View The Email to Get Hacked:
Attacking SMS-Based Two-Factor Authentication

Philipp Markert*
Ruhr University Bochum

Florian Farke*
Ruhr University Bochum

Markus Dürmuth
Ruhr University Bochum

Abstract

In the effort to improve the security of their logins, a growing number of online services offer two-factor authentication (2FA). Beside other mechanisms, one-time passwords sent via SMS are still one of the most used second factors. We empirically analyzed the top 100 of the Tranco top sites ranking and identified 31 unique online services that provide two-factor authentication. We also evaluated which forms of 2FA are used and found software tokens and SMS being the most widely used ones. Additionally, we present a phishing attack against Google's SMS-based two-factor authentication exploiting the similarity between the SMS containing the one-time password and SMS sent as part of Google Gmail's confidential mode. Through this attack, an adversary can obtain the one-time password for the 2FA by luring the victim to a site which mimics the look of the Gmail confidential mode without adding any steps that are not part of the original protocol flow.

1 Introduction

Websites increasingly use two-factor authentication (2FA) in order to strengthen password-based authentication by including an additional factor for authentication.

There are a variety of different solutions to prove the possession of such a second factor: (i) Transmit a one-time password (OTP) via SMS or by automated phone calls (robocalls) to a preregistered number; some services also offer to add an email address to receive the code. In all three cases, the user needs to provide the received OTP in a second step after the password-based login. (ii) Authentication tokens holding a cryptographic key and engaging in an authentication protocol with the server. Here, the user needs to connect the token to the device when asked during login, e. g., via USB to a computer or NFC when using a smartphone. (iii) Authentication apps that are paired once with the account and continuously generate a login code afterward. This app-based authentication is also known as software token, and just like for SMS, calls, or email, it requires the user to provide the code in an additional web form.

The security of 2FA depends on the security of the communication channel, which is used to transmit the OTP as well as the integrity of the second factor. SMS-based 2FA often fails to ensure these assumptions, and we have seen various examples in the past where attackers were thus able to login into an account despite the use of 2FA. In May 2019, a bitcoin wallet was stolen by a SIM port attack which enabled the attacker to receive the OTPs sent via SMS [3]. Similar attacks were also observed earlier [2, 4]. Other examples include phishing attacks [15] where the attacker tricks the victim into disclosing the OTP or man-in-the-browser attacks where the OTP is stolen during the transmission.

Mirian et al. [10] studied the black market segment of online account hijacking services more systematically and found that some of these services are sophisticated enough to bypass SMS-based 2FA.

For the described reasons, the National Institute of Standards and Technology (NIST) in the United States deprecated the use of SMS-based two-factor authentication in a draft of its Digital Authentication Guideline in 2016 [14]. Although the statement was weakened in the final version, using SMS is currently restricted due to the variety of possible threats. Furthermore, the NIST notes that the classification may be strengthened again in future releases if the threat landscape changes [7]. The British National Cyber Security Centre follows a similar approach, stating that "text messages are not the most secure type of 2FA, but still offer a huge advantage over not using any 2FA" [12].

*The first two authors contributed equally to the paper.

Who Are You?! Adventures in Authentication (WAY) 2019.
August 11, 2019, Santa Clara, CA, USA.

In this work, we are looking at a specific problem arising from the need to manually copy the OTP from an SMS to a web page when using 2FA. We have observed that Google Gmail's confidential mode uses a similar SMS-based mechanism to gain access to protected emails. The reuse of this mechanism is an example for a potential to mix up the different channels, and a blueprint for a phishing attack against SMS-based 2FA. This specific problem also raises the more general question how and if at all an SMS can be bound to a specific purpose and is yet another example for the shortcomings of SMS-based 2FA.

Overview In Section 2, we present the results of our investigation what factors websites offer for their 2FA implementations. In Table 2, we present a detailed account of a phishing attack that mimics the Gmail confidential mode to obtain the OTP sent via SMS as part of a 2FA login procedure. We describe a detailed account of a phishing attack that mimics the Gmail confidential mode to obtain the OTP sent via SMS as part of a 2FA login procedure in Table 2. Related work will be discussed in Section 4 before we conclude with Section 5.

2 Use of Two-Factor Authentication

To analyze the spread of two-factor authentication across popular service providers, we selected websites using the Tranco list [9]. The Tranco list offers the advantage of being reproducible as it is possible to obtain the list for a specific date. We obtained the list on May 22, 2019, and selected the top 100 domains. To decide whether a web service offers two-factor authentication, we searched `twofactorauth.org`[1] and also checked each web page manually.

Of the 100 analyzed domains, we found 75 that point to a website with registration and login, respectively. Out of these 75 domains, we identified 57 unique logins. For example, `bbc.co.uk` and `bbc.com` are both present in our list but use the same underlying login and only differ in the top-level domain. The same holds for `google.com` and `youtube.com`, where YouTube redirect its users to the Google login page.

Of the remaining 57 domains, we found 31 that allows users to secure their accounts by enabling 2FA. Table 1 presents in detail which factors each of the 31 services supports. Software tokens (SW), i. e., additional authentication apps, are the most frequently used form of 2FA, with 25 (81 %) services utilizing them. Noteworthy, despite the variety of known attacks against 2FA implementations that use phones as a second factor, SMS-based 2FA is similarly widespread with 24 (77 %) of the services using them to send an OTP.

Furthermore, 4 (12 %) services even rely solely on phone-based transmission channels (i.e., SMS or call), namely `linkedin.com`, `weibo.com`, `jd.com`, and `bit.ly`. Both hardware tokens (HW) and robocalls are not as widespread as former options. Surprising is the little support of hardware tokens. Only 7 (23 %) services support hardware tokens, although they offer the highest security level across all 2FA implementations. Having in mind that we analyzed the top 100 services, we only consider this as an upper bound and expect the numbers to be lower, e. g., across the top 1000 or 10 000.

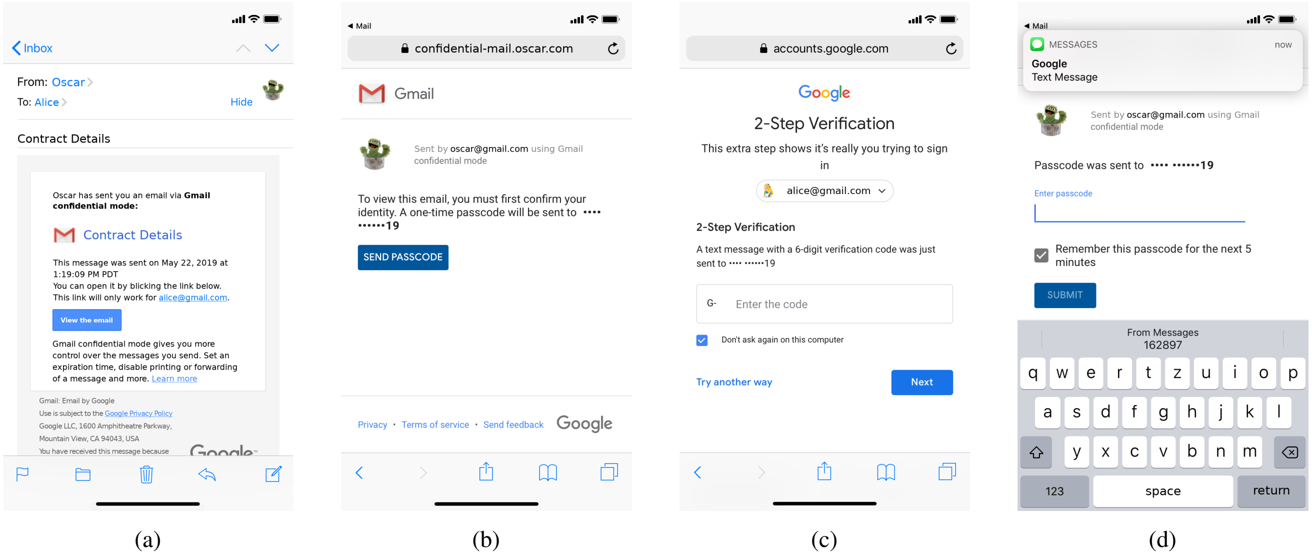
Table 1: Use of different 2FA implementations across popular domains. (HW stands for *hardware token* and SW for *software token* respectively.)

Rank	Domain	SMS	Call	Email	HW	SW
1	google.com	●	●	–	●	●
4	facebook.com	●	–	–	●	●
5	microsoft.com	●	●	–	–	●
6	twitter.com	●	–	–	●	●
11	linkedin.com	●	●	–	–	–
12	apple.com	●	●	–	–	●
14	yahoo.com	●	●	●	–	–
16	amazon.com	●	●	–	–	●
17	pinterest.com	●	–	–	–	●
23	adobe.com	●	–	●	–	●
25	reddit.com	–	–	–	–	●
29	wordpress.com	●	–	–	–	●
32	weibo.com	●	–	–	–	–
33	vk.com	●	–	–	–	●
36	jd.com	●	–	–	–	–
39	github.com	●	–	–	●	●
42	yandex.ru	–	–	–	–	●
44	ebay.com	●	–	–	–	●
45	360.cn	–	–	–	–	●
47	tumblr.com	●	–	–	–	●
48	bit.ly	●	–	–	–	–
50	godaddy.com	●	–	–	●	●
53	mozilla.org	–	–	–	–	●
55	twitch.tv	●	–	–	–	●
57	paypal.com	●	–	–	●	●
59	yahoo.co.jp	–	–	●	–	●
61	mail.ru	●	–	–	–	●
62	pornhub.com	●	–	–	–	●
65	dropbox.com	●	–	–	●	●
76	naver.com	–	–	–	–	●
83	sourceforge.net	–	–	–	–	●

Finally, only two services (Adobe and Yahoo) send second factors via email. In contrast to the hardware token support, this is a promising finding, because receiving an email does not prove possession of a specific device. For this reason, the NIST does not allow email for 2FA.

Our analysis shows that some of the most popular service providers use 2FA, yet, there is a remarkable amount that does not offer it. Among the services that offer 2FA, SMS-based implementation is still the second most popular variant despite various known attacks and governmental institutions discouraging its use. In the following section, we complement this by presenting a new attack which highlights the problems that arise when sending OTPs for 2FA via SMS.

[1] A web service providing a database of websites that support two-factor authentication, as well as the factors they offer.



Contract Details

Oscar has sent you an email via **Gmail confidential mode:**

Contract Details

This message was sent on May 22, 2019 at 1:19:09 PM PDT
You can open it by blicking the link below.
This link will only work for alice@gmail.com.

Gmail confidential mode gives you more control over the messages you send. Set an expiration time, disable printing or forwarding of a message and more. Learn more

Gmail: Email by Google
Use is subject to the Google Privacy Policy
Google LLC, 1600 Amphitheatre Parkway,
Mountain View, CA 94043, USA
You have received this message because

(a)

Sent by oscar@gmail.com using Gmail confidential mode

To view this email, you must first confirm your identity. A one-time passcode will be sent to •••• ••••••19

(b)

2-Step Verification

This extra step shows it's really you trying to sign in

alice@gmail.com

2-Step Verification

A text message with a 6-digit verification code was just sent to •••• ••••••19

G- Enter the code

☑ Don't ask again on this computer

(c)

Google
Text Message

Sent by oscar@gmail.com using Gmail confidential mode

Passcode was sent to •••• ••••••19

Enter passcode

☑ Remember this passcode for the next 5 minutes

From Messages
162897

(d)

Figure 1: (a) The victim receives the phishing email designed like a Gmail confidential email. (b) After clicking on "View the email" the victim is told that a passcode needs to be provided to be able to read the email. (c) Parallel to the victim requesting the passcode, the attacker initiates the 2FA. (d) The victim receives an SMS from Google and provides the attacker with the second factor code thinking that it is the passcode needed to read the confidential email.

3 Confidential Mode Attack Against 2FA

We now describe a phishing attack that copies the Gmail confidential mode to obtain the OTP sent via SMS as part of Google's two-factor authentication. We start by describing the confidential mode and the underlying threat model, including the hypothesized assumptions. Subsequently, we describe the attack sequence and conclude with a fix which reduces the chances for an attacker, yet, we also show why it is hard to prevent the attack entirely.

3.1 Google Gmail's Confidential Mode

In August 2018, Google introduced the Gmail confidential mode that allows Gmail users to send emails with a fixed expiration date and also to revoke them at any time. To implement the revocation and expiration features, emails in confidential mode only contains a URL pointing to a web server at `confidential-mail.google.com` which serves a website containing the original message. Figure 1a shows an example of such an email.

The sender of the confidential email can also enter the receiver's phone number to enable an additional authentication step. In this case, the receiver is required to request an OTP via SMS (Figure 1b) and enter it (Figure 1d) in order to gain access to the message website. For the attack we misuse this SMS-based version of the confidential mode.

3.2 Threat Model

Our underlying assumption is that we expect the attacker to be in possession of the victim's email address and password. Given that data breaches, unfortunately, occur regularly, we rate this requirement as realistic. Furthermore, the attacker can mimic the look and feel of the Google Gmail confidential mode with the SMS identity verification, which includes a website and an email. However, the attacker does not need to add any additional steps that are not part of the official confidential mode protocol. Only the domain name of the attacker's website differs from that of the actual confidential mode, which can be easily overlooked by users as previous studies on phishing showed [1, 5].

On the other hand, users may be more suspicious because they do not know the confidential mode: it is a niche feature solely offered by Google, and statistics about its usage and acceptance are not available. Hence, succeeding user studies are needed in order to be able to judge the real-world feasibility of the attack.

3.3 Attack Protocol

The necessary steps of the attack are presented in Figure 1. In the following, we describe the attack in detail by going through each of these steps individually:

(a) The attacker sends an email to the victim who owns a Google account and secures it with SMS-based two-factor authentication. This email copies the layout of an email sent via the Gmail confidential mode; thus, the victim has to click on the "View the email" button to read the confidential email.

Table 2: The SMS texts sent by the top 10 web services as part of their 2FA protocol runs and the SMS text of Google Gmail's confidential mode in comparison.

Domain	**2FA SMS text**	**Sender**
google.com	G-123456 is your Google verification code	Google
facebook.com	Use 123456 to log into Facebook	Facebook
microsoft.com	123456 Use this code for Microsoft verification	*Number*
twitter.com	123456 is your Twitter login code. Don't reply to this message with your code	Twitter
linkedin.com	Your LinkedIn verification code is 123456.	Linkedin
apple.com	Your Apple ID Verification Code is: 123456	*Number*
yahoo.com	1234 is your Yahoo verification code	Yahoo
amazon.com	123456 is your Amazon verification code.	Amazon
pinterest.com	Security code for Pinterest: 1234567	AUTHMSG
adobe.com	123456 is your Adobe code. Not you? Change your password	*Number*
	Google Gmail confidential mode SMS text	
	Your Google verification code is 123456	*Number*

(b) When clicking the button, the victim is redirected to a page that is controlled by the attacker, which mimics the authentication step of the confidential mode (cf. Figure 1b). Here the receiver, i. e., the victim, requests an OTP to read the content of the email.

(c) The attacker who is in possessions of the victim's credentials tries to log into the account and initiates the second-factor verification step. Google sends an OTP the victim via SMS.

(d) The victim is asked to enter a passcode sent via SMS to see the content of the confidential email. However, the SMS that the victim receives from Google is part of the two-factor authentication that the attacker just initiated. Table 2 shows the two SMS texts which Google sends in each of the cases. While the format differs, it seems unlikely that the average user can detect that the received code belongs to a 2FA login instead of the Gmail confidential mode.

3.4 Mitigating the Vulnerability

The described attack is possible due to two aspects. On the one hand, reading the text of a confidential email should not require users to click on a link in an email. This issue is already solved when confidential mode emails opened via the Gmail web interface. The web interface skips the link clicking step and shows the message, respectively, the authentication screen directly. In all other case, users have to click on a link which redirects them to a website containing the original message, and they most likely do this without checking if the link contains the correct domain name. An attacker can utilize this inattention for attacks like the one described in this work.

The second issue arises from the fact that Google sends confirmation codes via SMS both for 2FA logins and the confidential mode. One possible solution to at least reduce the chances of a successful attack is to enrich the SMS with information about the reason and correct use of the contained code. However, we do not recommend this solution. One the one hand, finding an appropriate text which users understand is not a straightforward task. Moreover, copying the code from an SMS to an input field is an additional step which users try to complete as quickly as possible to continue with their actual task, in this case, reading the content of a confidential email. Thus, many users will likely search for the code and skip over the SMS text even if it perfectly explains the situation and possible risks.

On smartphones employing iOS, the situation is even worse due to a feature called *Security Code AutoFill*, shown in Figure 1d. The operating system automatically scans incoming SMS for codes and allows users to directly paste them into the input field with a single tap on the keyboard and especially without having to open and read the SMS. While this feature may increase the usability, it prevents users from learning about the purpose of the SMS, which simplifies attacks.

For the described reasons, we conclude that the only countermeasure which completely prevents this and related attacks is to stop using SMS-based 2FA. The average user is most likely neither able nor willed to detect attacks like the one we present; hence, we encourage using other forms of 2FA like software or hardware tokens instead.

4 Related Work

In the context of two-factor authentication, the work of Siadati et al. [17] is most related to ours. They identified the wording of second factor SMS as a weakness and designed a phishing attack with a 50 % success rate against Google's SMS-based 2FA. Siadati et al. also proposed an alternative message with which they were able to reduce the success rate of the attack to 8 %. In our work, we demonstrate a new attack against Google's SMS-based 2FA that is possible even with the proposed message from Siadati et al.

In 2015 Siadati et al. [16] presented another phishing attack targeted against codes sent in SMS. The verification code forwarding attack utilizes the fact that users are not aware of the importance of an SMS code and can thus be lured to forward the code to an attacker via SMS. While the proposed fix to include a warning saying that the received code should not be forwarded may help to mitigate the risks of this specific kind of attack, our attack is still possible as it imitates an official Google web page.

In addition to the described attacks, there are also other approaches that do not employ phishing techniques. Mulliner et al. [11] demonstrated how Trojans installed on the victim's phone can be used to intercept the SMS communication in order to obtain 2FA codes. Konoth et al. [8] and Dmitrienko et al. [6], on the other hand, use malware on the mobile phone or the computer of the victim for a man-in-the-browser attack to obtain two-factor codes. Another scam method with several reported victims in the past [2, 3, 4] is SIM swapping [13]. Here, the attacker uses social engineering to convince operators to port the phone number of the victim to a new SIM card so that the attacker's phone receives SMS codes are henceforth.

5 Conclusion

To protect online accounts against attacks, 2FA has proven to be an effective mean. In our paper, we showed that the most popular online service implements 2FA, yet, there are still numerous services which do not offer it. Across the services who use it, software tokens are the most widely used form of 2FA. SMS-based 2FA is equally popular, and some service providers even rely on it solely although governmental institutions advise against using it and there are various known attacks.

In the second part, we presented a new attack against Google's SMS-based 2FA by exploiting a design flaw in Gmail's confidential mode. Our attack shows how an adversary who mimics the protocol run of the confidential mode can trick users into providing their OTPs sent via SMS. We discussed possible countermeasure and concluded that adding additional information to the SMS only prevents the attack to a certain extent, respectively not at all when considering Apple's Security Code AutoFill feature. Thus, this attack is yet another example in the long list of shortcomings of SMS-based 2FA, which is why service providers should offer other forms of 2FA, e. g., hardware tokens, and consider stop using SMS-based 2FA altogether.

Acknowledgments

This research was supported by the research training group "Human Centered Systems Security" and the MKW-NRW research training group SecHuman, both sponsored by the state of North Rhine-Westphalia, Germany.

References

[1] Mohamed Alsharnouby, Furkan Alaca, and Sonia Chiasson. Why Phishing Still Works: User Strategies for Combating Phishing Attacks. *International Journal of Human-Computer Studies*, 82:69–82, October 2015.

[2] Cody Brown. How to lose $8k worth of bitcoin in 15 minutes with Verizon and Coinbase.com, May 2017. https://link.medium.com/C5Ml2KmLRW, as of August 19, 2019.

[3] Sean Coonce. The Most Expensive Lesson Of My Life: Details of SIM port hack, May 2019. https://link.medium.com/udDy3fGLQW, as of August 19, 2019.

[4] Lorrie Faith Cranor. Your mobile phone account could be hijacked by an identity thief, June 2016. https://www.ftc.gov/news-events/blogs/techftc/2016/06/your-mobile-phone-account-could-be-hijacked-identity-thief, as of August 19, 2019.

[5] Rachna Dhamija, J. D. Tygar, and Marti Hearst. Why Phishing Works. In *ACM Conference on Human Factors in Computing Systems*, CHI '06, pages 581–590, Montréal, Québec, Canada, April 2006. ACM.

[6] Alexandra Dmitrienko, Christopher Liebchen, Christian Rossow, and Ahmad-Reza Sadeghi. On the (In)Security of Mobile Two-Factor Authentication. In *Financial Cryptography and Data Security*, FC '14, pages 365–383, Christ Church, Barbados, March 2014. Springer.

[7] Paul A. Grassi, James L. Fenton, Elaine M. Newton, Ray A. Perlner, Andrew R. Regenscheid, William E. Burr, Justin P. Richer, Naomi B. Lefkovitz, Jamie M. Danker, Yee-Yin Choong, Kristen K. Greene, and Mary F. Theofanos. Digital Identity Guidelines – Authentication and Lifecycle Management: NIST Special Publication 800-63B, June 2017.

[8] Radhesh Krishnan Konoth, Victor van der Veen, and Herbert Bos. How Anywhere Computing Just Killed Your Phone-Based Two-Factor Authentication. In *Financial Cryptography and Data Security*, FC '16, pages 405–421, Christ Church, Barbados, February 2016. Springer.

[9] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In *Symposium on Network and Distributed System Security*, NDSS '19, San Diego, California, USA, February 2019. ISOC.

[10] Ariana Mirian, Joe DeBlasio, Stefan Savage, Geoffrey M. Voelker, and Kurt Thomas. Hack for Hire: Exploring the Emerging Market for Account Hijacking. In

International World Wide Web Conference, WWW '19, pages 1279–1289, San Francisco, California, USA, May 2019. ACM.

[11] Collin Mulliner, Ravishankar Borgaonkar, Patrick Stewin, and Jean-Pierre Seifert. SMS-Based One-Time Passwords: Attacks and Defense. In *Detection of Intrusions and Malware, and Vulnerability Assessment*, DIMVA '13, pages 150–159, Berlin, Germany, July 2013. Springer.

[12] National Cyber Security Centre. Setting Up Two-Factor Authentication (2FA), August 2018. https://www.ncsc.gov.uk/guidance/setting-two-factor-authentication-2fa, as of August 19, 2019.

[13] Merve Sahin, Aurélien Francillon, Payas Gupta, and Mustaque Ahamad. SoK: Fraud in Telephony Networks. In *European Symposium on Security and Privacy*, EuroSP '17, pages 235–250, Paris, France, April 2017. IEEE.

[14] Bruce Schneier. NIST is No Longer Recommending Two-Factor Authentication Using SMS, August 2016. https://www.schneier.com/blog/archives/2016/08/nist_is_no_long.html, as of August 19, 2019.

[15] John Scott-Railton and Katie Kleemola. London Calling - Two-Factor Authentication Phishing From Iran, August 2015. https://citizenlab.ca/2015/08/iran_two_factor_phishing/, as of August 19, 2019.

[16] Hossein Siadati, Toan Nguyen, and Nasir Memon. Verification Code Forwarding Attack. In *International Conference on Passwords*, PASSWORDS '15, pages 65–71, Cambridge, United Kingdom, December 2015. Springer.

[17] Hossein Siadati, Toan Nguyen, Payas Gupta, Markus Jakobsson, and Nasir Memo. Mind your SMSes: Mitigating Social Engineering in Second Factor Authentication. *Computer & Security*, 67:14–28, March 2017.