Short Paper: How to Attack PSD2 Internet Banking

Vincent Haupert(✉) and Stephan Gabert

Friedrich-Alexander University Erlangen-Nürnberg (FAU), Erlangen, Germany
vincent.haupert@cs.fau.de

Abstract. Internet banking security is set to take a major step forward: On September 14, 2019, the Regulatory Technical Standards of the Revised Payment Service Directive (PSD2) are going to be effective within the European Union and the European Economic Area. This regulation makes two widely demanded transaction security properties mandatory: two-factor authentication, and the dynamic linking of the authentication code to the transaction's beneficiary and amount (full transaction authentication). Even though the regulation is undoubtedly a positive development from a security perspective, it does not account for all the technical and human weak points involved in the transaction process. In this paper, we look at a series of attacks targeting online and mobile banking that are possible even in a post-PSD2 era. Despite the regulatory motivation of this work, the presented issues and suggestions to address them are likely to be universal for internet banking in general.

Keywords: Online Banking · PSD2 · RTS · SCA · Attacks.

1 Introduction

At FC 2013, Adham *et al.* presented a work entitled "How to Attack Two-Factor Authentication Internet Banking" [1]. They outlined the current state of online banking transaction security in the United Kingdom (UK) and pointed out how it might be attacked. Although they appreciated the increasing adoption of a second factor for transaction authentication, they argued that an additional one-time password (OTP) alone would not sufficiently protect a customer from falling prey to malware. Their primary concern was that the to date employed transaction authentication methods did not provide full transaction authentication. That means, that the resulting OTP as generated by the respective two-factor authentication (2FA) method did not allow for an independent verification of the transaction's integrity. As a consequence, 2FA did only stop adversaries from performing arbitrary transactions at any time, but did not prevent a real-time transaction manipulation attack.

In March 2018, the Regulatory Technical Standards (RTS) came into force and are going to apply from September 2019 [8]. The RTS are part of the Revised Payment Service Directive (PSD2) which replaced its predecessor PSD that initially introduced the Single European Payment Area (SEPA). A primary goal of the RTS is to make payment services more secure and to foster the population's trust in online banking as it is still on a steady rise [9]. To achieve this, the RTS stipulate strong customer authentication (SCA) that requires remote payments to make use of at least two independent and mutually exclusive elements of the categories knowledge, possession and inherence. Additionally, the payment service provider must issue a single-use authentication code that is dynamically linked to the transaction's beneficiary and amount, with both being displayed to the customer for verification. The latter makes full transaction authentication mandatory. This enables a customer to reliably detect fraud even if the transfer-issuing device is infected with malware.

Even though undoubtedly a positive development, the RTS are not going to rule out all of the attack vectors. In the spirit of Adham *et al.*, we aim at identifying weak points that neither full transaction authentication nor the new regulation addresses. To that end, we include attacks that leverage technical as well as social engineering aspects.

2 Background and Related Work

Carrying out a credit transfer consists of two steps: issuing and confirming. At first, customers need to log in to their online banking. This process is usually secured through a knowledge authentication element, i.e., a password. After successful login, the customer *issues* a transfer to a desired beneficiary by specifying her account number and the amount. To make the transfer effective, the bank additionally requires the transaction's *confirmation*, usually by asking for an OTP that within online banking is frequently referred to as TAN (transaction authentication number). The method that dynamically links the transaction and yields the TAN is hence called TAN method.

In the following, we outline three popular TAN methods and attacks against each of them from the related work. All of these methods offer a 2FA as well as full transaction authentication and, hence, display the transfer details—i.e., the beneficiary's account number and the amount—on a second device. It is the responsibility of the customer to verify that the displayed transfer details match the desired ones [19]. If they do not match, a customer must abort the transaction ("What You See Is What You Sign", WYSIWYS).

SMS Authentication. The SMS-based authentication procedure (smsTAN) relies on the short message service (SMS) to transmit a text message with the transfer details and the TAN from the bank to the customer. In 2008 and 2014, Engel discovered several vulnerabilities in the Signalling System No. 7 (SS7) protocol that forms the foundation SMS messages are built on [21]. Also Long-Term Evolution (LTE)—the to date latest mobile communication standard—is prone to attacks [22]. Mulliner *et al.* also addressed the security of the SMS [18] and showed how to abuse flaws to attack the smsTAN method [17].

Smartcard Authentication. Particularly European banks rely on the Chip and PIN (EMV) standard to create a TAN using the customer's bank card. This method requires a dedicated reader device that also displays the transfer details to the customer. In 2009, Drimer *et al.* uncovered various design and protocol flaws in the respective implementation of banks in the UK and a year later, Murdoch *et al.* successfully launched an attack against the EMV protocol that allowed for using a stolen card without knowing the PIN [20]. In 2011, the attack of Murdoch *et al.* even appeared in the wild, when about 40 sophisticated card forgeries surfaced in France, causing an estimated net loss below €600,000 [12]. In 2014, Bond *et al.* discovered another flaw in the EMV protocol that enabled an adversary to de facto clone a card using a rogue point-of-sale terminal [3].

Smartphone Authentication. With the advent of smartphones, banks also started to leverage their high availability and cost effectiveness by implementing app-based TAN methods. These procedures work similar to the smsTAN method but deliver the data over the internet using a dedicated app developed by the bank. In 2014, Dmitrienko *et al.* identified various weaknesses in app-based 2FA solutions [5]. They successfully infected both authentication devices—personal computers and mobile phones—with a self-implemented cross-platform malware. Similarly, Konoth *et al.* presented an attack against smsTAN that only required the infection of the user's computer due to the high integration smartphones and PCs offer today [15]. Haupert *et al.* have contributed to the field of attacks that target one-device mobile banking, a transaction authentication scheme that is becoming increasingly popular [2]. They argue that core requirements of a secure 2FA are violated if both authentication elements are operated by the same multi-purpose device without providing a trusted path [14,13].

3 Threat Model

We suppose that a customer ordered a product online and pays by bank wire transfer through her online banking. This customer uses 2FA with a TAN method that provides full transaction authentication. To that end, the TAN method displays the transfer details on a second, independent device for verification.

An attacker targets at redirecting the customer's transfer order to another account. The adversary only replaces the beneficiary's account number and leaves the amount unchanged. This happens due to the following reason: when paying an invoice, the customer is usually aware of the amount but frequently unaware of the beneficiary's account number. For the purpose of manipulating a transaction, we assume that the adversary can completely compromise the transfer-issuing channel, which enables her to observe or tamper all the details the customer receives, sees, enters or sends. The attacker cannot, however, control the victim's TAN method. Instead, the attacker attempts to discourage the victim from performing a correct verification of the account number during confirmation.

The assumed threat model is rather weak as it does not require infection of both devices that are involved in the transaction authentication process. Owing to the success of banking malware families like *ZeuS* [7], it even became a best practice for banks to regard the customer's computer as malware-infected [10].

4 Attacks and Challenges

4.1 Clipboard Hijacking

On desktop and mobile operating systems, the clipboard is a shared resource that every application can read and write. This allows for stealing [11] and manipulating [26] the data by monitoring the contents of the system clipboard.

The international banking account number (IBAN)—the default within the SEPA—adheres to a well defined ISO standard. According to that standard, an IBAN can consist of up to 34 alphanumeric characters. In the case that a customer receives a digital invoice, e.g., a PDF, a customer is likely going to use the copy and paste method to avoid entering the IBAN manually.

Attack. As the IBAN also contains two check digits, it is easy to validate the correctness of a given candidate. Consequently, an attacker monitoring the clipboard can also detect an IBAN in the system clipboard and replace it with the IBAN of an attacker-controlled account. As a customer pasted the IBAN, she might assume it must be correct—ignoring a potentially infected computer—and, hence, skips the account number verification. As the customer did not enter the IBAN manually, she might be also less likely to recall the original IBAN.

Defense. To mitigate this attack, a bank should disable the possibility to paste clipboard data into a form element. Developers can prevent this within web and mobile applications by installing a custom listener for paste events.

4.2 SMS Autofill on iOS and macOS

In 2016, Konoth *et al.* already criticized the synchronisation of SMS from iOS to macOS, as this allows for an attacker to only infect the transfer-issuing channel to control both authentication elements [15]. With the release of iOS 12 and macOS 10.14 (Mojave) in September 2018, this integration became even closer: if a customer visits a webpage that asks for an OTP sent by SMS, Safari on macOS 10.14 offers automatic insertion of the OTP in a predefined field [16]. This feature is also available for text fields in apps running on iOS 12.

Attack. Autofilling an OTP is only meaningful and without security implications if the authentication happens without context. This is true for user but not for transaction authentication: the essential security task during transaction confirmation is the verification of the transfer details contained within the SMS. Autofilling the TAN encourages the customer to omit this verification step. An attacker who compromised the device and manipulated a transfer could trigger the autofill. Consequently, the victim might not verify the transfer for integrity as the TAN gets filled in automatically. Instead, she might just proceed making the transfer effective.

Defense. Our tests show that the keyword “code” is necessary within the SMS to trigger this feature. As a consequence, banks should avoid this word within their SMS text message. In our tests, the words “OTP” or “TAN” did not trigger the autofill feature, but Apple might change this behavior at any time.

4.3 Stealthy Transaction Manipulation

In the course of our research, we noticed that many important and large German banks—for example, *Sparkassen* as well as *Volksbanken und Raiffeisenbanken*—also show a transaction's details on the confirmation webpage that asks the customer for a TAN. This behavior has a counter-productive effect, as it suggests that the transfer-issuing channel is trustworthy. To make things worse, it might even habituate a user to perform a faulty transaction verification: instead of comparing the details shown on the customer's TAN device to the original invoice, she might compare them to the details shown within the transfer-issuing channel, e.g., the web browser.

Attack. An adversary can leverage this potential habituation: a customer who compares the information within the TAN device to the details shown within the transfer-issuing channel, is not going to spot a deviation. One might argue that a customer might recall the account number she originally entered. This is, of course, possible. In the case of an IBAN, however, this scenario is at least debatable because of the cumbersome format with up to 34 digits.

Defense. This is an issue of usable security [24]. From a technical point of view, banks should stop displaying transaction details within the transfer-issuing channel, as this behavior is plain unconducive.

4.4 Digital Invoice Manipulation

After purchase, online shops send out an e-mail to their customers that contains a PDF invoice or a link that displays the invoice and payment details within the browser. Even if a customer pays on account, they frequently do no longer receive a paper invoice along with the ordered item.

Attack. Instead of tampering with the transfer order, a malware might as well directly modify the invoice. Due to the IBAN's well defined format, it is easy to detect and replace occurrences within a PDF or HTML page. Hence, an attacker could manipulate the invoiced account number directly. Even in the case that a customer correctly verifies the transaction, she has no chance to spot the fraud.

Defense. Online shops could send out the payment details by postal mail only. Particularly for payments in advance, however, this is probably not an option. Signing PDF invoices is probably not going to help either as the majority of users probably would not deem an unsigned PDF suspicious.

4.5 Transfer Templates

To avoid having the customer enter the account number for recurring recipients, many banks offer explicit and implicit transfer templates. For explicit transfer templates, a customer has to actively create a new entry within the online banking that contains the beneficiary's name and account number. When a customer wants to perform a credit transfer to one of her contacts saved as transfer template, she can just select this contact from a list. Implicit transfer templates work similarly but do not require the customer to actively create entries: when a customer types the beneficiary's name into the transfer order form, the online banking automatically searches the past transactions and suggests filling in the corresponding account number.

Attack. Transfer templates operate on the client side. That means, that they only help to fill in a form but the data sent to the bank is the same as filling in that information manually. As a consequence, an attacker can fill in an arbitrary account number when a customer makes use of a transfer template. When asked for verification during transaction confirmation, the customer likely does not have an invoice or another channel to verify the displayed transaction details. That is likely the reason why the customer made use of a transfer template in the first place.

Defense. Transfer templates are difficult to reconcile with the principle of WYSIWYS. Therefore, it is hard to create a solution that offers the comfort of transfer templates on the one hand, but also encourages a customer to verify a transaction's account number on the other hand. Masking a small part of the account numbers for transfer templates and past transactions helps addressing this issue: it spares most typing but makes sure the customer has the beneficiary's account number available through a source different from the online banking.

5 Conclusion

In this paper, we presented five different attacks which target the way online banking credit transfers work and how the customer uses them. Most of the attacks have in common that the customer is not aware of the payee's account number. Moreover, account number formats like the IBAN make transaction verification a cumbersome task. In addition, the currently used TAN methods only display the IBAN but not the name of the recipient.

This, however, could make transaction verification an easier task: Apart from the beneficiary's account number and the amount of the transfer, the TAN method should also display the beneficiary's name. For that purpose, a TAN method could perform a lookup in the customer's transaction history. If a customer never used the given account number before, a bank could at least use their global transaction history to show a confidence level for the that account number. A similar service was already introduced by Dutch banks in 2017, with a system which ensures that the beneficiary's name belongs to the specified IBAN [6].

As our threat model assumes full control over the transfer-issuing channel, our proposed defenses are not going to fully eliminate but rather complicate a successful attack. To mitigate attacks, it is essential that a customer is aware of the untrustworthiness of the transfer-issuing channel. The user, however, frequently lacks this awareness [4,23,25]. To eliminate this attack vector, banks need to come up with procedures that guarantee integrity as soon as the customer enters the payment details. This, however, remains a medium-term task. Nevertheless, the PSD2 is a step into the right direction and will make payments more secure.

References

1. Adham, M., Azodi, A., Desmedt, Y., Karaolis, I.: How to attack two-factor authentication internet banking. In: Financial Cryptography and Data Security - 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers. pp. 322–328 (2013). https://doi.org/10.1007/978-3-642-39884-1_27
2. Bankenverband/GfK: Online-Banking in Deutschland (06 2018), http://go.bdb.de/UHbYz
3. Bond, M., Choudary, O., Murdoch, S.J., Skorobogatov, S.P., Anderson, R.J.: Chip and skim: Cloning EMV cards with the pre-play attack. In: 2014 IEEE Symposium on Security and Privacy, SP 2014, Berkeley, CA, USA, May 18-21, 2014. pp. 49–64 (2014). https://doi.org/10.1109/SP.2014.11
4. Dhamija, R., Tygar, J.D., Hearst, M.A.: Why phishing works. In: Proceedings of the 2006 Conference on Human Factors in Computing Systems, CHI 2006, Montréal, Québec, Canada, April 22-27, 2006. pp. 581–590 (2006). https://doi.org/10.1145/1124772.1124861
5. Dmitrienko, A., Liebchen, C., Rossow, C., Sadeghi, A.: On the (in)security of mobile two-factor authentication. In: Financial Cryptography and Data Security - 18th International Conference, FC 2014, Christ Church, Barbados, March 3-7, 2014, Revised Selected Papers. pp. 365–383 (2014). https://doi.org/10.1007/978-3-662-45472-5_24
6. Dutch Payments Association: Dutch banks introduce innovative IBAN-Name Check (2 2017), https://www.betaalvereniging.nl/en/actueel/persberichten/dutch-banks-introduce-innovative-iban-name-check/
7. Etaher, N., Weir, G.R.S., Alazab, M.: From zeus to zitmo: Trends in banking malware. In: 2015 IEEE TrustCom/BigDataSE/ISPA, Helsinki, Finland, August 20-22, 2015, Volume 1. pp. 1386–1391 (2015). https://doi.org/10.1109/Trustcom.2015.535
8. European Commission: Commission delegated regulation (EU) 2018/389 supplementing directive (EU) 2015/2366 of the european parliament and of the council with regard to regulatory technical standards for strong customer authentication and common and secure open standards of communication (3 2018), https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018R0389
9. European Commission: Internet banking on the rise (01 2018), http://ec.europa.eu/eurostat/web/products-eurostat-news/-/DDN-20180115-1
10. European Union Agency for Network and Information Security: Flash note: EU cyber security agency ENISA; "high roller" online bank robberies reveal security gaps (07 2012), https://www.enisa.europa.eu/news/enisa-news/copy_of_eu-cyber-security-agency-enisa-201chigh-roller201d-online-bank-robberies-reveal-security-gaps
11. Fahl, S., Harbach, M., Oltrogge, M., Muders, T., Smith, M.: Hey, you, get off of my clipboard - on how usability trumps security in android password managers. In: Financial Cryptography and Data Security - 17th International Conference, FC 2013, Okinawa, Japan, April 1-5, 2013, Revised Selected Papers. pp. 144–161 (2013). https://doi.org/10.1007/978-3-642-39884-1_12
12. Ferradi, H., Géraud, R., Naccache, D., Tria, A.: When organized crime applies academic results: a forensic analysis of an in-card listening device. J. Cryptographic Engineering **6**(1), 49–59 (2016). https://doi.org/10.1007/s13389-015-0112-3, https://doi.org/10.1007/s13389-015-0112-3
13. Haupert, V., Maier, D., Müller, T.: Paying the price for disruption: How a fintech allowed account takeover. In: Proceedings of the 1st Reversing and Offensive-

oriented Trends Symposium. pp. 7:1–7:10. ROOTS, ACM, New York, NY, USA (2017). https://doi.org/10.1145/3150376.3150383

14. Haupert, V., Maier, D., Schneider, N., Kirsch, J., Müller, T.: Honey, I shrunk your app security: The state of android app hardening. In: Detection of Intrusions and Malware, and Vulnerability Assessment - 15th International Conference, DIMVA 2018, Saclay, France, June 28-29, 2018, Proceedings. pp. 69–91 (2018). https://doi.org/10.1007/978-3-319-93411-2_4
15. Konoth, R.K., van der Veen, V., Bos, H.: How anywhere computing just killed your phone-based two-factor authentication. In: Financial Cryptography and Data Security - 20th International Conference, FC 2016, Christ Church, Barbados, February 22-26, 2016, Revised Selected Papers. pp. 405–421 (2016). https://doi.org/10.1007/978-3-662-54970-4_24
16. Miller, C.: Here's how iOS 12's new security code auto-fill feature works (06 2018), https://9to5mac.com/2018/06/04/safari-security-code-auto-fill
17. Mulliner, C., Borgaonkar, R., Stewin, P., Seifert, J.: SMS-based one-time passwords: Attacks and defense - (short paper). In: Detection of Intrusions and Malware, and Vulnerability Assessment - 10th International Conference, DIMVA 2013, Berlin, Germany, July 18-19, 2013. Proceedings. pp. 150–159 (2013). https://doi.org/10.1007/978-3-642-39235-1_9
18. Mulliner, C., Golde, N., Seifert, J.: SMS of death: From analyzing to attacking mobile phones on a large scale. In: 20th USENIX Security Symposium, San Francisco, CA, USA, August 8-12, 2011, Proceedings (2011), http://static.usenix.org/events/sec11/tech/full_papers/Mulliner.pdf
19. Murdoch, S.J., Becker, I., Abu-Salma, R., Anderson, R.J., Bohm, N., Hutchings, A., Sasse, M.A., Stringhini, G.: Are payment card contracts unfair? (short paper). In: Financial Cryptography and Data Security - 20th International Conference, FC 2016, Christ Church, Barbados, February 22-26, 2016, Revised Selected Papers. pp. 600–608 (2016). https://doi.org/10.1007/978-3-662-54970-4_35, https://doi.org/10.1007/978-3-662-54970-4_35
20. Murdoch, S.J., Drimer, S., Anderson, R.J., Bond, M.: Chip and PIN is broken. In: 31st IEEE Symposium on Security and Privacy, S&P 2010, 16-19 May 2010, Berleley/Oakland, California, USA. pp. 433–446 (2010). https://doi.org/10.1109/SP.2010.33
21. Rao, S.P., Kotte, B.T., Holtmanns, S.: Privacy in LTE networks. In: Proceedings of the 9th EAI International Conference on Mobile Multimedia Communications, MobiMedia 2016, Xi'an, China, June 18-20, 2016. pp. 176–183 (2016), http://dl.acm.org/citation.cfm?id=3021417
22. Rupprecht, D., Kohls, K., Holz, T., Pöpper, C.: Breaking LTE on layer two. In: IEEE Symposium on Security & Privacy (SP). IEEE (May 2019)
23. Schechter, S.E., Dhamija, R., Ozment, A., Fischer, I.: The emperor's new security indicators. In: 2007 IEEE Symposium on Security and Privacy (S&P 2007), 20-23 May 2007, Oakland, California, USA. pp. 51–65 (2007). https://doi.org/10.1109/SP.2007.35
24. Schneier, B.: Stop trying to fix the user. IEEE Security & Privacy **14**(5), 96 (2016)
25. Watson, B., Zheng, J.: On the user awareness of mobile security recommendations. In: Proceedings of the 2017 ACM Southeast Regional Conference, Kennesaw, GA, USA, April 13-15, 2017. pp. 120–127 (2017). https://doi.org/10.1145/3077286.3077563
26. Zhang, X., Du, W.: Attacks on android clipboard. In: Detection of Intrusions and Malware, and Vulnerability Assessment - 11th International Conference, DIMVA 2014, Egham, UK, July 10-11, 2014. Proceedings. pp. 72–91 (2014). https://doi.org/10.1007/978-3-319-08509-8_5