, K, I\}$ ) when an attacker compromises a multi-factor authenticator by reducing it to the very same authenticator but for the elimination of  $F$ .

$f(X) = X$  we say that  $f$  does not affect  $X$ . Given an MFA protocol  $S = X_1; \dots; X_n$  we define  $f(S) = f(X_1); \dots; f(X_n)$  and we say that  $f$  neutralizes  $S$  whenever  $f(S) = \text{skull}$ .

Note that, under our assumptions, the applicability of an attacker model to an MFA protocol does not automatically result in an actual threat. In fact, our attacker models represent the set of resources and capabilities that an adversary should have to interact with the elements of an MFA protocol. Reasonably, a threat analysis should consider the applicable attacker models to check whether they can effectively authenticate instead of the user. Such threat analysis is beyond the scope of this work.

Next, we present the attacker models that we consider in our evaluation process. For each of them, we provide a brief explanation in natural language as well as a formal definition (right-hand side). When an authenticator  $a \in A$  does not appear among the inputs of a function  $f$ , we intend that  $f(a) = a$ , i.e., the attacker corresponding to  $f$  does not affect  $a$ .

**DT.** A *Device Thief* has the ability to steal a physical object. More precisely, DT targets authenticators relying on ownership factors. If an ownership-based authenticator is single-factor, DT can effectively neutralize it. This is the case for  $\text{grid}$ ,  $\cdot \gg \text{grid}[O] \gg \cdot$ ,  $\cdot \gg \text{grid}^?[O] \gg \cdot$ ,  $\cdot \gg \text{lock}[O] \gg \cdot$  and  $\cdot \gg \text{lock}^?[O] \gg \cdot$ . Notice that DT also affects out-of-band authenticators. In particular, DT neutralizes  $\text{lock}$ , i.e., a special case of  $\cdot \gg \text{grid}[O] \gg \cdot$ , and applies to  $\text{lock}^?$ , which behaves as  $\text{lock}$ .

Nevertheless, DT affects neither knowledge (K) nor inheritance (I) factors. Thus, authenticator devices and software relying on an ownership factor (together with some other factors) are affected by DT only partially ( $\emptyset$ ). For instance,  $\text{DT}(\text{opid} \gg_h \text{grid}[O, K] \gg_h \text{otp}) = \text{opid} \gg_h \text{grid}[K] \gg_h \text{otp}$ .

**AD.** An *Authenticator Duplicator* makes a copy of an authenticator. We assume authenticator devices  $\text{grid}$  to be duplication-proof by construction (as most devices include some secure hardware element). As for DT, AD cannot compromise knowledge and inheritance factors.<sup>15</sup> Hence, AD can neutralize soft-

$X$	$\text{DT}(X)$
$\text{grid}$	$\text{skull}$
$\cdot \gg \text{grid}[O] \gg \cdot$	$\text{skull}$
$\cdot \gg \text{grid}[O, \dots] \gg \cdot$	$\emptyset$
$\cdot \gg \text{grid}^?[O] \gg \cdot$	$\text{skull}$
$\cdot \gg \text{grid}^?[O, \dots] \gg \cdot$	$\emptyset$
$\cdot \gg \text{lock}[O] \gg \cdot$	$\text{skull}$
$\cdot \gg \text{lock}[O, \dots] \gg \cdot$	$\emptyset$
$\cdot \gg \text{lock}^?[O] \gg \cdot$	$\text{skull}$
$\cdot \gg \text{lock}^?[O, \dots] \gg \cdot$	$\emptyset$

$X$	$\text{AD}(X)$
$\text{grid}$	$\text{skull}$
$\cdot \gg \text{lock}[O] \gg \cdot$	$\text{skull}$
$\cdot \gg \text{lock}[O, \dots] \gg \cdot$	$\emptyset$
$\cdot \gg \text{lock}^?[O] \gg \cdot$	$\text{skull}$
$\cdot \gg \text{lock}^?[O, \dots] \gg \cdot$	$\emptyset$




ware authenticators only attesting ownership factors  $\mathcal{O}[O]$  and  $\mathcal{K}$ . Moreover, it can partially compromise ownership factor of multi-factor software authenticators, e.g.,  $\mathcal{O}[O,K]$ . Notice that the same rules apply to out-of-band software authenticators, i.e.,  $\mathcal{O}$ .

**SS.** A *Shoulder Surfer* (defined in [13] as a sub-case of the eavesdropper attacker) targets authenticators relying on a knowledge factor. Trivially, SS neutralizes  $\mathcal{K}$  and  $\mathcal{K}$ . SS is also effective against single-factor, knowledge based authenticators, e.g.,  $\cdot \gg \mathcal{K}[K] \gg \cdot$  and  $\cdot \gg \mathcal{O}[K] \gg \cdot$ . Moreover, SS can partially compromise multi-factor authenticators relying on a knowledge factor by removing it ( $\mathcal{K}$ ). Clearly, this also includes out-of-band software, i.e.,  $\mathcal{O}$ , since they are specific sub-case. For example,  $SS(\text{opid} \gg_n \mathcal{O}^?[O,K] \gg_h \text{otp}) = \text{opid} \gg_n \mathcal{O}^?[O] \gg_h \text{otp}$ .

SS can also neutralize authenticators that generate an otp when (i) the generated otp is not specifically bounded to an operation (e.g., it is not generated from an opid) and (ii) the otp has to be manually copied by the user. The latter happens with  $\mathcal{K}[\dots] \gg_h \text{otp}$  and  $\mathcal{O}[\dots] \gg_h \text{otp}$ .






$X$	$SS(X)$
$\mathcal{K}$	$\mathcal{K}$
$\mathcal{K}$	$\mathcal{K}$
$\mathcal{K}[\dots] \gg_h \text{otp}$	$\mathcal{K}$
$\cdot \gg \mathcal{K}[K] \gg \cdot$	$\mathcal{K}$
$\cdot \gg \mathcal{K}[K, \dots] \gg \cdot$	$\mathcal{K}$
$\cdot \gg \mathcal{K}^?[K] \gg \cdot$	$\mathcal{K}$
$\cdot \gg \mathcal{K}^?[K, \dots] \gg \cdot$	$\mathcal{K}$
$\mathcal{O}[\dots] \gg_h \text{otp}$	$\mathcal{O}$
$\cdot \gg \mathcal{O}[K] \gg \cdot$	$\mathcal{O}$
$\cdot \gg \mathcal{O}[K, \dots] \gg \cdot$	$\mathcal{K}$
$\cdot \gg \mathcal{O}^?[K] \gg \cdot$	$\mathcal{O}$
$\cdot \gg \mathcal{O}^?[K, \dots] \gg \cdot$	$\mathcal{K}$







<sup>15</sup>Note that the NIST [13] assumes that AD can also clone a memorized secret since the user might have annotated it on paper. Here we neglect this case as it would reduce a memorized secret to a look-up secret.



**ES.** An *Eavesdropping Software* is a program that runs on the user endpoint and communicates with the attacker. For instance, this category includes key loggers and spywares. ES can read, but not modify, data exchanged between the user and her endpoint. Thus, ES neutralizes ,  and  (since a software with enough privileges can interact with the telephony network APIs).







Moreover, as for SS, ES can neutralize authenticators taking no input and providing the user with an `otp` (that must be inserted into the endpoint). Finally, ES can compromise the knowledge factor of software authenticators that are executed on the endpoint.


$X$	$ES(X)$
	
	
 [ $\dots$ ] $\gg_h \text{otp}$	
 [ $\dots$ ] $\gg_h \text{otp}$	
$\cdot \gg \text{[K]} \gg \cdot$	
$\cdot \gg \text{[K, \dots]} \gg \cdot$	$\mathcal{K}^*$
$\cdot \gg \text{[K]} \gg \cdot$	
$\cdot \gg \text{[K, \dots]} \gg \cdot$	$\mathcal{K}^*$
*  on the endpoint.	

**SE.** A *Social Engineer* exploits some typical human behavior for inducing the user to carry out some operation. For instance, the user can be fooled to accept a tampered input or to reveal confidential data. However, we assume that SE becomes ineffective when the user is aware of the ongoing operation, i.e., when an authenticator is labeled with . Therefore, SE neutralizes , ,  $\cdot \gg \text{[K]} \gg \cdot$  and  $\cdot \gg \text{[K, \dots]} \gg \cdot$  (including the sub-cases for  and .

$X$	$SE(X)$
	
	
$\cdot \gg \text{[K]} \gg \cdot$	
$\cdot \gg \text{[K, \dots]} \gg \cdot$	

**MB.** The *Man in the Browser*<sup>16</sup> has full control on the web browser of the user, i.e., the endpoint of the Internet payments (IP). Any piece of data displayed and typed by the user can be intercepted and modified. Therefore, MB neutralizes , ,  $\cdot \gg \text{[K]} \gg \cdot$  and  $\cdot \gg \text{[K, \dots]} \gg \cdot$  when they are in the scope of an IP.

$X$	$MB(X)$
	
	
$\cdot \gg \text{[K]} \gg \cdot$	
$\cdot \gg \text{[K, \dots]} \gg \cdot$	
*Only applies to IP.	

Nevertheless, MB does not affect authenticators that show the ongoing operation () since they allow the user to revise the authenticator input and block the execution of the MFA protocol when necessary.

<sup>16</sup>Here we distinguish between MB and MM (see below) as a refinement of the generic *endpoint compromiser* defined in [13].

**MM.** A *Man in the Mobile* controls the mobile device of the user. This happens when the user operates through a compromised device, e.g., due to a malware. When the mobile device is the endpoint (MP) of the MFA protocol, MM neutralizes the same authenticators as MB, i.e.,  $\mathcal{A}$ ,  $\mathcal{B}$ ,  $\cdot \gg \mathcal{B}[\dots] \gg \cdot$  and  $\cdot \gg \mathcal{B}^?[\dots] \gg \cdot$ .

Furthermore, MM neutralizes software authenticators that run on the compromised device even when running an IP.

Thus, MM neutralizes both  $\cdot \gg \mathcal{B}[\dots] \gg \cdot$  and  $\cdot \gg \mathcal{B}^?[\dots] \gg \cdot$  if they run on a mobile device, e.g., a smartphone.

$X$	$MM(X)$
$\mathcal{A}$	$\mathcal{A}^*$
$\mathcal{B}$	$\mathcal{B}^*$
$\cdot \gg \mathcal{B}[\dots] \gg \cdot$	$\mathcal{B}^*$
$\cdot \gg \mathcal{B}^?[\dots] \gg \cdot$	$\mathcal{B}^{\dagger}$
$\cdot \gg \mathcal{B}^?[\dots] \gg \cdot$	$\mathcal{B}^{\dagger}$

\*Only applies to MP.  
 $\dagger$  on mobile.

In our analysis (Section 5.3), we consider both single attackers and their combination. Specifically, we combine the attacker models presented above by aggregating the respective capabilities. We indicate such combination with symbol  $\circ$ , e.g.,  $AD \circ SS$  denotes the combination of Authenticator Duplicator and Shoulder Surfer. To exemplify such attacker models, consider the MFA protocol  $S = \mathcal{A}; \text{opid} \gg_h \mathcal{B}[O, K] \gg_h \text{otp}$ . It is neutralized by  $SS \circ DT$ . As a matter of fact,  $SS \circ DT(S) = DT(SS(S)) = DT(\mathcal{A}; \text{opid} \gg_h \mathcal{B}[O] \gg_h \text{otp}) = \mathcal{A}; \mathcal{B} = \mathcal{A}$ . The approach above is centered on single MFA protocols. For the evaluation of the security level offered by banks, we applied it to every MFA protocol they employ.

Note that here we neglect some of the attacker models of [13]. In particular, we omit offline cracking, side channel and the online guessing. This is because these threats depend on flaws in the implementation of the authenticators that we cannot evaluate with the information at our disposal.

#### 4.3.3. Complexity

A key aspect for the adoption of MFA protocols is their ease-to-use [4, 6, 21]. Indeed, the users might be discouraged to execute a cumbersome or complex protocol. Moreover, if a protocol is too complex and hard to follow, a user might incur in errors either spontaneous or, even worse, induced by an attacker.

A standard approach [40] for evaluating the usability of a system is to measure its *effectiveness* (i.e., the accuracy and completeness with which users achieve specified

goals), *efficiency* (i.e., the resources used in relation to the results achieved) and *satisfaction* (i.e., how the system meets the user expectations) related to reaching of a given goal. A number of studies [6, 21, 41] have applied those measures to analyze MFA protocols and, in general, solutions for digital authentication. For instance, Weir et al. [41] applied them for the analysis of two-factor authentication protocols where effectiveness was assessed by checking task completion records and usage of help, efficiency by counting the time needed to complete the authentication process and satisfaction by questioning users immediately after they authenticated. The same usability metrics were used in [21], where a broader scope of MFA protocols was investigated. Other studies [4, 5] focus on user satisfaction and, in general, perceived usability of MFA protocols for online banking. These studies apply the System Usability Scale (SUS) [42], which relies on a predefined questionnaire to provide a subjective measure of the usability perceived by users about a target system.

In our study, we focus on the efficiency and, in particular, on the complexity of MFA protocols as several studies [4, 5, 6] recognized the complexity of MFA protocols as a critical aspect in the adoption of those protocols. In particular, we define a metric to evaluate the complexity of an MFA protocol (i.e., how much a protocol is difficult to use) that measures the amount of “resources” necessary to execute an MFA protocol (e.g., remembering a password, bringing a device). We consider three main types of resources for our evaluation, namely *memory*, *(manual) operations* and *(extra) devices*. The first type is used to determine the number of knowledge factors used in a MFA protocol (i.e.,  $\mathcal{Q}$  and  $[K]$ ), the second the number of input/output operations that have to be carried out by the user ( $\gg_h$ ) and the third the number of (non general-purpose) devices that have to be carried by the user (i.e.,  $\boxplus$  and  $\boxtimes$ ). For example, consider the MFA protocol  $\mathcal{Q}; \text{opid} \gg_h \boxplus[\text{O}, K] \gg_h \text{otp}$ . Its memory, operations and devices values are 2, 2 and 1, respectively. The overall complexity value (hereafter called *complexity score*) is obtained as the sum of these three numbers, e.g., 5 in the previous example.

These criteria can be objectively measured using our dataset (see Section 4.2). In contrast, other usability metrics proposed in [40] either require subjective measures (e.g., customer satisfaction, perceived usability) or involve empirical studies on the field (i.e., the observation of the error rate to evaluate the effectiveness of an MFA protocol).

Despite being interesting, those measures are out of the scope of this work.

As for the resistance to attacker models, the complexity is evaluated for single MFA protocols. To evaluate a bank in terms of ease-of-use, we compute the average complexity scores of all MFA protocols employed by the bank.

#### *4.4. Correlations between compliance of MFA implementations (with requirements and best practices), robustness against security threats and complexity of the MFA protocols adopted by banks*

The features introduced in the previous sections focus on specific aspects concerning the adoption of MFA in the online banking sector. We hypothesize that these features might not be independent from each other. Thus, we investigate possible relationships between different features of each bank and related MFA protocols. In particular, we verify the following hypotheses:

**H1** *Are banks that offer complex MFA protocols more likely to adopt exemptions?* This hypothesis aims to understand the implementation choices of a bank in terms of usability. In particular, we hypothesize that banks offering complex MFA protocols are more prone to adopt exemptions in order to support step-up authentication (best practice **BP3**). Therefore, we investigate whether the adoption of exemptions by banks is related to the complexity of MFA protocols they offer to their users. The outcome will provide us with additional insights regarding the compliance of a bank with **BP3**.

To evaluate this hypothesis, we check if there exists a correlation between the level of exemption adopted by each bank and the complexity of the MFA protocols that it offers – calculated as the minimum complexity score of all MFA protocols offered by the bank. In order to compute the correlation, we use the Pearson’s correlation coefficient<sup>17</sup> as implemented by Weka [43]. We consider the hypothesis to be verified if the coefficient is greater than 0.5, which is considered the threshold for having a moderate correlation (as a rule of thumb [44]). We perform the Fisher’s exact test to verify the significance of the calculated correlation. This test allows determining if the null hypothesis can

---

<sup>17</sup>Pearson’s correlation coefficient is a measure of the linear correlation between two variables. It ranges between 1 and -1, where 1 is positive linear correlation, 0 is no correlation, and -1 is negative linear correlation.

be rejected, hence proving the significance of our findings. Specifically, if the obtained p-value<sup>18</sup> is lower than the significance level of 5%, we conclude that our results are statistically significant.

***H2** Does the compliance of an MFA protocol with security requirements and best practices make the protocol more resistant against attacker models (described in Section 4.3.2)?* This hypothesis aims to investigate the actual impact of the identified requirements and best practices on MFA protocol implementations. In particular, we expect that the compliance of security requirements and best practices improves the security level of an MFA protocol.

To verify this hypothesis, we check if there exists a correlation between the level of compliance of MFA protocols with requirements and best practices and their resistance to attacks. The resistance of an MFA protocol is computed as the number of single attacker models that can compromise the protocol. The choice of this measurement is due to the nature of the legal (security) requirements. As discussed in Section 3, these requirements have been introduced to force an attacker to adopt multiple techniques for compromising a protocol, hence aiming to limit the chances of single attacker models. Therefore, the lower the number of singleton attacker models a protocol is vulnerable to, the more resistant the protocol is considered.

The level of compliance is assessed with respect to the requirements and best practices concerning security aspects of MFA protocols. Specifically, we determine the compliance of each MFA protocol with requirements **RL3**, **RL4**, **RL5** and **RL6** and best practice **BP4**. Note that, although these requirements and best practices are defined per bank, they assess properties of MFA protocols, as explained in Section 4.3.

As for the previous hypothesis, we calculate the Pearson's coefficient of correlation between the number of requirements and best practices an MFA protocol meets and the number of singleton attacker models the protocol is vulnerable to. Also in this case, a coefficient higher than 0.5 indicates the existence of a correlation between the two variables. Moreover, we determine whether the correlation is statistically significant

---

<sup>18</sup>The p-value is the probability of obtaining a result at least as extreme, given that the null hypothesis is true.



using the Fisher’s exact test, expecting a p-value lower than the statistical level of 5%.

***H3** Does the use of complex MFA protocols imply more resistance against attacker models?* Banks might employ complex MFA protocols with the expectation that they are more resistant against attacks. Our hypothesis is that there does not exist any correlation between the complexity of an MFA protocol and its resistance to attacker models. By evaluating this hypothesis, we aim to understand if we can design MFA protocols with a low complexity that are resistant against attacks.

In order to evaluate our hypothesis, we check whether there is a lack of correlation between the complexity score of an MFA protocol and its resistance against the attacker models described in Section 4.3.2. The complexity of an MFA protocol is calculated using the metric described in Section 4.3.3. This metric allows comparing MFA protocols leveraging objective measures of the effort required by a user to execute them. As for *H2*, the resistance of an MFA protocol against attacker models is computed as the number of single attacker models that can compromise the MFA protocol. The lower is the number of attacker models able to compromise an MFA protocol, the more robust it can be considered. These two aspects assess the trade-off between ease of use and security risks associated to MFA protocols.

To assess the independence between the complexity score on an MFA protocol and the number of attacker models able to compromise it, we perform the Fisher’s exact test. If the obtained p-value is higher than the significance level (5%), the null hypothesis must be taken into account, hence acknowledging the independence of the two variables.

## **5. Results**

In this section, we present the results of our investigation. In particular, we answer to the research questions introduced in Section 4.1 by means of the data and criteria discussed in Section 4.2 and Section 4.3, respectively. Moreover, we verify whether the hypotheses presented in Section 4.4 hold.

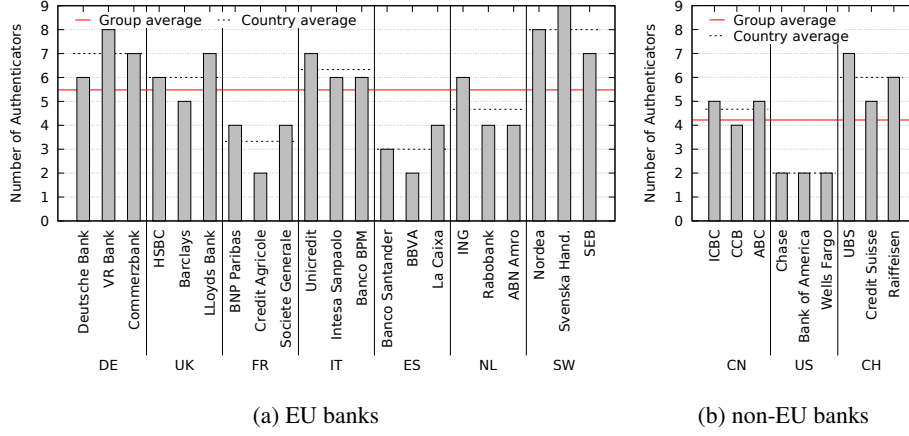


Figure 4: Number of authenticators provided by banks.

### 5.1. Answer to RQ1: Demographics

To characterize the landscape of how MFA has been adopted in the online banking sector, we consider several perspectives. Below we present some statistics on authenticators, MFA protocols, exemptions, and enrollment and binding procedures.

*Authenticators.* Figure 4 shows the number of authenticators that each bank offers to its customers. Vertical bars indicate the number of distinct authenticators per bank; horizontal lines indicate the average per nation (blue dashed line) and per group (red solid line). All banks employ a minimum of 2 and a maximum of 9 authenticators. At national level, values appear homogeneous. As a matter of fact, the maximum variation in the number of authenticators per nation is 2 and the average variation per nation is 1.7. This may indicate that some national trends exist. These trends may be the result of national laws, market strategies or adoption of national identity systems (e.g., BankID [45] for Swedish banks).

Although the number of employed authenticators provides an indication on the variability of the authenticators commonly offered by banks, we are also interested in their type (see Section 4.2.2). Figure 5 shows the distribution of authenticators per type, i.e., devices, software, look-up secrets and memorized secrets, adopted by EU and non-EU banks. White bars indicate the percentage of banks that employ at least one

authenticator of the given type. Moreover, for device and software authenticators, the figure shows the percentages with respect to sub-categories, i.e., single-factor, multi-factor and out-of-band. Note that, in order to differentiate the employment of credentials (username and password) and additional memorized secrets, we represent them in two separate columns, namely “credentials” and “2nd memorized secret”, respectively.

We observe some facts. All considered banks provide their users with credentials. Secondly, almost all of them (except one) offer at least one device authenticator. Moreover, EU banks offer 1.4 device authenticators on average, while the average number of device authenticators for non-EU banks is 1.9. Among device authenticators, out-of-band authenticators (i.e., SIM cards) are more common (48% for EU and 78% for non-EU banks). We can also observe that EU banks employ multi-factor device authenticators more frequently than single-factor device authenticators (38% and 29% of the banks, respectively), whereas non-EU banks do the opposite (with 67% and 33% of non-EU banks employing single and multi-factor device authenticators, respectively).

The adoption of other types of authenticators differs between the two bank groups. For EU banks, the second most frequent type of employed authenticators is software. As a matter of fact, 86% of EU banks adopt at least one authenticator of this type.<sup>19</sup> Among these banks, the average number of software authenticators is 3.1. Multi-factor ones are dominant (62%), although a significant number of out-of-band software authenticators are also employed (57%). Single-factor authenticators are less common (43%). Look-up secrets and extra memorized secrets follow in the order, being employed by 29% and 14% of EU banks, respectively.

On the other hand, non-EU banks present a different trend. After device authenticators, look-up secret is the second most adopted type of authenticator (44%). Software authenticators are employed by 33% of the banks. All banks adopting software authenticators provide at least one that is multi-factor. Only one non-EU bank provides also a single-factor software authenticator. Finally, only 11% of the banks in this group adopt additional memorized secrets.

---

<sup>19</sup>Note that every bank has an official mobile application, but here we only consider those that act as an authenticator.

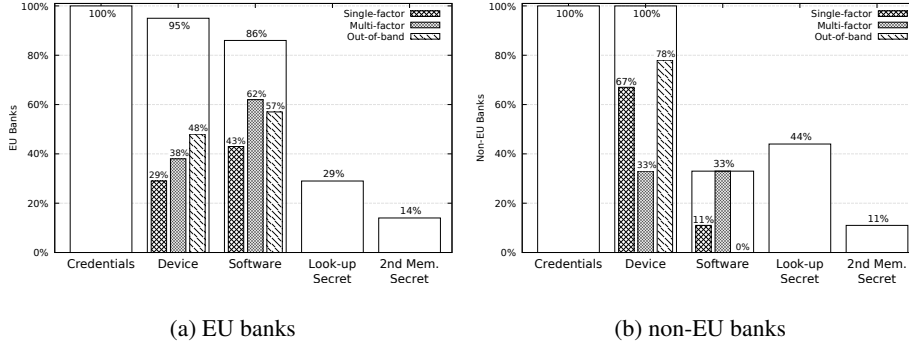


Figure 5: Percentage of authenticators types offered by banks.

*MFA protocols.* Our analysis reveals that banks usually provide their clients with a variety of MFA protocols. For each bank, we distinguish MFA protocols for IP and for MP. As mentioned in Section 4.2.2, these two kinds of payment methods differ for the endpoint on which they are executed. This distinction is necessary to evaluate the compliance with requirements and best practices as well as to analyze the security and complexity of MFA protocols (see following sections). Figure 6 shows the number of MFA protocols adopted by each bank. The figure also reports the average number of MFA protocols for IP (red lines) and MP (blue lines), for each nation (dashed lines) and for bank group (solid line). Overall, we counted 32 distinct MFA protocols employed by banks for IP and 29 protocols for MP.

We observe a few facts. Except for two banks, the number of MFA protocols for IP is equal or greater than the one for MP. In particular, four banks do not support MFA protocols for MP at all.<sup>20</sup> As expected, we observe a correspondence between the number of authenticators (Figure 4) and the number of MFA protocols per bank. As a matter of fact, most authenticators are associated to a limited number of MFA protocols (often only one). Moreover, most of the MFA protocols (around 80%) rely on two authenticators.

To better understand the differences between the adopted MFA protocols, we investigated how many (and which kind of) AFs are used by them. Figure 7 shows the result

<sup>20</sup>Note that the absence of MFA protocols does not imply that MP is not supported. In fact, Barclays, Credit Agricole and all American banks do support MP – even though only low risk operations are supported.

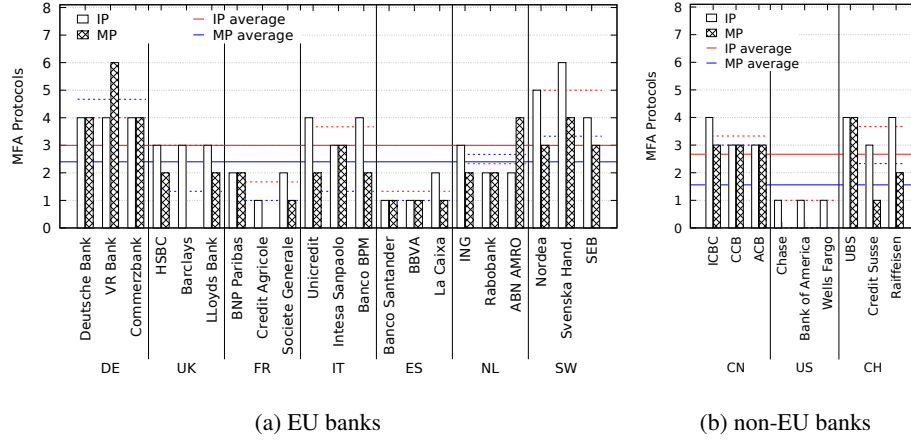


Figure 6: Number of MFA protocols supported by banks.

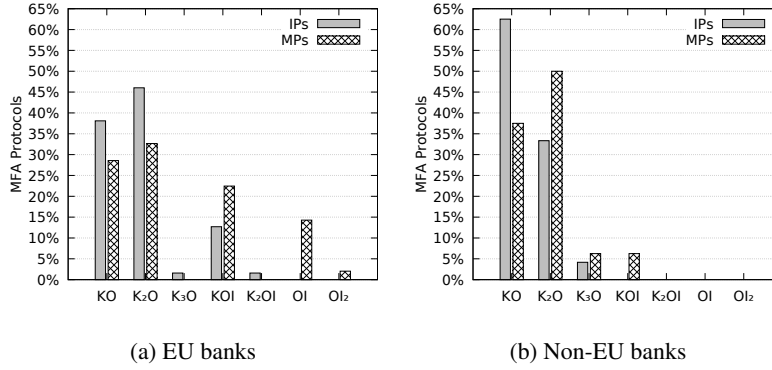


Figure 7: Combinations of AFs used in MFA protocols.

of this analysis, displaying the combinations of AFs that the analyzed MFA protocols employ. Each combination can be composed by **K**nowledge, **O**wnership and **I**nherence factors. If more than one factor of the same type is used, the number of factors is reported as a subscript for the corresponding type. For example, if an MFA protocol leverages a memorized secret and a multi-factor authenticator device attesting both ownership and knowledge factors, such an MFA protocol is annotated with combination  $K_2O$ .

Figure 7 shows that a large number of MFA protocols leverage more than two AFs (the minimum for an MFA protocol) for authenticating the user. This fact is particularly noticeable for EU banks, where 52% of the employed MFA protocols for IP and 59% of



those for MP leverage at least three AFs. A slightly different situation can be observed for non-EU banks. In this group, only 37% of the MFA protocols for IP rely on more than two AFs. In the case of MP, however, the trend is opposite, with 62% of the MFA protocols employing at least three AFs.

We also observe that, both for IP and MP, combinations involving knowledge and ownership factors are the most frequent. The employment of inference factors is more frequent in MFA protocols for MP employed by EU banks. In particular, 38% of these protocols employ at least one inference factor and around 15% of them leverage only inference and ownership factors. Interestingly, this type of AF is usually not employed by non-EU banks, where only 6% of the MFA protocols adopted by those banks leverage an inference factor. Finally, no combinations constituted only by knowledge and inference factors have been observed.

*Exemptions.* The adoption of exemptions can influence both the security level and perceived ease-of-use of MFA protocols. To this end, we investigated the type of exemptions allowed by each bank. The consequent evaluation in terms of ease-of-use (**BP3**) will be discussed in the following sections.

The type of exemption for every bank is reported in Table 8. We can observe from the table that exemptions are widely adopted. As a matter of fact, 27 of the considered banks adopt some form of exemption. The adopted level of exemptions seems to be homogeneous for each country. The only 3 banks that do not support exemptions are located in two countries, i.e., Sweden and Switzerland.

*Enrollment.* The enrollment phase plays a critical role in MFA. The analysis of the offered modalities allows to understand at which level of security the verification of user identity is performed. This information will be used in the next section to assess the compliance of banks with **RL7**, **RL8** and **RL9**.

The enrollment modalities offered by banks are reported in Table 8. We observed that every bank allows enrollment at the bank. In the table, symbol  indicates the possibility for the user to choose between remote or *de visu* enrollment, whereas symbol  indicates that only the second option is available for a given bank. We can observe

	Deutsche bank VR Bank Commerzbank	HSBC Barclays Lloyds bank	BNP Paribas Credit Agricole Société Générale	Unicredit Intesa Sanpaolo Banco BPM	Banco Santander BBVA La Caixa	ING Rabobank ABN AMRO	Nordea Svenska Handelsb. SEB	ICBC CCB ABC	Chase Bank Of America Wells Fargo	UBS Credit Suisse Raiffeisen
Country	DE	UK	FR	IT	ES	NL	SW	CN	US	CH
Exemptions	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ✓	✓ × ×	✓ ✓ ✓	✓ ✓ ✓	✓ ✓ ×
Enrollment	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘
Binding	Request	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘
	Delivery	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘ ⌘
	Activation	⌘* ⌘* ⌘*	⌘ ⌘ ⌘	⌘* ⌘* ⌘*	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘* ⌘* ⌘*	⌘ ⌘ ⌘	⌘ ⌘ ⌘	⌘ ⌘*

Table 8: Exemptions, Enrollment and Binding procedures per bank.

that remote enrollment is fairly common. Indeed, out of 30 banks, 18 allow remote enrollment. Also in this case, values appear homogeneous at a national level.

*Binding.* The binding of an authenticator to a user’s identity can influence the security of MFA protocols leveraging that authenticator. Here, we analyze the modalities offered by banks for binding, which will allow us to evaluate the compliance of banks with requirements **RL8** and **RL9**.

Table 8 presents the worst case (in terms of security) of the binding procedures offered by every bank. Further details on the enrollment and binding procedures adopted by all banks are given in the supplementary material. Intuitively, the analysis of the worst case provides an indicator of the compliance of banks with **RL8** and **RL9** and, in particular, the resistance of a procedure to attacks: if even in the worst case an attacker is not able to compromise an authenticator, the others will be reasonably secure.

From the table, we can observe that the remote request of authenticators is massively supported. A similar trend can also be observed for the delivery of authenticators. The only exception is represented by Chinese banks in which all binding operations – request, delivery and activation – are performed at the bank. The majority of banks also allow a remote activation of authenticators, but using a weak procedure. Only 12 banks (among the 30 considered) ensure an adequate level of security for the activation of authenticators (either requiring clients to activate them at the bank or through MFA leveraging previously activated authenticators). 6 of these banks would actually offer

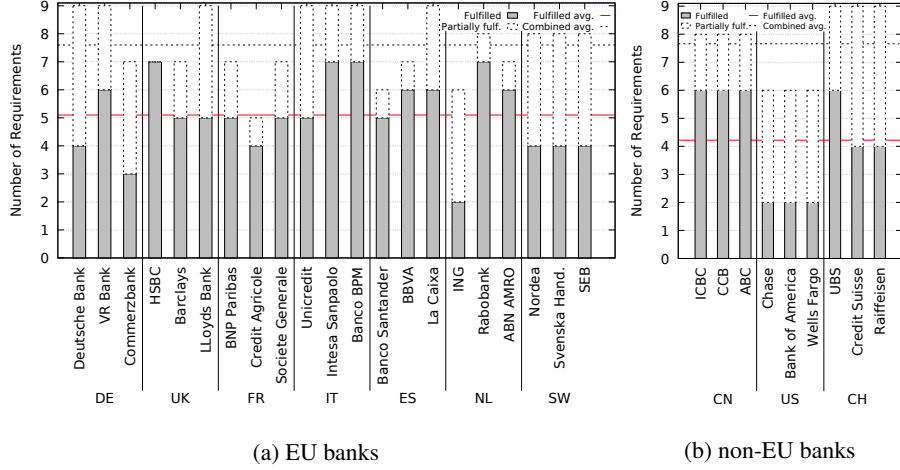


Figure 8: Fully and partial fulfilled requirements per bank.

an activation step leveraging an MFA protocol, but the employed authenticators have not been bound with a sufficient security level. In Table 8, this is marked with 🌐\*. An example is the binding procedures offered by Commerzbank and BNP Paribas: those banks offer the possibility to activate a software authenticator through an MFA protocol based on the reception of SMS on an out-of-band device. However, the binding of the out-of-band device can be performed remotely, hence not complying with **RL8** and lowering the security level of the binding procedures relying on it.

## 5.2. Answer to RQ2: Compliance with requirements and best practices

In this section we discuss the compliance of banks with the requirements and best practices presented in Section 3 based on the criteria presented in Section 4.3. A complete view of the analysis is reported in the supplementary material.

**Requirements.** Figure 8 shows, for each bank, the number of fulfilled (solid bar) and partially fulfilled (dashed bar) requirements. The average number of fulfilled requirements is represented by a solid (red) line and the average number of fulfilled and partially fulfilled requirements by dashed (blue) line.

We observe that none of the considered banks does meet all nine identified requirements. For EU banks, the average number of requirements fulfilled by banks is 5.1. If



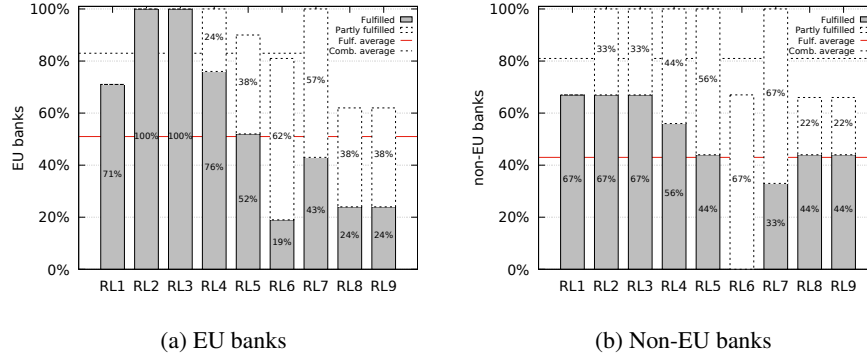


Figure 9: Percentage of banks that fully and partially comply with the requirements.

we combine both fulfilled and partially fulfilled requirements, the average increases to 7.6. The number of fulfilled requirements appears to be homogeneous among the banks of the same country (average variation is 2). The maximum variation is more than 2 only for Dutch and German banks (5 and 3, respectively). However, we observed some differences in the level of compliance for different countries. Three countries – Germany, France and Sweden – have an average of fulfillment below 4.6. On the contrary, Italian banks adhere to regulations and directives more than others, with an average of 6.3 fulfilled requirements.

For non-EU banks, the average number of fulfilled requirements is 4.2. This is not surprising, since our survey focused on requirements derived from EU regulations and directives. However, a deep look at the data showed that this value is strongly influenced by the low number of requirements met by US banks, which fulfill only two requirements. On the contrary, all Chinese banks comply with six requirements, which is higher than the average number of requirements met by EU banks. This means that, even if they are not subject to the same regulations and directives as EU banks, Chinese banks are aligned to EU security requirements. Similar observations apply to Swiss banks. Even if the average number of requirements met by these banks is not as high as the one met by Chinese banks, it matches the average number (5.1) met by EU banks.

We now analyze the compliance of banks with single requirements. Figure 9 shows the percentage of banks that fulfill each requirement. For each requirement, the gray

bar indicates the percentage of banks that fulfill the requirement and the dashed bar the percentage of banks that partially fulfill it.

**RL1**, which concerns integrity checks on multi-purpose devices, is met by 71% of EU banks. In particular, none of the Swedish banks meet this requirement, along with two Dutch and one French bank. However, it is worth noting that this requirement will enter into force in the first half of 2019 [9] and, thus, EU banks do not have to comply with it yet. For what concerns non-EU banks, 67% of them comply with **RL1**. Interestingly, all US banks comply with this requirement.

**RL2**, which requires the employment of MFA for risky operations, is fulfilled by all EU banks. This meets our expectations, since a first definition of this requirement [9] was introduced in 2014. This requirement is also met by all Chinese and Swiss banks, but by none of the US banks.

**RL3** and **RL4** concern the usage of distinct and independent authentication factors in MFA protocols, respectively. While the first one is fulfilled by all EU banks, the second is only met by the 76% of them. The remaining 24% of EU banks, however, partially fulfill **RL4**, since they offer at least one MFA protocol employing two authenticators. This is due to the fact that some EU banks employ MFA protocols that only leverage a mobile application attesting both inherence and ownership factors (hence not leveraging independent AFs). Similarly to **RL1**, both these requirements were introduced in [9], which will enter into force in 2019. However, the high level of compliance with **RL3** and **RL4** might indicate that EU banks have already taken actions to adhere to this regulation. The percentage of non-EU banks that comply with **RL3** and **RL4** is 67% and 56%, respectively.

**RL5**, which requires the generation of a unique authentication code in every MFA execution, is fulfilled by 52% of EU banks and partially fulfilled by 38% of them. This can be explained by the fact that half of the banks offer a large range of heterogeneous MFA protocols where at least one does not employ an otp generated using an opid. It is worth mentioning that the fulfillment of this requirement will become mandatory only in 2019. For non-EU banks, instead, we have 44% of them complying with **RL5**, while the others (56%) partially fulfill it.

The level of compliance with **RL6** is the lowest, when compared to those of other

requirements. Indeed, this requirements is met by 19% of EU banks and partially fulfilled by 62% of them. In words, this means that the majority of the banks (81%) employ at least one MFA protocol that does not inform the user about what operation she is authorizing. Similarly to **RL5**, the compliance with this requirement will become mandatory in 2019. It is worth noting that none of the non-EU banks comply with **RL6**, while 67% of them partially fulfill it.

The fulfillment of **RL7**, which concerns user enrollment, strongly reflects the results presented in Section 5.1. Being influenced by enrollment modalities, **RL7** is fulfilled by those banks only providing enrollment in their branches (43% and 33% of EU and non-EU banks, respectively). The other banks fulfill **RL7** only partially.

Finally, we analyze requirements **RL8** and **RL9**. Recall from Section 3.1 that **RL8** concerns the level of security of the binding phases, whereas **RL9** concerns the activation of remotely delivered authenticators. These requirements exhibit the same level of compliance: 24% of EU banks fulfill these requirements. In several cases, banks employ authentication protocols leveraging multiple factors for activating an authenticator; however, the binding of these authenticators was performed without a proper security level, causing the requirements not to be fulfilled. Similarly to other requirements, **RL8** and **RL9** will enter into force in 2019. Among non-EU banks, 44% of them comply with both **RL8** and **RL9**, whereas 22% partially fulfill them.

To summarize, EU banks comply, on average, with half of the considered requirements. This may be due to the fact that five of them are specified on directives (and regulatory standards) entering into force only in 2019. Indeed, the percentage of EU banks complying with **RL1**, **RL5**, **RL8**, **RL9** and especially with **RL6** is low. However, a large number of EU banks offers at least one MFA protocol that meets all criteria defined in Table 5, thus partially fulfilling these requirements. Therefore, the majority of the banks can easily become compliant with these requirements just by offering a subset of the MFA protocols they currently support. When considering both the fulfillment and partial fulfillment of **RL5** and **RL6**, more than two-third of EU-banks comply with these requirements. If we assume that all requirements that are currently partially fulfilled will be fully met by 2019, EU banks will comply, on average, with more than 7 requirements

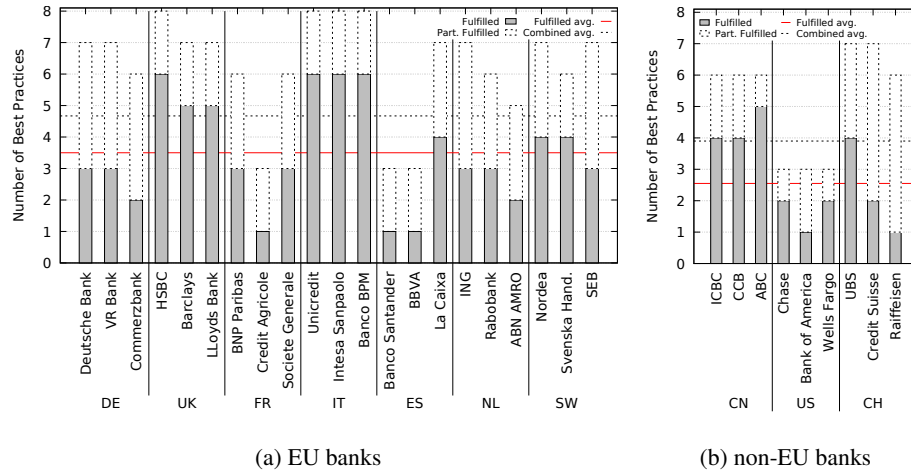


Figure 10: Fully and partial fulfilled best practices per bank.

out of 9.

*Best practices.* We also analyzed the compliance of banks with best practices. Figure 10 shows the result of our analysis. For each bank, the number of fulfilled best practices is represented by a solid bar and the number of partially fulfilled best practices by a dashed bar. In the figure, we also report the average number of fulfilled best practices (solid line) and the average number of fulfilled and partially fulfilled best practices (dashed line).

As for the requirements, no bank fulfills all eight identified best practices. The average number of best practices fulfilled by a bank is 3.5 and 2.5 for EU and non-EU banks, respectively. If we consider both fulfilled and partially fulfilled best practices, the average number is 4.6 and 3.9, respectively.

Among EU banks, Spain is the country in which banks met less best practices, with an average of two best practices. On the other hand, Italian banks lead also in terms of fulfilled best practices, with an average of 6 best practices. Among non-EU countries, the results for best practices are similar to the ones concerning requirements. US banks fulfill an average of one best practice, whereas Chinese banks an average of 4.3 best practices. The position of Swiss banks is quite heterogeneous with a variation in the number of fulfilled best practices equal to 3 (the average number of fulfilled best

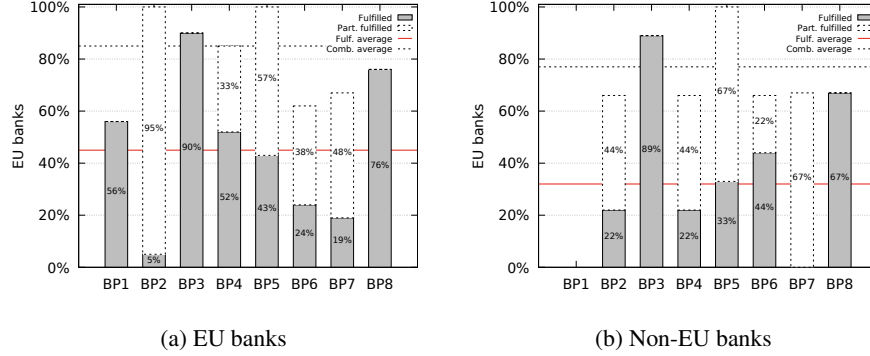


Figure 11: Percentage of banks that fully and partially comply with the best practices.

practices is 2.3).

To gain more insights, we analyzed the compliance of banks per best practice. Figure 11 shows the percentages of banks that fulfill and partially fulfill each best practice. From the figure, we observe that the percentage of banks that met the best practices is lower than the one of banks that met the requirements (45% compared to 51% for EU banks and 32% compared to 43% for non-EU banks). However, if we consider both the fulfillment and partial fulfillment of requirements and best practices, the percentage of banks are aligned (85% vs. 83% and 77% vs. 81%, for EU and non-EU banks, respectively).

We now discuss the compliance with each best practice against the fulfillment criteria in Section 4.3.<sup>21</sup> **BP1**, which concerns the integration of software authenticators in mobile banking applications, is fulfilled by 56% of EU banks. On the other hand, 67% of non-EU banks fulfill this best practice because they do not offer any software authenticator. The software authenticators offered by the remaining 33% (i.e., the Swiss banks) are instead not integrated with the respective mobile banking applications, thus violating **BP1**.

**BP2** concerns the usage of commonly used libraries to execute security-relevant operations. The fulfillment of this best practice is 5% for EU and 22% for non-EU banks.

<sup>21</sup>Recall that **BP5** and **BP6** are subsumed by **RL7** and **RL8**. In the following, the same observations apply to both the requirements and the best practices.

It is worth noting that all Chinese banks do not comply with this best practice. This may be due to the fact that the majority of services on which Android applications rely on (e.g., Google Play Services, Firebase) are blocked by the Chinese firewall [46, 47]. The least used APIs are those concerning the integrity checks and keystore, while those related to SSL are always implemented. Moreover, we observed that several software authenticators use commercial solutions instead of those we considered in Section 4.3. We will discuss this aspect in Section 6.

**BP3**, which concerns the adoption of step-up authentication mechanisms, is fulfilled by 90% and 89% of EU and non-EU banks. Our analysis revealed that almost all banks employ some form of exemption (see Table 8), thus supporting step-up authentication.

**BP4** concerns the usage of SMS messages in MFA protocols. 52% of EU banks do not employ any MFA protocol leveraging SMS messages, whereas 15% of the same group of banks employs only MFA protocols relying on them. For what concerns non-EU banks, only 22% of them employ MFA protocols that do not use SMS messages while 33% of them (specifically, US banks) employ only MFA protocols relying on SMS.

**BP7** concerns the binding of two physical authenticators immediately after the enrollment. This best practice is fulfilled by 14% of EU banks and never fulfilled by non-EU banks. However, 48% and 67% of EU and non-EU banks (respectively) partially fulfill **BP7**. It is also worth noting that 33% of both EU and non-EU bank violates the best practice, not offering the user any physical authenticator immediately after her enrollment. Finally, **BP8** concerns the availability of multiple types of authenticators. This best practice is fulfilled by 76% of EU banks and 67% of non-EU banks. It is worth noting that all non-EU banks that violate this best practice are US banks.

Globally, we observe that the considered best practices are fulfilled, on average, by more than a half of the EU banks. For non-EU banks the level of compliance is lower. The most violated best practices are **BP2** and **BP7**. The first one is rarely fulfilled because almost every application released by banks relies on proprietary or commercial solutions, rather than the APIs we identified in Section 4.3. In the case of **BP7**, the lack of fulfillment is due to the fact that, if two physical authenticators are offered, one of the two is usually given upon request and payment of a little sum of money. On the other

hand, the most fulfilled best practices are those related to the perceived ease-of-use of the digital authentication, namely **BP3** and **BP8**. Indeed, as seen in Section 5.1, the majority of the banks employs both exemptions and a high variety of authenticators.

### 5.3. Answer to RQ3: Resistance to attacker models

In this section, we discuss how the MFA protocols adopted by banks behave with respect to the attacker models described in Section 4.3.2. Here, we provide an overview of the results and only report when an MFA protocol can be successfully compromised by one of the attacker models individually or only by their combination. We refer to the supplementary material for a detailed evaluation of MFA protocols (e.g., which and how many attackers that can compromise a protocol by acting individually).

Figure 12 shows, for each bank, the percentage of MFA protocols for IP that can be compromised by single attacker models and their combinations (composed either by two or three attacker models), whereas Figure 13 shows the results for MFA protocols for MP. The percentage of MFA protocols that are vulnerable to attacker models acting individually is represented by solid gray boxes, whereas the percentage of MFA protocols that are only vulnerable to attacker combinations is represented by white (two attacker models) or light blue (three attacker models) pattern-filled boxes with dashed lines. Trivially, MFA protocols that are vulnerable to single attackers, are also vulnerable to their combination with other attacker models. We refer to Section 5.1 for the number of MFA protocols for IP and MP offered by each bank.

We observe that 46% (on average) of MFA protocols for IP adopted by each EU bank are vulnerable to single attacker models. In particular, at least half of the MFA protocols offered by 10 EU banks are vulnerable to single attacker models. All MFA protocols offered by 5 banks (all English, one Spanish and one French) can be compromised by single attacker models. Non-EU banks offer an average of 62% of MFA protocols for IP that can be compromised to single attacker models. It is worth noting that all MFA protocols offered by all US banks and by one Swiss bank are vulnerable to single attacker models.

In the context of MP, the percentage of vulnerable protocols is higher. 85% (on average) of MFA protocols offered by each EU bank are vulnerable to single attacker models. Only 6 EU banks offer at least one MFA protocol that cannot be compromised

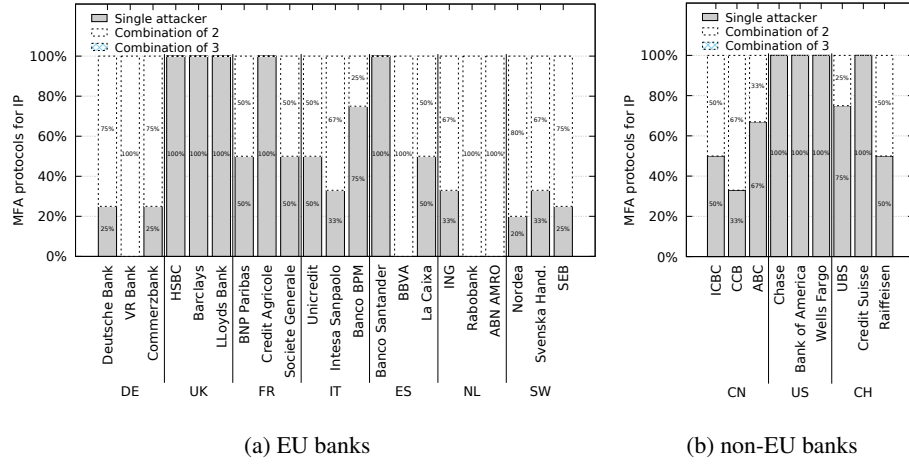


Figure 12: Percentages of MFA protocols for IP vulnerable to single or combined attackers.

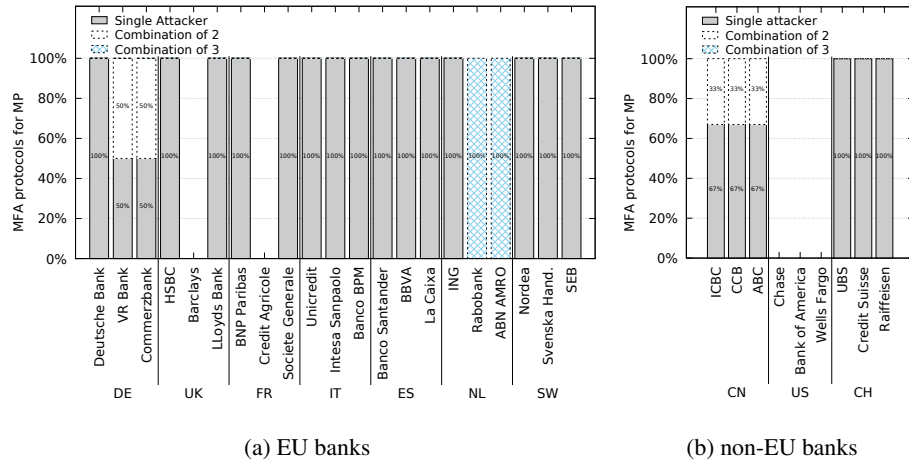


Figure 13: Percentages of MFA protocols for MP vulnerable to single or combined attackers.

by single attacker models, but only by their combination. It is worth noting that the missing box for Barclays and Credit Agricole is due to the fact that these banks do not provide any MFA protocol for MP (see Figure 6). On the other hand, 83% (on average) of MFA protocols offered by each non-EU bank are vulnerable to single attacker models. US banks do not provide any MFA protocol for MP, and only the Chinese banks offer at least one MFA protocol that cannot be compromised by single attacker models.



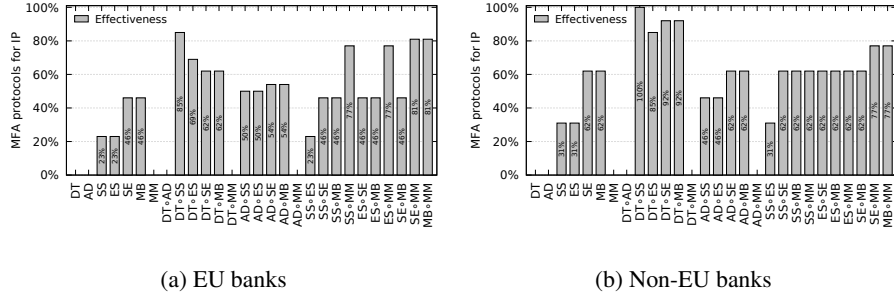


Figure 14: Percentage of MFA protocols for IP that are vulnerable to given attackers.

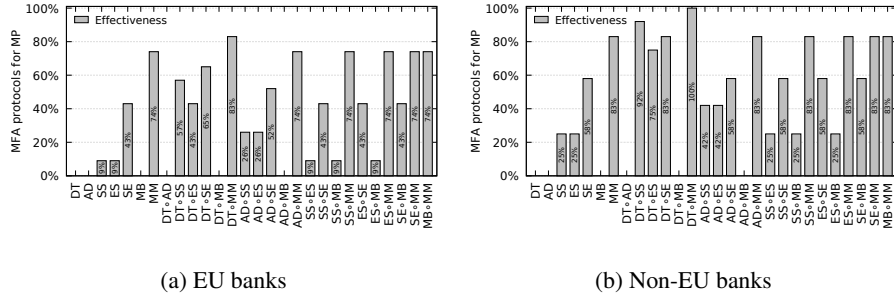


Figure 15: Percentage of MFA protocols for MP that are vulnerable to given attackers.

Interestingly, our analysis revealed that 97% of all MFA protocols (both for IP and MP) can be compromised by at least one combination of two attacker models. The other 3% (2 MFA protocols for MP offered by Rabobank and ABN Amro) require a combination of at least three attacker models to be compromised.

We now present an analysis of the effectiveness of different attacker models over the employed MFA protocols (see Section 4.3.2 for the details on each attacker model). Figure 14 and 15 show the percentage of MFA protocols (for IP and MP, respectively) that can be compromised by single attacker models or by combinations of 2 of them. The effectiveness of single and composed attacker models is represented by a gray box.

From the figures, we can observe that the most effective attacker models against MFA protocols for IP are Man in the Browser (MB) and Social Engineer (SE). When taken individually, these attacker models can compromise 48% and 67% of the MFA protocols for IP employed by EU and non-EU banks, respectively. On the contrary, Man in the

Mobile (MM), Device Thief (DT) and Authenticator Duplicator (AD) are never able to compromise any MFA protocol for IP by themselves. If we consider combinations of attacker models, the most effective combination of two attacker models on MFA protocols for IP employed by EU banks is constituted by DT and Shoulder Surfer (SS), being able to compromise 84% of these protocols. In the case of non-EU banks, the combined attacker model can potentially compromise all adopted MFA protocols for IP.

For what concerns MFA protocols for MP, Man in the Mobile (MM) is the most effective attacker model. Indeed, it can compromise – by itself – 74% and 83% of the MFA protocols for MP offered by EU and non-EU banks, respectively. The most effective combination of two attacker models is “DT $\circ$ MM”, managing to compromise 83% and 100% of the protocols offered by EU and non-EU banks, respectively. Also in the context of MP, DT and AD are not effective against any MFA protocol when acting by themselves. In particular, these attacker models compromise the ownership factors asserted by an authenticator, but they are not able to compromise knowledge factors, which are used in all analyzed MFA protocols.

We stress that there is no combination of two attacker models that is able to compromise all MFA protocols (either for IP or MP) offered by EU banks. The minimum number of attacker models required to achieve this result is 3. In particular, only a combination of DT, SS and MM, is able to compromise all MFA protocols (either for IP or MP) offered by EU banks.

We now analyze the effectiveness of combinations of attacker models. In particular, we assess the effectiveness “gain” that combinations of attacker models have with respect to the effectiveness of the attacker models that they include. The gain (represented with a box filled with gray pattern) is the percentage of protocols that can be compromised only by exploiting the capabilities of all attacker models in the combination. Moreover, the figures show the “inherited” percentage of effectiveness, i.e., the percentage of protocols compromised by the attackers in the combination when acting individually (represented with a white box with dashed line). In the case attackers can compromise the same MFA protocol, it will be considered only once in the computation of the inherited value.

Consider, for instance, combination “DT $\circ$ SS $\circ$ SE”. According to Figure 17a, this

combination has 78% of inherited effectiveness and 14% of gained effectiveness. The inherited effectiveness is obtained by considering the protocols that can be compromised by DT, SS or SE individually or by all combinations including two of these attacker models. If an MFA protocol is compromised by more than one of these (composite) attacker models, it is counted only once for assessing the inherited effectiveness. As shown in Figure 15a, “DT◦SS” has 57% of effectiveness, “DT◦SE” has 65% and “SS◦SE” has 43%. However, “DT◦SE” compromises all MFA protocols compromised by “SS◦SE” and the majority of those compromised by “DT◦SS”, managing to compromise only 23% additional MFA protocols in respect with “DT◦SS”. Therefore, the inherited effectiveness is 78%. The gained effectiveness, instead, derives from the number of MFA protocols that can only be compromised by DT, SS and SE acting together (i.e., that cannot be compromised by any of the two combinations). In this case, “DT◦SS◦SE” (acting together) can compromise 13% of the MFA protocols in addition to the 78% compromised by “DT◦SS” and “DT◦SE”, obtaining a total effectiveness of 91%.

Figures 16 and 17 present the results of our analysis. Note that combinations not having any gain with respect to the effectiveness of the attacker models that they include are not shown in the figures. We observe that the most effective combination (i.e., with higher gain) is “DT◦SS”. In general, these two plots show that the most effective combinations are those combining attacker models having different targets (in terms of authenticators, authenticator outputs and authentication factors). The “DT◦SS” combination, for example, includes DT – that targets ownership factors – and SS – targeting knowledge factors and manually copied `otps`). This result is not surprising, since MFA protocols should be designed in such a way that a potential attacker is required to execute multiple (and different) malicious actions in order to compromise the protocol.

To conclude, we observe that the least secure MFA protocols, both for IP and MP, are those employing authenticators generating an `otp` without receiving an `opid` as an input. Indeed, as shown in the supplementary material, these protocols are the most vulnerable ones in terms of resistance to singletons and to attackers combinations.

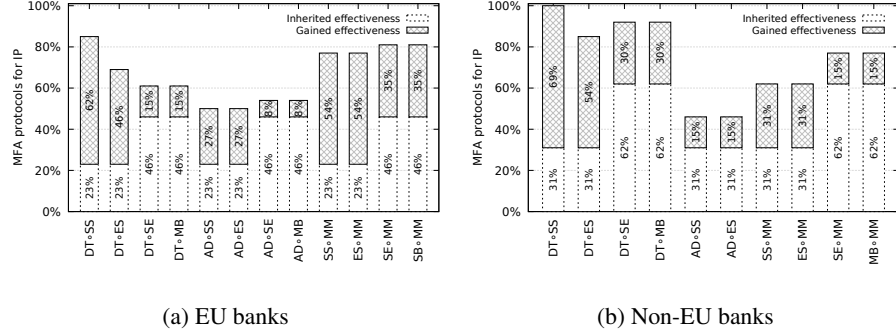


Figure 16: Percentage of MFA protocols for IP that are vulnerable to given attackers.

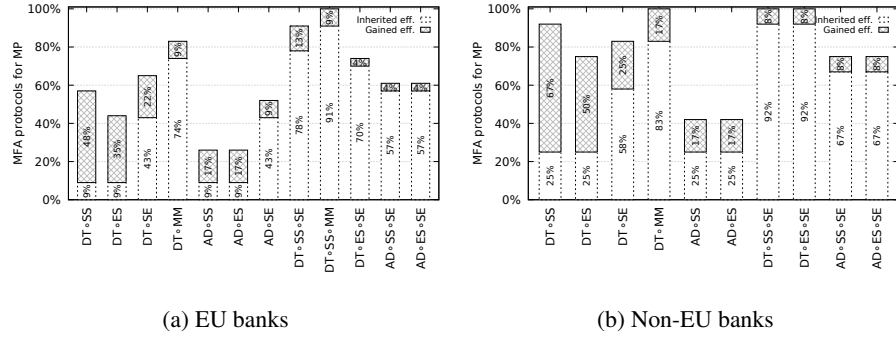


Figure 17: Percentage of MFA protocols for MP that are vulnerable to given attackers.

#### 5.4. Answer to RQ4: Complexity of MFA Protocols

In this section, we analyze the complexity of the MFA protocols adopted by banks against the criteria defined in Section 4.3.3. We first compute the complexity score<sup>22</sup> of the MFA protocols adopted by each bank and then investigate to what extent the various types of resources affect the overall complexity of MFA protocols. A detailed analysis of the complexity of the MFA protocols is given in the supplementary material.

Figure 18 shows the complexity score of the MFA protocols for IP and MP (represented by gray and pattern-filled boxes, respectively) employed by each bank. In the plots, we represent the average complexity score of the MFA protocols for IP and MP

<sup>22</sup>Recall from Section 4.3.3 that the complexity score of a MFA protocol is computed by summing up the amount of resources – memory, manual operations and extra devices – required by the protocol.

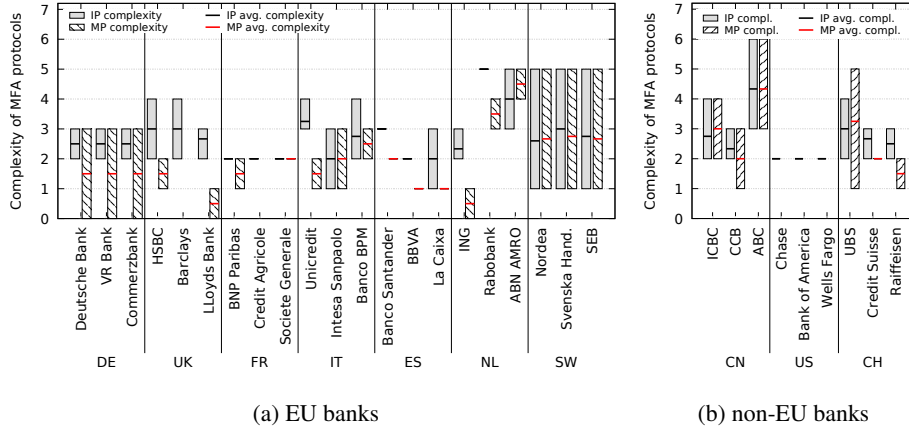


Figure 18: Complexity of MFA protocols adopted by banks.

employed by each bank using a black and red line respectively, where the average is computed over the number of MFA protocols employed by each bank as reported in Section 5.1. From the figure, we observe that the average complexity for each bank is homogeneous between banks of the same country, with the exception of the Netherlands. In particular, the difference in the complexity of MFA protocols adopted by banks in the same country never exceeds 1.6 (except for the Netherlands, where this difference is 2.67 and 4 for IP and MP, respectively). The difference observed in the Netherlands is mainly due to a single bank, i.e., ING, which offers its customers a set of MFA protocols that notably differ from those offered by the other banks in the country. Given the small number of banks considered for each country, we cannot determine to what extent this result represents a national trend (see further discussion on this point in Section 6).

Moreover, we observe that MFA protocols for IP are, in general, more complex than those for MP. This fact is particularly noticeable for the MFA protocols employed by EU banks where the average complexity of MFA protocols for IP is 2.8 and the average complexity of MFA protocols for MP 2.1. On the other hand, this difference is lower for non-EU banks (3.5 against 3).

Figure 19 shows the impact of the different types of resources – memory, manual operations and extra devices – on the overall complexity of MFA protocols both for IP and MP. We observe that memory efforts have typically a higher impact on the overall com-

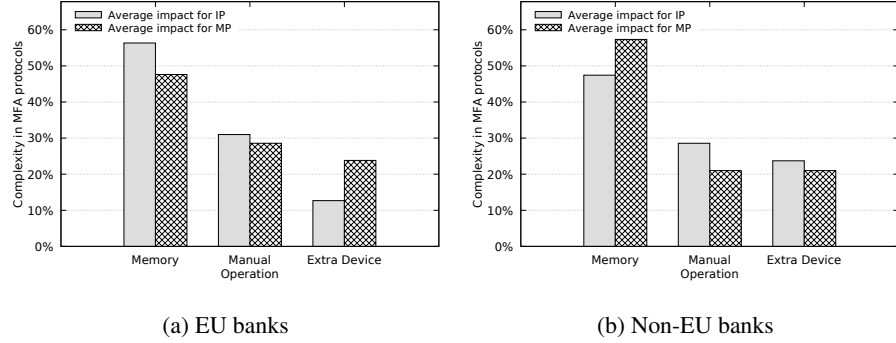


Figure 19: Average complexity of MFA protocols.

plexity of MFA protocols compared to the other two complexity aspects (around 50% of the complexity score). This is because the majority of the MFA protocols offered by banks rely on at least one knowledge factor and more than 30% of MFA protocols leverage two of them (see Section 5.1). On the other hand, the average complexity deriving from bringing an extra device is usually lower than the other two complexity aspects.

An in-depth analysis reveals that MFA protocols with lowest complexity are the ones for MP that only leverage combinations of inherence (i.e., fingerprints) and ownership factors and that do not require users to bring any extra device, any memory effort or manual input. On the contrary, the most complex MFA protocols are the ones involving at least one knowledge factor and a multi-factor hardware authenticator that requires users to manually insert `opid` and manually copy the obtained `otp` into the endpoint.

##### 5.5. Answer to RQ5: Correlations between compliance (with requirements and best practices), robustness against security threats and complexity of the MFA protocols adopted by banks

In this section we evaluate the three hypotheses presented in Section 4.4. Recall that *H1* is evaluated per bank, whereas *H2* and *H3* are evaluated per MFA protocol.

*Evaluation of H1.* We hypothesized that there exists a correlation between the adoption of exemptions and the complexity of MFA protocols employed by banks.

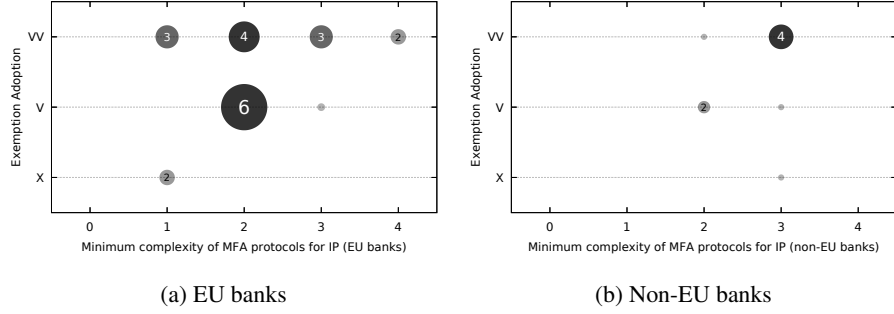


Figure 20: Relationship between exemptions and complexity (IP).

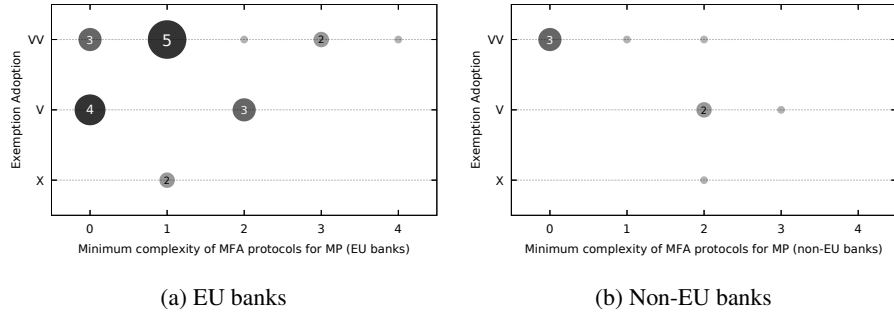


Figure 21: Relationship between exemptions and complexity (MP).

The results of our analysis is presented in Figures 20 and 21, which show the number of banks employing exemptions and the minimum complexity of the adopted MFA protocols for IP and MP, respectively. From three of these figures, we observe the lack of correlation between these two aspects. Figure 21b, instead, seem to show an inverse correlation. This is confirmed by the Pearson's correlation coefficients. For EU banks, we obtained coefficient of 0.37 and 0.19 (for IP and MP, respectively). For non-EU banks, we obtained a coefficient of 0.11 for IP and -0.64 for MP.

To assess if our results are statistically significant, we used the Fisher's exact test. The obtained p-values for EU banks are 0.073 and 0.066 for IP and MP respectively, whereas for non-EU banks they are 0.64 and 0.40 for IP and MP, respectively. Since the p-values are higher than the significance level of 5%, we cannot reject the null hypotheses.

Therefore,  $H1$  is not supported by the results. We conclude that the adoption of

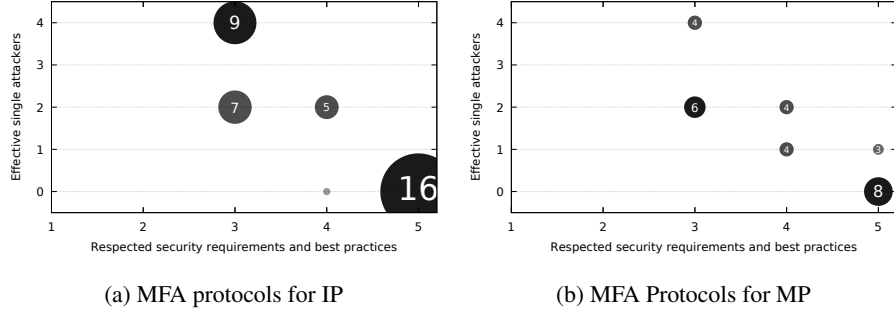


Figure 22: Relationship between compliance with security requirements (and best practices) and resistance to attacker models.

exemptions (hence the compliance with **BP3**) – even if very frequent – is not related to the employment of complex MFA protocols.

*Evaluation of H2.* The second hypothesis aims to assess the effectiveness of security requirements and best practices on the robustness of MFA protocols against attacker models. To this end, we verify whether there exists a correlation between the compliance of an MFA protocol with requirements and best practices (related to security aspects) and its resistance against the attacker models presented in Section 4.3.2.

Figure 22 presents the results of the analysis by indicating the number of MFA protocols (both for IP and MP) that comply to requirements and best practices and are vulnerable to single attackers. The level of compliance with the aforementioned security requirements and best practices is computed as the number of requirements and best practices met by the protocol.

It is interesting to observe that all MFA protocols comply with at least 3 security requirements and best practices. Furthermore, we observe that the more security requirements and best practices are met by an MFA protocol, the more robust the protocol is against attacker models. This is especially evident in the case of MFA protocols for IP: 16 MFA protocols comply with all considered security requirements and best practices and none of them is vulnerable to single attacker models. On the other hand, almost all MFA protocols complying with less than four security requirements and best practices are vulnerable to at least two single attacker models. A similar trend



can also be observed in the case of MFA protocols for MP. The Pearson's coefficient confirms our intuition. Specifically, the Pearson's coefficient is -0.93 and -0.84 for MFA protocols for IP and MP, respectively. These values show the presence of a very strong inverse correlation between the two variables.

To investigate the most relevant security requirements in the correlation, we calculate the correlation between the resistance of MFA protocols against attacks (i.e., the number of singletons) and every security requirement and best practice in the set. We notice that **RL6** and **RL5** are the requirements that mainly influence the robustness of MFA protocols against attacker models. In particular, the correlation between **RL6** and the number of attackers compromising an MFA protocol is 0.90 and 0.82 (for IP and MP, respectively), while between **RL5** and the number of attackers is 0.86 and 0.73 (for IP and MP, respectively). These results show that the use of operation-dependent otp and keeping the user aware of the operation she is going to authorize are the more effective solutions against the identified attacker models. We performed the Fisher's exact test to verify the statistical significance of our findings. The obtained p-values are  $5.93e-10$  and  $6.08e-7$  (for IP and MP, respectively), which are lower than the fixed significance level of 5%. Therefore, the null hypotheses can be rejected and our results can be considered to be statistically significant.

We can conclude that *H2* holds, indicating that the compliance with security requirements and best practices increases the resistance of MFA protocols to the considered attacker models.

*Evaluation of H3.* The third hypothesis aims to assess the independence between the complexity of an MFA protocol and its resistance against individual attacker models.

Figure 23 shows the results of the analysis for MFA protocols both for IP and MP. We note that there is not a clear correlation between the complexity of an MFA protocol and its resistance against attacks in both IP and MP context. An example of this can be observed for MFA protocols with complexity equal to 3. We can find both MFA protocols resistant against every individual attacker models and MFA protocols vulnerable to 4 of them.

We compute the Fisher's exact test to assess the independence between the com-

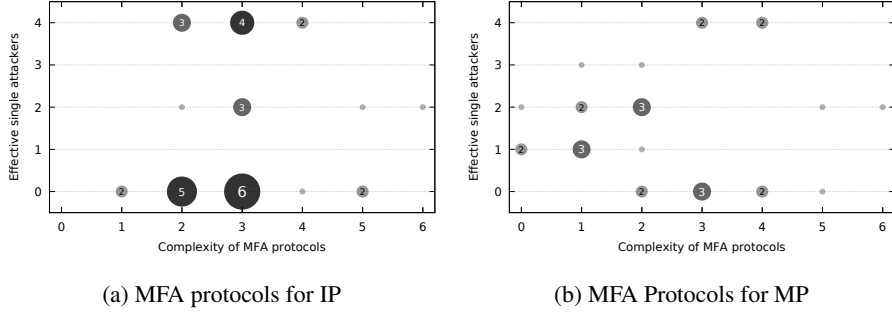


Figure 23: Relationship between complexity and resistance to attacker models.

plexity score of an MFA protocol and its resistance to attacker models. The obtained p-values for the IP and MP cases are 0.83 and 0.44, respectively. Both the values are higher than the significance level of 5%. Hence, we can confirm that the two variables are independent.

Therefore, we can conclude that  $H3$  holds, indicating that a complex MFA protocol is not necessary more resistant against attacker models.

## 6. Threats to Validity and Generality

In this section we list the limitations of our study and discuss their potential impact on the validity of our work. We distinguish between four types of threats, namely *internal*, *external*, *construct* and *conclusion*. We also discuss to what extent the methodology employed for the analysis can be generalized to other application domains.

*Internal threats.* Internal threats to validity are mostly related to our bank and, thus, MFA protocol dataset. We obtained all information relevant and necessary for the analysis from the documentation, tutorial and demos made available by banks. In fact, banks tend to publish as much documentation as possible to help their customers in the use of their services. However, as mentioned in Section 4, we had no direct interaction with banks and we only accessed public information and documentation. Therefore, we did not have access to information concerning the server side of the MFA implementation or technical features of the employed authenticators. These technical features could have effects that can be only partially captured by our analysis.

Moreover, we did not consider the release time and evolution of MFA protocols in our dataset. Reasonably, an MFA protocol should be evaluated against the regulations and directives that were in force when it was released. The changes in the legal framework (see Section 3.1) might have resulted in security patches and updates of older MFA protocols and older protocols might be supported for backward compatibility. In this case, banks might require new customers to only use the newer MFA protocols and even force old customers to use them.

*External threats.* The main external threat to the validity of our study is related to the size and composition of our dataset. Since we only consider three banks per country, we cannot fully support statements on national trends. Moreover, we selected the banks according to their dimension. Arguably, large banks have more resources to invest on the security of their services. Thus, extending our analysis to smaller banks would result in a more precise characterization of the online banking services landscape.

While MFA has been adopted in several types of online services, our analysis only focuses on MFA implementations adopted for online banking. Different types of online services can have very different business models and requirements. These differences can have a significant impact also on the employed MFA implementations. In this respect, our findings cannot be generalized to other types of online services. Below (generality), we discuss to what extent our methodology can be used for the analysis of MFA implementations adopted in other application domains.

*Construct threats.* A potential construct threat is our interpretation of regulations, directives and best practices. As a matter of fact, some definitions and descriptions contained in the documentation are informal or vague. This is probably done on purpose to make the rules generic and widely applicable. From our perspective, we had to cast these concepts to more rigorous and precise definitions. A concrete example is the interpretation of **BP2**, where we provided a list of standard solutions. Although based on standards defined by manufacturers, such a list might be too restrictive. For instance, some commercial solutions are commonly adopted and they represent the de facto standard.

Another construct threat is related to our representation of MFA protocols. In this work, we reconstructed MFA protocols by observing the client side of the authentication process. Since we have no information concerning the sequence of operations performed on the server side, we assume that such operations are executed properly and the communications between server and client work as intended. Moreover, we abstracted some details regarding the sequence of messages exchanged for the protocol execution. This lack of information has also an impact on our representation of authenticators. Although authenticators are, by definition, used on client side, their behavior might depend on some procedure executed remotely, e.g., the generation of authentication tokens. For instance, we did not consider the possible impairment of keys and seeds for the `otp` generation on the server side. Overall, the analysis of a low-level implementation might be subject to additional security risks and considerations.

*Conclusion threats.* In the analysis of the correlation between different criteria (Section 5.5), the validity of our conclusions were evaluated in terms of the statistical significance of the obtained results. In some cases, e.g., for hypothesis *H1*, the null hypotheses cannot be discarded and, thus, our conclusions cannot be considered statistically significant.

*Generality.* In this work, we focused on the MFA implementations used by some online banking services. Nevertheless, our approach for analyzing the robustness and complexity of MFA protocols is independent from the specific application domain. Therefore, our approach can be applied to analyze MFA protocols in general. On the other hand, assessing the compliance with laws and regulations is context-dependent. For instance, the European laws for digital identity (e.g., eIDAS [33]) do not require “dynamic linking”, which is a critical factor for the online banking sector [11, 12]. Therefore, a shift in the application domain would require rethinking the evaluation of the compliance with laws and best practices.

## **7. Lessons Learned**

In this section we summarize our findings and provide lessons learned that should be taken into account when designing MFA implementations.

*Lack of standardization brings high variety of MFA protocols.* Our study revealed that banks often offer several MFA protocols, which can be very different from each other. As shown in Section 5.1, these protocols vary for the employed authenticators and AFs, input/output data objects and data channels, providing different levels of security and complexity. One may argue that the standardization of the MFA protocol design could limit the proliferation of MFA protocol designs. However, to date only very few initiatives for standardization (e.g., FIDO [7] and OATH [8]) have emerged, and none of the considered banks is employing any of the proposed schemes. In our opinion, further standardization efforts and a better cooperation between banks and standardization bodies could help in limiting the fragmented landscape of MFA protocols and in improving their security level. Additionally, certifications – executed by third parties – could be established to attest the security level provided by each MFA protocol offered by a bank. In this way, users could make an informed choice of the MFA protocols to be used.

*Authenticators and AFs need further investigation.* The design of MFA protocols requires a deep understanding of authenticators and AFs. However, standardization bodies and legislators as well as banks seem to have not fully grasped their potential and security properties. Our analysis shows that the compliance with the legal framework in force and best practices does not guarantee a high security level of MFA protocols and further refinements to the available guidelines seem to be needed. For instance, the NIST labels SMS messages (named “Out-of-Band authenticators leveraging PSTN”) as *restricted* [13], i.e., “the authenticator capability to resist attacks is decreased, due to the evolution of threats”, and advises against their use in MFA protocols (**BP4**). By applying the same considerations presented in [13] and considering a slightly different set of attacker models (but mostly based on the ones presented in [13]), the results in Section 5.3 show that MFA protocols employing look-up secrets are less robust against attacker models compared to those employing SMS messages. We can thus argue that also look-up secrets should be considered restricted authenticators and their usage avoided. In addition, our analysis shows that MFA protocols employing inheritance factors are characterized by a low complexity and tend to be more resistant to attacker

models. However, as shown in Section 5.1, these AFs are not widely used in MFA protocols. We believe that further investigation on authenticators and AFs is needed for the design of effective and secure MFA protocols.

*Preliminary phases require more attention.* Although the enrollment and binding phases play a critical role in the security level of MFA protocols (see Section 3.1), the security of these phases is often overlooked. Banks often allow users to enroll and bind their authenticators remotely, and – on average – they tend not to comply with requirements and best practices concerning these phases (see Section 5.2) with possible serious implications on the overall security level of their MFA implementation. In our opinion, it would be necessary to increase the attention given to enrollment and binding. We believe that the compliance of these phases with the related requirements and best practices would help in reaching an adequate level in the provided security.

*Staying abreast of new MFA developments.* Our study revealed that – on average – the considered banks comply with the majority of legal requirements – extracted from EU directives and regulations currently in force. Interestingly, this is the case not only for EU banks, but also for Swiss and Chinese banks. The three US banks, instead, comply with only 2 of the identified requirements, highlighting a more permissive legal framework in the US. Nonetheless the high level of compliance, the robustness of the analyzed MFA protocols against attacker models is, in general, lower than expected. This means that the directives and regulations currently in force are not enough to guarantee a proper level of security. In this perspective, in our study we also considered the compliance of MFA protocols with RTS [12], which will become in force from mid-2019. Our results show that the compliance with the requirements introduced by this regulation (e.g., **RL5** and **RL6**) will provide a better resistance against attacks. Therefore, we expect that the progressive compliance with new regulations will improve the design of MFA protocols and increase their security level. However, more studies are required to assess the impact of new regulations on the adoption of MFA in the banking sector and to identify possible shortcomings.

## 8. Conclusion

This study has investigated the current situation regarding the adoption of MFA in the online banking context. In particular, we analyzed the MFA solutions adopted by 30 banks operating in different countries with respect to their compliance with laws and guidelines, the provided security and complexity.

Although MFA promises high security guarantees, our study shows that the security level offered by MFA protocols currently employed by banks is not as high as expected. In particular, half of the analyzed banks adopt at least one MFA protocol that is vulnerable to the considered attacker models acting individually. However, our results show that the compliance with requirements defined in RTS [12], which will become in force by mid-2019, will increase the resistance of MFA protocols against attacks and, thus, will improve the overall security of MFA implementations. Our analysis also shows that MFA protocols are usually not very easy to execute. We believe that a wider adoption of authenticators leveraging inherence factors will improve both the security level and the complexity of MFA protocols.

The online banking sector is continuously evolving, with new protocols being adopted to respond to threats and meet the requirements imposed by regulations and best practices. Therefore, we envision that the analysis presented in this work should be performed at regular basis to monitor the compliance, security and complexity of MFA adoptions, possibly considering a larger number of banks and countries. Collaboration with banks would provide access to their low-level implementation of MFA protocols, allowing for a more in-depth analysis. For instance, knowledge of technical features would permit to refine the abstract model of MFA protocols considered in this work. Leveraging this refined model, it would be possible to formally analyze the security of MFA implementations and identify attack patterns that would apply to the specific implementations. Moreover, an interesting direction for future work is the analysis of MFA solutions in other contexts besides online banking. This analysis would allow us to compare the status of MFA adoption in different contexts and identify possible discrepancies in the regulations and best practices, and consequently in MFA solutions, across contexts.

## Acknowledgments

This work has been partially supported by the FINSEC Project – Integrated Framework for Predictive and Collaborative Security of Financial Infrastructures<sup>23</sup> and by the H2020 Project SPARTA – Strategic Programs for Advanced Research and Technology in Europe.<sup>24</sup>

## References

- [1] Eurostat, Internet banking on the rise, <https://ec.europa.eu/eurostat/web/products-eurostat-news/-/DDN-20180115-1> (2018).
- [2] Federal Financial Institutions Examination Council, Authentication in an Internet Banking Environment, [https://www.ffiec.gov/pdf/authentication\\_guidance.pdf](https://www.ffiec.gov/pdf/authentication_guidance.pdf) (2007).
- [3] G. Lowe, Breaking and fixing the Needham-Schroeder Public-Key Protocol using FDR, in: Tools and Algorithms for the Construction and Analysis of Systems, Springer, 1996, pp. 147–166.
- [4] E. D. Cristofaro, H. Du, J. Freudiger, G. Norcie, Two-Factor or not Two-Factor? A Comparative Usability Study of Two-Factor Authentication, CoRR abs/1309.5344 (2013).
- [5] K. Krol, E. Philippou, E. D. Cristofaro, M. A. Sasse, "They brought in the horrible key ring thing!" Analysing the Usability of Two-Factor Authentication in UK Online Banking, CoRR abs/1501.04434 (2015).
- [6] C. S. Weir, G. Douglas, T. Richardson, M. Jack, Usable security: User preferences for authentication methods in eBanking and the effects of experience, Interacting with Computers 22 (3) (2010) 153–164.

---

<sup>23</sup><https://www.finsec-project.eu/>

<sup>24</sup><https://www.sparta.eu/>



- [7] FIDO, The FIDO Alliance, <https://fidoalliance.org/about/overview/> (2017).
- [8] OATH Authentication, OATH - initiative for open authentication, <https://openauthentication.org/>.
- [9] EBA, Recommendations for the Security of Internet Payments, <https://www.EBA.europa.eu/pub/pdf/other/recommendationsforthesecurityofinternetpaymentsen.pdf> (2013).
- [10] EBA, Recommendations for the Security of Mobile Payments - DRAFT, <https://www.EBA.europa.eu/pub/pdf/other/recommendationssecurityinternetpaymentsoutcomeofpcfinalversionafterpc201301en.pdf> (2013).
- [11] EBA, Directive 2015/2366 of the European Parliament and of the Council on payment services in the internal market (PSD2), <https://eur-lex.europa.eu/legal-content/en/TXT/?uri=CELEX:32015L2366> (2015).
- [12] EBA, Regulatory Technical Standards on Strong Customer Authentication and common and secure communication under Article 98 of PSD2, <https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018R0389&from=EN> (2017).
- [13] NIST, Special Publication - Digital Identity Guidelines, <https://pages.nist.gov/800-63-3/> (2017).
- [14] PCI Security Standards Council, Information Supplement - Multi-Factor Authentication, <https://www.pcisecuritystandards.org/pdfs/Multi-Factor-Authentication-Guidance-v1.pdf> (2017).
- [15] Centrify, Best Practices for Multi-factor Authentication, <https://www.centrify.com/media/3403844/bpb-best-practices-for-multi-factor-authentication.pdf> (2016).

- [16] Gemalto, Authentication BestPractices: Put control where it belongs, [http://www2.gemalto.com/email/2011/authsomethingstronger/whitepaper/Authentication\\_Best\\_Practices\\_WP \(EN\)\\_A4\\_web.pdf](http://www2.gemalto.com/email/2011/authsomethingstronger/whitepaper/Authentication_Best_Practices_WP_(EN)_A4_web.pdf) (2015).
- [17] PingIdentity, Multi-Factor Authentication: Best Practices for Securing the Modern Digital Enterprise, <https://www.pingidentity.com/content/dam/ping-6-2-assets/Assets/white-papers/en/mfa-best-practices-securing-modern-digital-enterprise-3001.pdf?id=b6322a80-f285-11e3-ac10-0800200c9a66> (2009).
- [18] J. Choubey, B. Choubey, Secure user authentication in internet banking: A qualitative survey, *International Journal of Innovation, Management and Technology* 4 (2) (2013) 198–203.
- [19] S. Kiljan, K. Simoens, D. De Cock, M. Van Eekelen, H. Vranken, A Survey of Authentication and Communications Security in Online Banking, *ACM Computer Surveys* 49 (4) (2016) 61:1–61:35.
- [20] A. Dmitrienko, C. Liebchen, C. Rossow, A. Sadeghi, Security Analysis of Mobile Two-Factor Authentication Schemes, *Intel Technology Journal* 18 (4) (2014) 138–161.
- [21] M. Althobaiti, Assessing usable security of multifactor authentication, Ph.D. thesis, University of East Anglia (2016).
- [22] C. G. Sinigaglia F., Carbone R., Strong authentication for e-banking: A survey on European regulations and implementations, in: *Proceedings of International Joint Conference on e-Business and Telecommunications*, SciTePress, 2017, pp. 480–485.
- [23] A. Armando, R. Carbone, L. Zanetti, Formal Modeling and Automatic Security Analysis of Two-Factor and Two-Channel Authentication Protocols, in: *Network and System Security, LNCS 7873*, Springer, 2013, pp. 728–734.

- [24] D. DeFigueiredo, The Case for Mobile Two-Factor Authentication, *IEEE Security and Privacy* 9 (2011) 81–85.
- [25] K. Furst, W. W. Lang, D. E. Nolle, Internet banking: Developments and prospects, *Economic and Policy Analysis Working Paper No. 2000-9*, Office of the Comptroller of the Currency (2000).
- [26] F. Hao, D. Clarke, Security Analysis of a Multi-factor Authenticated Key Exchange Protocol, in: *Applied Cryptography and Network Security, LNCS 7341*, Springer, 2012, pp. 1–11.
- [27] E. Kennedy, C. Millard, Data security and multi-factor authentication: Analysis of requirements under EU law and in selected EU Member States, *Computer Law and Security Review* 32 (2016) 91–110.
- [28] G. Sciarretta, R. Carbone, S. Ranise, L. Viganò, Design, Formal Specification and Analysis of Multi-Factor Authentication Solutions with a Single Sign-On Experience, in: *Principles of Security and Trust, LNCS 10804*, Springer, 2018, pp. 188–213.
- [29] EBA, Directive 2007/64/EC of the European Parliament and of the Council on payment services in the internal market (PSD), <https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=celex%3A32007L0064> (2007).
- [30] PCI Security Standards Council, Data Security Standard, [https://www.pcisecuritystandards.org/documents/PCI\\_DSS\\_v3-2.pdf](https://www.pcisecuritystandards.org/documents/PCI_DSS_v3-2.pdf) (2016).
- [31] PCI Security Standards Council, Payment Application Data Security Standard, [https://www.pcisecuritystandards.org/documents/PA-DSS\\_v3-2.pdf](https://www.pcisecuritystandards.org/documents/PA-DSS_v3-2.pdf) (2016).
- [32] V. Hauptert, T. Müller, On App-based Matrix Code Authentication in Online Banking, in: *Proceedings of International Conference on Information Systems Security and Privacy*, SciTePress, 2018, pp. 149–160.

- [33] European Parliament and Council, Regulation (EU) No 910/2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC, <https://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:32014R0910> (2014).
- [34] Virus Bulletin, The Evolution of Webinjects, <https://www.virusbulletin.com/uploads/pdf/conference/vb2014/VB2014-Boutin.pdf> (2014).
- [35] Kaspersky Lab, Online and Mobile Banking Threats, <https://media.kaspersky.com/en/business-security/online-and-mobile-banking-threats-kaspersky-whitepaper.pdf?icid=en-UK:ent-content> (2013).
- [36] A. M. Hagalisletto, Analyzing two-factor authentication devices, Tech. rep., University of Oslo (2007).
- [37] Android Developers Documentation, Android Guides - Protect against security threats with SafetyNet, <https://developer.android.com/training/safetynet/>.
- [38] Android Developers Documentation, Android Guides - Security Tips, <https://developer.android.com/training/articles/security-tips#UserData>.
- [39] Android Developers Documentation, Android Guides - Security with HTTPS and SSL, <https://developer.android.com/training/articles/security-ssl>.
- [40] ISO, ISO 9241-11:2018(en), Ergonomics of human-system interaction – Part 11: Usability: Definitions and concepts.
- [41] C. S. Weir, G. Douglas, M. Carruthers, M. Jack, User perceptions of security, convenience and usability for ebanking authentication tokens, *Computers & Security* 28 (1) (2009) 47–62.
- [42] J. Brooke, SUS: A quick and dirty usability scale, in: *Usability Evaluation in Industry*, Taylor and Francis, 1996.

- [43] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, I. H. Witten, The WEKA Data Mining Software: An Update, SIGKDD Explor. Newsl. 11 (1) (2009) 10–18.
- [44] D. E. Hinkle, W. Wiersma, S. G. Jurs, Applied statistics for the behavioral sciences, Houghton Mifflin, 2003.
- [45] BankID - electronic identification solution, <https://www.bankid.com/en/>.
- [46] CNBC, Google is missing out on billions of dollars by not having an app store in China, new data shows, <https://www.cnbc.com/2018/01/17/google-misses-out-on-billions-in-china.html>.
- [47] Google Developers, <https://developers.google.com/china/> (2017).