Vulnerability Exploiting SMS Push Notifications

Rina Shibayama, Hiroaki Kikuchi
Graduate School of Advanced Mathematical Sciences, Meiji University, Tokyo, Japan

***Abstract*—SMS (Short Message Service)-based authentication is widely used as a simple and secure multi-factor authentication, where OTP (One Time Password) is sent to user's mobile phone via SMS. However, SMS authentication is vulnerable to Password Reset Man in the Middle Attack (PRMitM). In this attack, the attacker makes a victim perform password reset OTP for sign-up verification OTP. If the victim enters OTP to a malicious man-in-the-middle site, the attacker can overtake the victim's account.**

We find new smartphone useful functions may increase PRMitM attack risks. SMS push notification informs us an arrival of message by showing only beginning of the message. Hence, those who received SMS OTP do not notice the cautionary notes and the name of the sender that are supposed to show below the code, which may lead to be compromised. Auto-fill function, which allow us to input authentication code with one touch, is also vulnerable for the same reason.

In this study, we conduct a user study to investigate the effect of new smartphone functions incurring PRMitM attack.

I. Introduction

Multi-factor authentication combines two or more authentication methods in order to prevent unauthorized access. Passwords are used as the first authentication method. As the second method, Short Message Service (SMS) is widely used. SMS is a representative service of sending short messages to a mobile phone.

The SMS-based authentication has been used in two typical cases. (1) If a user forgets his password, he requests a password reset and safely receives one-time password via a SMS. (2) With an enrollment process, a service provider confirms that the requesting user receives SMS text at the correct phone number.

In 2017, Gelernter et al. found that SMS authentication is vulnerable to the password reset man in the middle attack (PRMitM) [1]. In this attack, an attacker makes a victim enter a verification code sent by SMS to a malicious man in the middle site without being aware that the code is for password reset and then the attacker overtakes the victim's account. To prevent the PRMitM attack, they suggest explicitly indicating the purpose of the code, e.g., "password reset", and the receiver's name in SMS. After Gelernter published their work, many websites have fixed their website so that the PRMitM is no longer available.

However, we find a new account hijacking threat that exploits the enhanced features deployed in the latest smartphone OSs. It is a "push notification" feature that lets owner to receive an arrival of new SMS message by showing the beginning part of the SMS message. Notification is very useful for saving time in checking SMS message. However, it could incur a serious risk of compromised. If the user learns the verification code via the notification, he may not read the entire message. He loses a chance to find a warning note written at the bottom of the SMS message. Thus, since he does not know that the code is for a password reset of a fake sender, he enters the code to the fake website.

We argue that SMS push notification increases a PRMitM risk. We come up with some questions; *Does PRMitM risk decrease if a presence of warning is given? Does the risk depend on whether the warning is written at the top of the message or not? What if the warning message is written in non-native language?*

Our hypothesis is that the risk of PRMitM attack increases if a user gets OTP from push notification without reading the whole SMS text. To verify our hypothesis, we conduct a user experiment using fake websites testing the password to be hacked in simulated websites. We also identify the demographic properties of vulnerable subjects, e.g., a subject' skill level and a security awareness by means of the questionnaire. We suppose that ICT skills and security knowledge would help effectively in mitigating this attack.

Subsequently, we quantify a *successful attack rate* of the PRMitM in various conditions, e.g., specifying a warning at the top and the bottom of SMS message. We report the results of these experiments and the analysis of statistical hypothesis testing to confirm that SMS push notification increases the risk of compromised. Our experiment reveals that some subsets of subjects have higher successful attack rate than others. For example, elder people are more likely to be compromised. We discuss some potential factors that drives a successful attack. Based on our findings from the observation, we also propose some countermeasures to prevent the PRMitM attacks.

With this work, we make the following contributions:

1) a new vulnerability of SMS push notification enhanced in the latest smartphone OSs that incurs account hijacking,
2) user experimental results using fake websites that quantify the risk of the attack,
3) an analysis of statistical hypothesis analysis to identify the most significant factors of the vulnerability and the human factors that are likely to be suffered, and
4) some countermeasures against the new attacks.

This paper is organized as follows. In Section 2, we give some backgrounds of this study, including the PRMitM attack, the subsequent study, and a questionnaire survey that estimate the security knowledge. In Section 3, we define the new vulnerability abused the SMS notification and auto-fill features. The difference of behavior among operating systems is also introduced. Section 4 describes the objective, the methodology and the results of online user experiment using three fake websites. We discuss the primary factors of vulnerability and particular attributes that are subjects to be attacked in Section 5. Based on our analysis, we propose a set of countermeasures that mitigate the attack in Section 6. We give ethical and privacy considerations in section 7. In Section 8, we conclude out work and show future studies.

II. Background

A. Related Studies of a password-based Authentication

A password is widely used for user authentications though it is known as vulnerable to some attacks such as a man-in-the-middle attack [11], [12], a password guessing attack [9] and a phishing and a spyware attacks [10]. Password authentication has another drawback that users easily forget their passwords [15].

To reset a password a user needs to verify identity by email, voice call, SMS, etc. Among them, SMS is the most widely used method, in which the user is identified by receiving an OTP via SMS [16]. However, SMS OTP authentication has significant security implications.

To make SMS OTP authentication more convenient, Android has a feature called API SMS Retriever API, with which user can perform SMS-based verification in the Android app automatically, without requiring the user to manually type verification codes. This feature poses a risk of having your OTP stolen by local apps [19].

SMS OTP authentication is also vulnerable to brute force attacks [18]. Ideally, time limit of the authentication code and the limit of number of times the code can be entered should be set. Insufficient randomness and insufficient length of authentication codes for some services is also designated [17]. Another implication is a PRMitM attack [1].

B. PRMitM attack

A new attack exploiting password-based authentication is a password reset man-in-the-middle (PRMitM) attack proposed by Gelernter et al. in 2017 [1].

Fig. 1 shows the process of a PRMitM attack. Suppose that user U has an account on target site A and the attacker prepared a man-in-the-middle site B. To sign up for site B, user U gives information such as name, email address, and phone number. With this information, the attacker redirects it to site A to initialize the user's password. Then, site A sends back a password reset code by SMS to user U to confirm that the request is from the user U. In the process of signing up for site B, user U mistakenly assumes that it is the authentication code for B's registration and enters the reset code to B. By this procedure, the attacker resets the password registered to A and overtakes the victim user's account.

Gelernter et al. claim that cautionary notes that the code is for password reset and indicating explicitly the name of the sender in a SMS are the basic countermeasures against PRMitM attacks [1]. Sending a URL for the password reset instead of a reset code is also recommended.

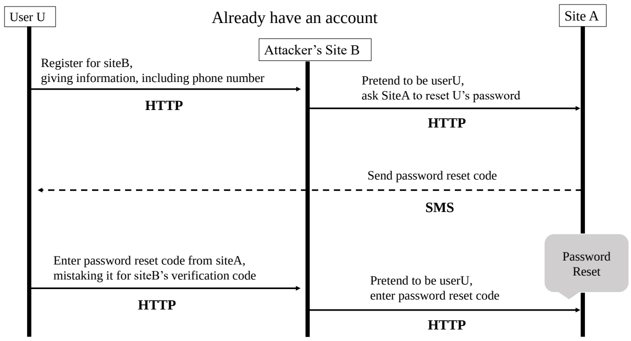

User U
Already have an account
Site A
Attacker's Site B
Register for siteB,
giving information, including phone number
HTTP
Pretend to be userU,
ask SiteA to reset U's password
HTTP
Send password reset code
SMS
Password
Reset
Enter password reset code from siteA,
mistaking it for siteB's verification code
HTTP
Pretend to be userU,
enter password reset code
HTTP

Fig. 1. PRMitM attack process

Although these basic countermeasures reduce the successful attack ratio, they do not sufficiently prevent PRMitM attack. In previous studies, many efforts were done to make warninigs effective [13], [14]. Since it takes users a few seconds to click notifications [3], warnings should be quickly told to the users. In another case, dark pattern interface makes users perform actions they did not mean to do without users noticing it [6]–[8].

Sasa et al. suggested using an *alphanumeric reset* for mitigating the PRMitM attack, and conducted user experiments [2], [5]. They argue that an alphanumeric code takes more effect of reducing attack risk than numeric reset code because the code part is emphasized in blue letters. They also suggested a threat of *long-winded attacks*, where a user first receives a long SMS message with a confirmation code and is required to enter it and then receives a message with the real password reset code. They expected that the attack risk would increase because of the familiarity of entering the code once. The results of user experiments showed no significant differences in the successful attack ratio between numeric and alphanumeric codes and between long and short SMS texts.

C. Security Behavior Intentions Scale

Security Behavior Intentions Scale (SeBIS) is a measure of security awareness developed by Egelman et al. in 2015 [4]. The instrument consists of 16 questions in four categories; a device securement, a password generation, a proactive awareness, and an updating to quantify the security awareness. Subjects respond in a 5-point Likert scale.

III. New SMS-based Password Reset Vulnerability

A. Overview

Password reset SMS message consists of a reset code, a service name and a description. Table I shows the samples of SMS texts for password reset sent by major websites. Some services such as Amazon use a password reset URL instead of code.

In the top eight sites in Japan listed in Alexa[§], four services write the code at the beginning of the SMS message, two sites specify the service name at the first, and the other two services do not use SMS authentication when resetting passwords.

Fig. 2 shows an example of SMS message for password reset. Typical behavior is to open a SMS message and read

TABLE I
EXAMPLES OF PASSWORD RESET SMS TEXT

Service	SMS Texts
Google	`X-XXXXXX is your Google confirmation code.`

Amazon	`amazon.co.jp/XXXX...`
	`Attempt to change your Amazon JP password.`
	`Please tap the link to respond.`

Twitter	`Verification code to reset your Twitter password is XXXXXX. Please do not reply to this message.`

the whole text. However, there are alternative ways to retrieve the code without fully opening SMS.

- **Browsing SMS message** Open the full text of SMS in the application and browse the full message (Fig. 2).
- **Listing messages** Check the list of the digests of the first few lines of the SMS text shown in the message listing in the application (Fig. 3).
- **Push notification** Check the push notification of arrival of a SMS message with the short summary of the message. Note push notification is shown whenever using other applications (Fig. 4). Only the first few lines of the message are shown (summarizing way sightly differs between OSs).

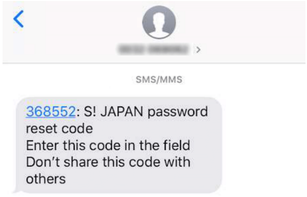
In using these functions, we read only the beginning of the SMS message and skip reading the remaining message, which may lead to suffer the PRMitM attack. This increases a probability of an attack because the warnings and name of the sender below the code are not noticed.

If registration is performed with a PC, the authentication code is displayed on the lock screen of a smartphone. As shown in Fig. 5, the iPhone shows the beginning or the whole SMS body, whereas the Android does not show the whole text by default.

In addition, iPhone currently has an ***auto-fill function*** that automatically recognizes an authentication code sent in SMS and inputs it in behalf of the owner with a single touch. Fig. 6 illustrates how the reset code `601633` is forwarded to browser automatically. By simply touching the code on the keypad, a user can enter the code without even having a chance to check the SMS message without checking the service name or warning. Accordingly, any of the useful features enhanced to improve the usability can increase the risk of PRMitM attacks.

B. Man-in-the-middle attacks that exploit push notifications

This vulnerability does not always lead to user-account hijacking. The vulnerable probability varies depending on the sign-up device, device model, and user's notification configuration. Careful users may be aware of the attack. Vulnerability of PRMitM depends on small difference between models, such as number of digest lines, and whether the full text is checked on the lock screen or not.

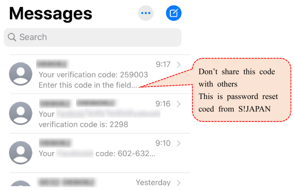


[redacted]

SMS/MMS

368552: S! JAPAN password reset code
Enter this code in the field
Don't share this code with others

Fig. 2. Browsing full SMS message (iPhone)



Messages

Search

[redacted] 9:17
Your verification code: 259003
Enter this code in the field...

[redacted] 9:16
Your [redacted]
verification code is: 2298

[redacted] 9:10
Your [redacted] code: 602-632...

[redacted] Yesterday

Don't share this code with others
This is password reset coed from S!JAPAN

Fig. 3. SMS message listing (iPhone)

Vulnerability of PRMitM depends on individuals. If a user has sufficient security knowledge, he will be able to avoid this attack by carefully checking the SMS text and entering the code appropriately. Those who are not familiar with smartphones can be more vulnerable.

IV. EXPERIMENT

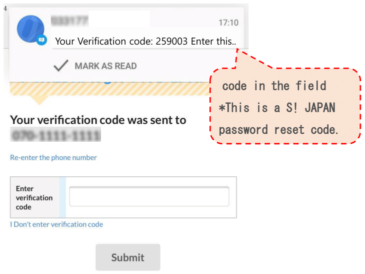
A. Purpose

We conducted an experiment to reveal the following purposes:

1) Examining the effect of warnings to prevent attacks.
2) Examining the impact of new smartphone features such as push notifications on PRMitM attacks, prefixing warnings and how users check and input the code.
3) Seeing how users are compromised in the different language of SMS messages.
4) Revealing the human characteristics vulnerable.

B. Method

We conduct an experiment with subjects who were collected via the crowdsourcing services, CrowdWorks[*] and Lancers[†]. In our experiment, 81 subjects are instructed to sign up for dummy websites using SMS authentication. To create an account, a subject submits a nickname, password and a



[redacted] 17:10
Your Verification code: 259003 Enter this..

MARK AS READ

code in the field
*This is a S! JAPAN password reset code.

Your verification code was sent to
[redacted]

Re-enter the phone number

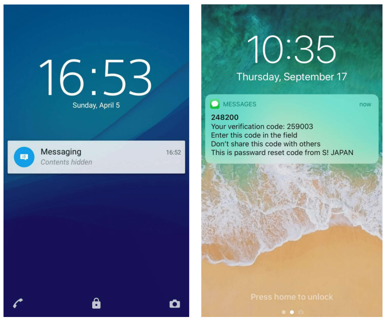
Enter verification code

I Don't enter verification code

Submit

Fig. 4. Push notification(Android)

[§] Alexa, https://www.alexa.com/topsites/countries/JP



248200
Your verification code: 259003
Enter this code in the field
Don't share this code with others
This is passward reset code from S! JAPAN

Fig. 5. notification on the lock screen(Android, iPhone)

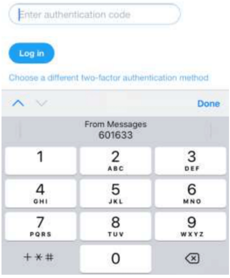

Enter authentication code
Log in
Choose a different two-factor authentication method
Done
From Messages
601633
1
2 ABC
3 DEF
4 GHI
5 JKL
6 MNO
7 PQRS
8 TUV
9 WXYZ
+*#
0

Fig. 6. Auto-input (iPhone)

phone number. We used "Twilio"[‡] to send SMS messages from the experimental sites to the subjects. We did not store password nor phone number. We divide the set of subjects into five groups and send different messages as in Table III to investigate the causes of the successful attack.

The steps of the experiment are as follows:

1) **Instruments and gets consent of the experiment.** Before start experiments, subjects are required to consent to the privacy policy and the experimental purpose (usability study of sign-ups using SMS authentication).
2) **Signup for enrollment of three websites.** We instruct subjects sign up for three toy websites with SMS authentication. The three sites and their purposes are summarized in Table II. The first session is for registration only. Second session is for user registration and testing SMS authentication. In the third session, we carried out the PRMitM attack to hijack victim's account of the first site S!JAPAN. We send different SMSs to different groups. The subjects are instructed in advance to cancel if they have any doubts about their enrollment. If they enter the password reset code from S!JAPAN, their accounts are supposed to be hijacked.
3) **Questionnaires on usability.** For every signup, the subjects are asked two questions: "Is the registration form easy to use?" and "Do you feel that the site is secure?" by using the Likertscales.
4) **Questionnaires on knowledge.** After all, they answered the SeBIS questions to measure security awareness and computer skills. We observe their HTTP headers to see the device they used.

TABLE II
PROCEDURE OF THREE WEBSITES SIGN-UPS IN THE EXPERIMENT

	Service name	Purpose	Verification code	Expected Behavior
1	*S! JAPAN*	test sign-up	–	–
2	*Cowtter*	test OTP	Cowtter verification code	submit the code
3	*Majebook*	PRMitM attack	S!JAPAN password reset code	cancel

C. Definition of victim

On the third signup, careful subjects may find an inconsistency between the signup site's name and the sender of the OTP. Alternatively, they find a wrong message received as "password reset". To avoid being attacked, they should choose the option "cancel input". However, careless subjects do not notice the inconsistency, and then would mistakenly input the password reset code as the authentication code for registration.

A subjects who submit OTP for *S!Japan* during the *Majebook* signup is regarded as a ***victim*** of the attack. The fraction of victims in a given condition is defined as ***successful attack rate***. For example, suppose that 14 subjects out of 19 enter the reset code. In this case, the successful attack ratio is $R = 14/19$.

D. Results

The successful attack ratio varied with different warning positions and languages are shown in Table IV. No subjects (out of 7) submitted the OTP when received Japanese (native) SMS code and warnings indicated at the top (type 3). The average successful attack ratio was 61.7%.

Table VI shows the results of the successful attack ratio for each device. We conduct the Chi-square hypothesis testing that successful attack ratio is independent from the subject's device. Devices are identified from the HTTP user agents[*].

Table V shows the mean and the standard deviations (SDs) of the usability and trust-worthy for three toy sites (1: very difficult to use/ not reliable at all; 7: very trustful/very easy to use). The standard deviations are small, less than 2 for all items. There is no significant difference between the sites.

Table VII shows the main reasons for the canceling of the reset code. The most common reason for the cancellation was noticing of wrong service name specified.

Table VIII shows the results of the questionnaire on whether subjects agree to submit their phone number to the sites they have ever seen, and well-known sites.

Table IX shows the successful attack ratio for way of checking and submitting OTP. The most common way was a manual input, chosen by 64 out of 81 subjects. The method of checking

[*] CrowdWorks, https://crowdworks.jp/
[†] Lancers, https://www.lancers.jp/
[‡] Twilio, https://twilio.kddi-web.com/

[*] We also asked a questionnaire on the device they use and found that two subjects answer wrongly. Therefore, we exclude some subjects who did not correctly know their device OS.

TABLE III
Examples of SMS text

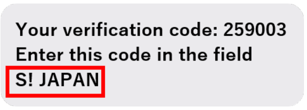
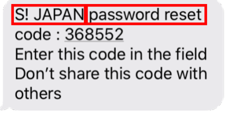
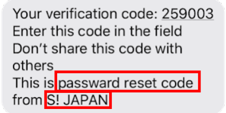
	Type 0: No warning	Type 2: Warning at the top	Type 4: Warning at the bottom
Reset message	 Your verification code: 259003 Enter this code in the field S! JAPAN	 S! JAPAN password reset code : 368552 Enter this code in the field Don't share this code with others	 Your verification code: 259003 Enter this code in the field Don't share this code with others This is passward reset code from S! JAPAN
Typical behavior	Misunderstanding the code is for sign up without the usage of the code.	Noticing both the servicer and password reset at the top.	Quickly finding code and not noticeing any details of the text.
Potential risk	middle	low	high

TABLE IV
attack rates by SMS types

Type	SMS feature		accept	Total	attack
	Warning	Language			rate[%]
0	None	Japanese	14	19	73.7
1	Middle	Japanese	15	19	78.9
2	Middle	English	16	20	80.0
3	Beginning	Japanaese	0	7	0.0
4	Beginning	English	10	16	67.9

TABLE V
Statics of usability and Ressurance for each experimental site

Websites	Usability		Reassurance	
	Mean	SD	Mean	SD
S! JAPAN	4.07	1.70	5.78	1.12
Cowtter	5.91	1.08	4.95	1.61
Majebook	5.72	1.33	4.22	1.94

TABLE VIII
Statistics of resistance to phone number entry

	accept	Cancel	SD
Well-known services	4.02	3.73	1.79
Services you never seen before	2.65	2.27	1.52

TABLE IX
Successful attack rates by the ways of checking and inputting

		accept	Total	attack rate[%]	χ	p
Input	manual	44	64	78.8	1.70	0.428
	copy paste	7	10	70.0		
	auto-fill	4	6	66.7		
Check	browsing	27	40	67.5	1.74	0.418
	listing	6	11	54.5		
	notification	22	29	75.9		

the code includes browsing message, listing messages, push notification and others. For each method for checking the code, 40 out of 81 subjects chose “browsing” while 29 chose push notification.

Table X shows the successful attack rates for gender and age. The successful attack rates exceeded 80% for those in their 20s and 50s or older.

Table XI shows the statistics of computer skills of the subjects. Table XII shows the result of the SeBIS. 16 questions in the SeBIS were divided into four categories: device securement, password management, proactive awareness and software updating. The sums of scores for each of categories are analyzed.

The number of subjects who received SMS warning at the top is smaller than any other types. Although the same number of subjects were assigned to each type at random, some SMS messages were not sent because of errors.

E. Analysis

Table XIII shows the results of the hypothesis test of independence of SMS type from the chi-square test with one degree of freedom. The differences between the presence and

TABLE VI
Successful attack rates by device models

Device	accept	Total	attack rate	χ	p
iPhone	21	30	70.0	3.11	0.37
Android	23	31	74.2		
PC	11	20	55.0		

TABLE VII
Reasons of cancel

Reason	N
A lack of understanding of the message	7
Written as S! JAPAN	11
Written as password reset	6
Unreliable sites	1
Written in English	1

TABLE X
Successful attack rates by attributes

		accept	total	attack rate[%]	χ	p
Gender	Male	32	44	72.7	1.03	0.319
	Female	23	37	62.2		
Age	Under 20	0	1	0.0	13.26	0.021*
	20s	14	17	82.4		
	30s	14	28	50.0		
	40s	15	22	68.2		
	50s	8	9	88.9		
	Over 60	4	4	100.0		

TABLE XI
Computer skill measuring questionnaire and its results (mean)

	Question	accept	cancel	SD
1	Have you ever installed an OS yourself?	1.51	1.50	0.61
2	Have you ever set up your own network?	1.18	1.23	0.40
3	Have you ever created your own web page?	1.24	1.23	0.42

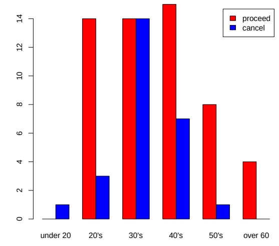


Fig. 7. Counts of subject against PRMitM attack

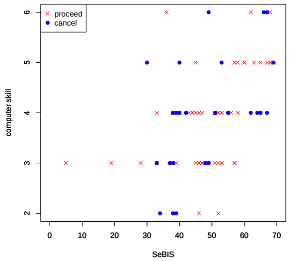


Fig. 8. Scatterplot of SeBIS, Computer Skills and PRMitM Attack Damage

absence of warnings, Japanese (native) and English (non-native), and warning positions top or bottom are significant level (less than 0.05).

Table IX shows the chi-square test for independence (degree of freedom 2). There was no significant difference in any of the input nor check method ($p = 0.428$, $p = 0.418$).

Table X shows that the tests of independence revealed no significant difference in gender ($p = 0.319$) but a difference in age at a significance level ($p = 0.021$).

Table VI shows the results of the chi-square test of independence where the degrees of freedom is 2. There is no statistically significant difference among devices, i.e., iPhone, Android, and PC ($p = 0.374$).

Security awareness (SeBIS) and computer skills are distributed over a range of 0-80 and 0-6, respectively. Figure 8 shows a scatterplot of the SeBIS and skill scores. Table XIV shows the statistics of the SeBIS index and skills and the results of the Weltch's hypothesis test.

To identify the main factors that make an attack successful, we performed a logistic regression analysis to derive a logistic model in the form

$$\frac{p}{1-p} = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_{19} x_{19}$$

. Table XII shows the results. Among the 4 categories, there were no indicators that had a significant impact. Accept (vulnerable) /cancel (secure) is the objective variable, and the explanatory variables are the types of SMS (i.e., without warning x_1, top warning x_2, and bottom warning x_3); usability of the three websites, $x_{1,1}$, $x_{2,1}$, and $x_{3,1}$; a trust-worthy, $x_{1,2}$, $x_{2,2}$, $x_{3,2}$; responses to questions in the questionnaire, x_{q1},$x_{q2} \ldots x_{q7}$; and the SeBIS for each of categories, $x_{c1} \ldots x_{c4}$. The results are shown in TableXV.

The adjusted odds ratio (OR) for SMS types warning at the top to SMS without warnings is

$$\frac{\text{(attack rate with top warning)/(cancel rate with top warning)}}{\text{(attack rate without warning)/(cancel rate without warning)}} = e^{-2.435} = 0.088$$

This means that SMS warning at the top reduces the attack risk by the 0.088 of the odds of attack without warning. Likewise, people with high security awareness for password generation tend to be less vulnerable to attack.

V. Discussion

A. Risk reduction factors

1) SMS message type: As shown in Table XIII, a significant difference between the presence and absence warnings in the successful attack ratio was found including the sender's name and the use of code. Therefore, we find placing warnings in the SMS is effective, as same as previous studies [1], [2].

When warnings are indicated at the top of the message, the number of subjects who accepts the attack was less than that of the warnings at the bottom. Hence, we claim that explicitly indicating the name of the service and the use of the authentication code reduces the attack risk certainly.

Our subjects are non-English native speakers. Hence, receiving a SMS in English, 10 out of 16 subjects wrongly entered the OTP, whereas none out of 7 subjects entered the OTP when receiving a SMS in Japanese. Therefore, even if the users do not understand the content of the message at first glance, they do submit the OTP. Only one user finds strange that the SMS was written in English and suspends submitting the code.

2) Way for Checking and inputting the code: There was no significant difference between three inputting methods: manual, copy-and-paste, and auto-input (Table IX). In the case of an "auto-fill", we predict that all subjects would input the code because an auto-fill method proceeds without checking the SMS text. However, only 4 out of 6 subjects did. One of possible reasons is that some subjects double check the push notifications.

Table IX shows that there was no significant difference in ways of checking the verification code: "browsing", "listing messages" and "notification". When a warning and service name are written at the bottom of the message (types 1 and 2), we guess that most subjects would enter the code because only the first two lines of the message are shown in the message listing. However, the successful attack rates were only 54.5% and 75.9% for the listing and notification, respectively. One reason for this may be that some subjects wrongly chose "listing messages" as an answer in the questionnaire, even though they actually "browsing" and checked the full text. Our descriptions may not be interpreted as we intended.

3) Demographic Attributes: Table X shows that the successful attack rates was higher in their 20s and 50s and older. We suppose it is because those in their 20s were familiar

TABLE XII
SeBIS questions and result of logistic regression

	Questions	e^{β}	p
1	I set my computer screen to automatically lock if I don't use it for a prolonged period of time.	0.996	0.933
2	I use a password/passcode to unlock my laptop or tablet.		
3	I manually lock my computer screen when I step away from it.		
4	I use a PIN or passcode to unlock my mobile phone.		
5	I do not change my passwords, unless I have to.	1.038	0.607
7	When I create a new online account, I try to use a password that goes beyond the site's minimum requirements.		
8	I do not include special characters in my password if it's not required.		
9	I use different passwords for different accounts that I have.		
10	When someone sends me a link, I open it without first verifying where it goes.	0.981	0.837
11	I know what website I'm visiting based on its look and feel, rather than by looking at the URL bar.		
12	I submit information to websites without first verifying that it will be sent securely (e.g., SSL, "https://", a lock icon).		
13	When browsing websites, I mouseover links to see where they go, before clicking them.		
14	If I discover a security problem, I continue what I was doing because I assume someone else will fix it.	1.014	0.870
15	When I'm prompted about a software update, I install it right away.		
16	I try to make sure that the programs I use are up-to-date.		
18	I verify that my anti-virus software has been regularly updating itself.		
	Total score	1.568	0.656

TABLE XIII
Successful attack rates and test result by SMS features

type	feature	Input	Total	Damage rate[%]	χ	p
0	No Warning	15	20	75.0	11.81	0.001***
3	With a warinig	0	7	0.0		
3	Japanaese	0	7	0.0	7.74	0.005***
4	English	10	16	62.5		
3+4	Upper	10	23	43.5	8.37	0.004***
1+2	Lower	31	39	79.5		

TABLE XIV
Statistics of SeBIS and skills

	Mean		SD	t	p
	accept	cancel			
SeBIS	49.6	48.8	12.0	0.132	0.896
skill	3.93	3.96	1.01	-0.266	0.792

TABLE XV
Logistic regression result

	Estimateβ	Std. Error	z value	$Pr(>\lvert z\rvert)$
(Intercept)	8.082	5.521	1.464	0.143
x_2	-6.673	2.440	-2.734	0.006***
x_3	-2.244	1.444	-1.554	0.120
x_4	-4.776	1.674	-2.853	0.004***
$x_{1,1}$	-1.381	0.714	-1.934	0.053*
$x_{1,2}$	0.617	0.394	1.569	0.117
$x_{2,1}$	2.372	0.930	2.550	0.011*
$x_{2,2}$	-1.303	0.445	-2.931	0.003***
$x_{3,1}$	-1.294	0.508	-2.546	0.011*
$x_{3,2}$	0.792	0.286	2.766	0.006***
x_{q1}	0.993	0.504	1.971	0.049*
x_{q2}	-2.283	1.254	-1.821	0.069*
x_{q3}	-1.604	0.785	-2.042	0.041*
x_{q7}	0.643	0.396	1.626	0.104
x_{q8}	3.583	1.773	2.021	0.043*
x_{c1}	-0.043	0.058	-0.749	0.454
x_{c2}	0.302	0.576	0.525	0.600

with a verification code and those in their 50s and older were distracted by the SMS verification code itself.

4) Device models: The successful attack rates among iPhones, Androids, and PCs were not statistically significant (Table VI). When a notification is shown on the lock screen, iPhone shows the first two lines of the SMS text, while Android does not show the text (only notify the arrival of the messages). We expected that the amount of information shown in the notifications take some effects on attack risk, but there was no significant difference.

5) Security awareness and computer skills: Although there is a slight positive correlation between SeBIS and skill, as shown in Fig. 8, the results are widely distributed, and security awareness/skill and successful attack rates are considered to be independent. The results of the Weltch's test in Table XIV also showed no significant difference in the mean scores of SeBIS and skills. We expected that subjects with knowledge in security and ICT would be less vulnerable to attack, but neither of these factors helps against the attack.

B. Limitations

We conducted an experiment in Japan with Japanese subjects. Therefore, our results show a trend in Japan, and do not necessarily apply to the other countries. The number of subjects is not enough. The more detail properties of subjects are not clear such as their language ability and their occupation.

The gap between real environment and experiment is another concern. Knowing the experiment is conducted by an academic study, the subjects may trust too much and feel the experiment safe.

Another concern with this study is dishonest subjects. We rely on subjects' self-reports on how they viewed the message, but some subjects mistakenly reported. Observing subject' behavior should be ideal.

VI. Countermeasure

A. Developers

Based on our experimental results, we propose three countermeasures for service developers to prevent PRMitM attack.

1) Disable auto-fill function at different domains.
2) Specify the servicer and the purpose explicitly in SMS.
3) New smartphone feature to observe URL of a website in the SMS message and enables auto-entry only when the URL in the SMS matches the one that the user is trying to enter the code into. This is the method proposed by Apple, and while it prevents automatic entry into third party sites, it may cause the user to enter the code manually.

B. Users

We suggest two practices for users to prevent PRMitM attacks.

1) Understand the purpose for user authentication.
2) Carefully check the SMS servicer and the usage of code before submitting.

The experimental result shows that the even users who have high security awareness for password generation tend to be not resistant to PRMitM attack. Understanding the system of password resetting or other authentication may help them to notice the attack. We suggest checking the SMS sender and the usage of code before submitting even while using auto-fill function can decrease the risk of PRMitM.

VII. Ethical and Privacy Consideration

We called for 81 subjects via crowdsourcing services CrowdWorks and Lancers to conduct user experiments. Before the experiment, subjects read the explanation about purpose of our experiment and participated with consent of privacy statement. In this experiment, we used the workers' phone numbers only to make proxy SMS service to send them SMS text, and did not store these personal data. We note that some services' agreement prohibits clients from obtaining direct contact information of workers, such as phone numbers.

VIII. Conclusion

In this paper, we have proved that a push notification feature that displays the top of SMS text incurs an account hijacking. The results of the user experiment show that the absence of warnings, warning at the bottom and non-native messages are vulnerable for PRMitM attack. As a result of logistic regression, the $OR = 0.088$ for SMS message warning at the top to message without warning, reducing the odds to about 1/12th. Whereas warning at the bottom of the SMS messages brings no significant differences. Based on the experimental results, it is important to indicate warnings appropriately in order to avoid a PRMitM attack.

We also found that users' age has a certain effect reducing successful attack rates. In contrast, the way of checking and entering the code does not work significantly on the successful attack ratio.

In our experiment, the usage rate of the code auto-fill function was less than 10%, but the attack risk may increase if it becomes widespread in the future.

References

[1] Nethanel Gelernter, Senia Kalma, Bar Magnezi, Hen Porcilan, The Password Reset MitM Attack, IEEE Symposium on Security and Privacy (SP), pp. 251-267, 2017.

[2] Kota Sasa, Hiroaki Kikuchi, Impact Assessment of Password Reset PRMitM Attack with Two-Factor Authentication, IEEE Conference on Dependable and Secure Computing, pp. 90-97, 2018.

[3] Alireza Sahami Shirazi, Niels Henze, Tilman Dingler, Martin Pielot, Dominik Weber, Albrecht Schmidt, Large-Scale Assessment of Mobile Notifications, CHI One of a CHInd, pp.3055–3064, 2014.

[4] Serge Egelman, Eyal Peer, Scaling the Security Wall Developing a Security Behavior Intentions Scale (SeBIS), ACM Conference on Human Factors in Computing Systems, pp. 2873-2882, 2015.

[5] Kota Sasa, Hiroaki Kikuchi, Impact Assessment of Password Reset PRMitM Attack with Two-Factor Authentication, Journal of Internet Technology, vol. 20, no. 7, pp. 2297-2306, 2019.

[6] Linda Di Geronimo, Larissa Braz, Enrico Fregnan, Fabio Palomba, Alberto Bacchelli, UI Dark Patterns and Where to Find Them: A Study on Mobile Applications and User Perception, ACM CHI 2020, April 25–30, 2020.

[7] Madison Fansher, Shruthi Sai Chivukula, and Colin M Gray, dark patterns: UX Practitioner Conversations About Ethical Design, Extended Abstracts of the Conference on Human Factors in Computing Systems, 2018.

[8] Colin M. Gray, Yubo Kou, Bryan Battles, Joseph Hoggatt, and Austin L. Toombs, The Dark (Patterns) Side of UX Design, Proceedings of the Conference on Human Factors in Computing Systems, 2018.

[9] Wang, D., Cheng, H., Wang, P., Yan, J., Huang, X., A security analysis of honeywords, In: Proceedings of the 25th Annual Network and Distributed System Security Symposium, 2018.

[10] Paterson K.G., Stebila D., One-Time-Password-Authenticated Key Exchange. In: Steinfeld R., Hawkes P. (eds) Information Security and Privacy. ACISP 2010. Lecture Notes in Computer Science, vol 6168. Springer, 264-281, 2010.

[11] Ryu, G.; Kim, S.-H. Choi, D., Implicit Secondary Authentication for Sustainable SMS Authentication, Sustainability 2019, 11, 279, 2019.

[12] Shetty, R., Grispos, G., Choo, K.K.R., Are you dating danger? An interdisciplinary approach to evaluating the (in) security of android dating apps. IEEE Trans. Sustain. Comput., 197-207, 2017.

[13] Devdatta Akhawe, Adrienne Porter Felt, Alice in Warder-land: A Large-Scale Field Study of Browser Security Warning Effectiveness, USENIX Security Symposium. August 14–16, 2013.

[14] Anthony Vance, David Eargle, The Fog of Warnings: How Non-essential Notifications Blur with Security Warnings, Symposium on Usable Privacy and Security, 2019.

[15] J. J. Yan, A. F. Blackwell, R. J. Anderson, A. Grant, Password Memorability and Security: Empirical Results, IEEE Security and Privacy, Vol. 2, No. 5, pp. 25-31, September-October, 2004.

[16] Rishabh Dudheria, Assessing Vulnerability of Mobile Messaging Apps to Man-in-the-Middle (MitM) Attack, I.J. Computer Network and Information Security, 2018, 7, 23-35.

[17] Siqi Ma, Runhan Feng, Juanru Li, Yang Liu, Surya Nepal, Diethelm, Elisa Bertino, Robert H. Deng, Zhuo Ma, and Sanjay Jha. 2019. An empirical study of SMS one-time password authentication in Android apps. In Proceedings of the 35th Annual Computer Security Applications Conference (ACSAC '19). Association for Computing Machinery, New York, NY, USA, 339–354.

[18] Dong Wang, Jiang Ming, Ting Chen, Xiaosong Zhang, and Chao Wang. 2018. Cracking IoT Device User Account via Brute-force Attack to SMS Authentication Code. In Proceedings of the First Workshop on Radical and Experiential Security (RESEC '18). Association for Computing Machinery, New York, NY, USA, 57–60.

[19] Zeyu Lei , Yuhong Nan, Yanick Fratantonio, and Antonio Bianchi, On the Insecurity of SMS One-Time Password Messages against Local Attackers in Modern Mobile Devices, Network and Distributed Systems Security (NDSS) Symposium 2021.