X-Platform Phishing: Abusing Trust for Targeted Attacks

Short Paper

Hossein Siadati, Toan Nguyen, Nasir Memon

{hossein, toan.v.nguyen, memon}@nyu.edu
New York University

Abstract. The goal of anti-phishing techniques is to reduce the delivery rate of phishing emails, and anti-phishing training aims to decrease the phishing click-through rates. This paper presents the *X-Platform Phishing Attack*, a deceptive phishing attack with an alarmingly high delivery and click-through rates, and highlights a subclass of phishing attacks that existing anti-phishing methods do not seem to be able to address. The main characteristic of this attack is that an attacker is able to embed a malicious link within a legitimate message generated by service providers (e.g., Github, Google, Amazon) and sends it using their infrastructure to his targets. This technique results in the bypassing of existing anti-phishing filters because it utilizes reputable service providers to generate seemingly legitimate emails. This also makes it highly likely for the targets of the attack to click on the phishing link as the email id of a legitimate provider is being used. An X-Platform Phishing attack can use email-based messaging and notification mechanisms such as friend requests, membership invitations, status updates, and customizable gift cards to embed and deliver phishing links to their targets. We have tested the delivery and click-through rates of this attack experimentally, based on a customized phishing email tunneled through GitHub's *pull-request* mechanism. We observed that 100% of X-Platform Phishing emails passed the anti-phishing systems and were delivered to the inbox of the target subjects. All of the participants clicked on phishing messages, and in some cases, forwarded the message to other project collaborators who also clicked on the phishing links.

Keywords: Targeted Attack, Phishing, Cross-Platform Attack

1 Introduction

Social engineering has become a core component of cyberattacks with financial and political incentives. Recent high profile attacks, such as the Target [20] and Sony [3] attacks used phishing emails to steal credentials of employees to infect their machines and establish a foothold inside a target network. *Business Email Compromise* (BEC) scams use phishing emails to deceive employees of a target company into transferring money to scammers' accounts [4]. *Political and celebrity hacks* such as recent attacks on the Democratic National Committee (DNC) [12] have used multistage phishing techniques to access

confidential information. These phishing attacks focus on social engineering of a certain person or population and therefore are referred to as *targeted attacks*.

Advances in techniques for phishing detection impede the *delivery* of phishing emails. Security awareness improves the users' vigilance and therefore decreases the *click-through* rate on phishing emails. For example, Sender Policy Framework (SPF) [10] and DomainKey Identified Mail (DKIM) [1] have made it harder for the attackers to spoof a sender's email, and *Blacklist* of IP addresses has made it harder to use botnets for sending phishing emails. In addition, content-based anti-phishing engines combined with other signals have been successful in stopping large volume of phishing emails. Moreover, companies have invested in phishing training campaigns, that improve the overall awareness and resilience of their users. As a result, it is harder to deceive enterprise users. Therefore, it is natural for attackers to invest in devising new ways for delivering and luring victims to respond to phishing emails. For security researchers, it is important to be ahead of the curve, predict potential attacks, and provide required fixes.

This paper describes an advanced form of targeted attack which we call *X-Platform Phishing* (XPP)[1] that can bypass existing phishing filtering techniques and is able to elicit a high amount of responses from victims. This attack exploits the email-based messaging and notification mechanism of reputable platforms and leverages the trust of the end-users to the services they use, to deliver customized phishing messages to a target victim and deceive her/him into clicking on the phishing links. Examples of customizable messages sent by platforms include Github notifications, Google Scholar alerts, LinkedIn friend requests, Dropbox notifications, and Amazon gift card notifications. These messages are sent from a fixed email address of a reputable platform or service (e.g., notification@[domain name of the service provider]) and therefore are trusted by email services. Moreover, users have subscribed for the service, trust the emails from the service provider, and frequently receive and therefore expect to receive such emails. Consequently, it is very likely for them to read the phishing email and visit the malicious link.

To demonstrate the possibility of XPP, we ran a pilot study on the Github platform. We used the pull request functionality of Github to send customized phishing messages to subjects of our experiment. The results show that 100% of the subjects clicked on the phishing links. More surprisingly, not only did these subjects click on the links, they also forwarded the email to their colleagues, who in turn, fell for the attack.

Existing anti-phishing mechanisms are not able to detect and block this type of phishing attack. The main reason is that the email filtering mechanisms do not differentiate between emails from an enterprise and a customized email containing user messages delivered by an enterprise email address. This is very similar to X-Site Scripting (XSS) attack where a user-generated input containing a malicious script is allowed to run in the context and origin of the service provider on a browser. A potential remedy to stop X-Platform Phishing includes sanitizing the contents of user messages before embedding them in the emails. Another possible approach is creating and exchanging a user trust

[1] It is pronounced Cross-Platform Phishing

score between service providers to facilitate the assessment of emails delivered by service providers.

The main contributions of this paper are the introduction of X-Platform Phishing and preliminary measurement of its delivery and click-through rates. We also discuss the shortcomings of anti-phishing mechanisms and propose remediation.

2 Background

X-Platform Phishing as described in this paper has the capability of bypassing existing email filtering mechanisms as well as driving high click-through rates when customized for specific targets. In this section, we discuss the methods of phishing email filtering, and characteristics of targeted phishing attacks in connection with XPP.

2.1 Anti-Phishing Techniques

Unwanted emails initially were used for advertising and later for spreading malware, phishing, and scamming people [14]. The traditional approach of filtering unwanted emails rely on blacklisting spamming IP addresses [9]. These lists are updated quickly with a median of 1.5 hours to include new spamming IPs in the blacklist. Spammers have responded to blacklisting using a "Snowshoe spam" strategy that spreads the workload of spamming IPs by sending very short bursts of spam from several IPs [15]. Blacklisting IPs is not effective against XPP since the emails are originated and sent from IP address of legitimate service providers such as Github, Amazon, and LinkedIn.

Another approach is content-based spam filtering, that is mostly effective when the content of spamming messages are distinguishable from normal conversational emails due to the usage of words and links [11]. In the XPP, content of the phishing email is a mix of content from legitimate service provider and a portion customized by the attacker. This combination of good and bad content makes the task of classification for text-based classification more challenging. Moreover, previous work has shown that content-based filtering can be easily circumvented [13].

Email source authentication is another anti-spam mechanism, which has reduced the possibility of spoofing dramatically. These mechanisms include Sender Policy Framework (SPF) [10], DomainKey Identified Mail (DKIM) [1], and Domain-based Message Authentication, Reporting, and Conformance (DMARC) [2]. In a XPP attack, phishing email is sent from a reputable service provider by all valid signatures and from a legitimate IP address. This makes it very easy for XPP emails to get delivered into the Inbox of the victims. Therefore, more advanced tools and techniques are required to detect and block delivery of XPP.

2.2 Targeted Phishing Attacks

A targeted attack is a form of phishing attacks that includes deceptive messages and links customized for a high value target (e.g., staff of a financial

company, a politician) in order to increase the yield of the attack response. This method has been used extensively as a starting point of many high profile attacks. In fact, Verizon's Data Breach Investigations Report has listed phishing as the favorite method used by attackers [17]. Existing targeted attacks usually spoof the email address of a well-known service providers to appear legitimate. For example, a phishing email sent to John Podesta [5] spoofed "googlemail.com" domain that belongs to Google. Spoofing is becoming harder due to the deployment of more strict email rejection policies by domains. One logical move of the attackers then, as envisioned in this paper, would be to piggy-back over the trust of messaging between legitimate service providers to deliver their phishing emails.

In a 2011 report [6], Cisco reported that 70% of users who receive targeted phishing emails open and read them. In comparison, only 3% for traditional mass phishing emails are read by users. This shows the comparably higher persuasiveness of targeted phishing emails. In that report, however, Cisco has considered the block rate of both types of phishing attack as 99%, meaning that the majority of the mass as well as targeted phishing emails are blocked by anti-phishing engines. The attack discussed in this paper proves this otherwise by experimenting a phishing attack that can not be blocked by existing anti-phishing engines and yields open rate and click-through rate of 100%. This calls for new anti-phishing approaches.

3 X-Platform Phishing

X-Platform Phishing, analogous to the X-Site Scripting (XSS) attack, exploits the email-based messaging and notification mechanism of a legitimate service to deliver phishing messages to target victims. For example, an attacker can send an electronic gift-card to victim with a customized message that includes a phishing link. Since the gift-card is sent by the email address of a reputable service provider (e.g., Amazon, Starbucks), the receiving email domain delivers it to the Inbox of the target.

Many service providers use email-based messaging and notification mechanisms for different purposes including *friend requests* (e.g., Linkedin), *membership invitations* (e.g. Telegram), *status updates* (e.g., Github pull request, Google Scholar notification), and Gift Card (e.g., Amazon, Starbucks). These communications are feasible even between users that do not trust each other. Further, the messages themselves are customizable by the attackers and the final message is embedded in a template prepared by the sending service provider.

Users usually respond to benign messages from service providers in a certain way. For example, users click on "Apply to your Amazon Account" button when they open an Amazon Gift Card. In the XPP, an attacker customizes the message in a way that deceives the targeted victim to click on elements they control. The content of the message sent to victims are highly customizable. The attacker can link an HTML tag or an image that loads in the email client of the target. For example, an attacker may create an "Apply to your Amazon Account" button inside the message section of the email and link it to a phishing website.

The phishing link bundled in a message sent by a reputable service provider will be delivered by an email address owned by service provider. This kind of email addresses is highly trusted and therefore the email will be delivered to the target's Inbox. Moreover, users find these messages as routine due to the trust built over years. The high delivery rate mixed with the trust of users, result in a very powerful attack. In the rest of this section, we detail use cases of this attack on users of two well-known platform with big user-bases and potentially high impact.

Use Case I. *Github* is a platform for collaborative software development. It has about 14 million users and more than 35 million repositories [18]. Security of this platform and its users are very important, specifically because software bundled and distributed based on Github projects are installed on millions of devices around the world. For example, half a million servers were identified to be vulnerable due to Heartbleed [19], a critical security bug discovered in OpenSSL, which has been developed and maintained on Github. Attackers are very interested in injecting vulnerabilities in software by compromising platforms such as Github [16]. Indeed, Free Software Foundation's repository was under the control of hackers for more than two months and potentially served backdoored versions of GNU software to millions of users [7]. Therefore, stealing credentials of Github developers can be disastrous.

An attacker who targets a developer is able to use XPP to launch a phishing attack by abusing the pull request on Github. This is because the messages of such requests are customizable by the person who creates them. An attacker who uses XPP technique customizes the message in the pull request and adds malicious content. Upon issuing the pull request, an email containing the malicious message will be delivered via *notifications@github.com* email address to the Inbox of target. It should be noted that any Github user can create a pull request for any public project and *mention* a specific user as the receiver of the request. For example, an attacker pulls a request on a project of his target on Github and provides following message. "I've found a critical security vulnerability in your project. Detail and a proof-of-concept are provided *[link]*," in which *link* can lead the target to a phishing website that requires Github login credential to access the proof-of-concept code or to a drive-by-download malware. The attacker then mentions the target using @ + target's Github username. After this pull request is submitted by the attacker, Github will send an email which contains the message and the link to the target. As shown in our experiment later, this attack yields very high delivery and click-through rates.

Use Case II *Google Scholar* is one of the services provided by Google, specialized in indexing research publications and scholarly books. The number of publications indexed by this service is estimated to be 160 million documents as of May 2014. Millions of researchers from academia and industry use this service to access scholarly materials. Google indexes material from reputable publishers as well as open publication websites. Users of Google Scholar can subscribe to alerts to receive a notification when a new publication of their interest gets published. Notification emails of this type typically contain links to published documents or sources of documents. An attacker who targets researchers of special interest can create a Google Scholar account and create a

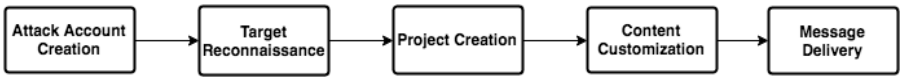

Fig. 1. Steps of X-Platform Phishing in our experiment

fake publication in the topic of his target's interest and uploads to a website in his control. Once he adds this publication to his profile, Google Scholar will index and notify the interested users via emails. When the targets click on the link in the emails, they are directed to the attacker's website where the attacker may present them with a fake Google Scholar login page which asks the targets to login to view the document or plant a malicious document (a rouge PDF or DOC file) for the targets to download. This attack scenario, without doubt, leads to innocent clicks and resultant compromise.

4 Experiment

To demonstrate the feasibility of launching an XPP attack and measure its delivery and click-through rate, we ran a pilot study on a small population of users of the Github platform.

4.1 Attack Setup

The instance of X-Platform Phishing used in our experiment has several steps as depicted in Figure 1. We describe each step in this subsection.
Attack Account Creation. We created a Github account and set up the profile of this account to appear as a developer from our institution.
Target Reconnaissance. XPP is a targeted attack meaning that the attacker uses contextual information about the target to improve yield. For each subject in this experiment, we found a Github project that she/he had been working on recently. We adjusted the phishing messages in the context of this project to make it more likely that the target would respond.
Project Creation. We cloned the active project of each subject, selected in the step above, and sent phishing messages to the subject by pulling a request from this project.
Content Customization. The content of a message sent to a Github developer is customizable. We used Markdown[8] to customize the text, add an image for tracking user, and put a click-able hyperlink to the message. One aspect of customization of the phishing message was to push Github's default message down to an invisible area so the subjects do not get distracted. For this, we added a number of newline tags
 to the end of the phishing message.
Message Delivery. There are several ways to send an email message using Github. In this experiment, we used Github's Pull Request (PR) to send phishing messages. A pull request allows a member of a project to receive notification about changes that other contributors make and want to merge them into the project source. We created a pull request on projects that we cloned from the subjects in our experiment, customized the message in the

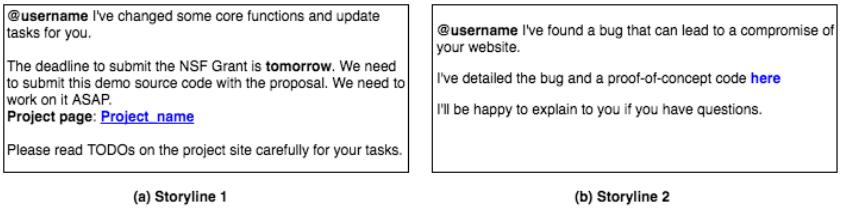

Fig. 2. The content of the messages in the story lines used in the experiment

pull request using Markdown, and mentioned subject's Github handler[2] in the pull request. Github automatically sent the customized phishing message to their email addresses. Other possible methods of sending messages include adding target users as collaborator of a project, creating an issue report, or adding comment and mentioning their Github handler.

4.2 Phishing Message Design

We experimented the effect of two different story lines in the context of X-Platform Phishing. The first story line was *an approaching NSF grant proposal*. In this variation, we mentioned victims, who were graduate students, in a Github project with a message asking them to follow a todo-list regarding a grant proposal. A snapshot of this attack message is shown in Figure 2(a). The second story line was *a bug report* in which we notified subjects to fix a bug in their project. A snapshot of this attack is shown in Figure 2(b).

4.3 Subject Recruitment

We selected 20 subjects from a convenient pool of students from our institution. Candidate subjects were selected because they had active Github accounts working on some projects over the past few months. All these students were graduate students, five of which were doctoral and the rest of them were master students. NSF grant message was used for doctoral students, and bug report message was used for master students. Subjects were not notified prior to the experiment but were debriefed and interviewed afterward.

4.4 Collected Data

We collected data about two different aspects of the users response to XPP namely *message delivery* and *phishing click*. To know if a message was delivered, we added an invisible 1x1 image in the phishing message to notify our server whenever the message was loaded in a browser or application. A separate project was created for each subject so a customized link could easily identify the click-responses of subjects.

[2] A user identifier of users inside Github starting with @

4.5 Result

The first scenario (i.e., NSF Grant) yielded 100% message delivery as well as 100% phishing click-through rates. For this case, we added a link to a website in our control on the description of the Github project page. When a subject clicks on this link, a request is sent to our website where we log the event (timestamp, subject ID, etc.) In a real attack, this link might lead to a fake Github login page or to a drive-by-download malware which may cause harms to the subjects or victims. We observed that all of the subjects clicked on this link. In one case, the email was circulated among other members of the projects and we observed multiple clicks on the link embedded in that project's description. This was confirmed later by interviewing the subject.

In the second scenario (i.e., bug fix), we also observed that all subjects opened the phishing emails and clicked on the phishing links embedded in the emails. In both scenarios, the victims clicked on the phishing links and visited our Github profile within an hour since the pull requests were submitted.

5 Discussions, Limitations, and Conclusion

Github is only one example of the platforms that can be misused for launching XPP attacks. Many other collaborative software development platforms such as Bitbucket, SourceForge, and Gitlab with millions of users, as well as any other platforms including LinkedIn, Amazon, Telegram, Google Scholar, ResearchGate, Academia.edu that have methods for email-based message exchange using a fixed identity of service provider are susceptible to be used for X-Platform Phishing attack.

It is challenging for an email platform to verify the legitimacy of user generated messages sent across platforms using service provider's email address. First of all, the messages are sent from a trustworthy domain. In addition, email services do not have fine grained information about the sender and the context of which users communicate on other platforms. Lack of information makes the usage of user-based trust score for filtering such massages challenging.

Possible countermeasures to XPP attacks include spam filtering mechanisms that consider trust relations of entities mentioned in delivered messages (e.g., handlers in form of @ in the Github pull requests) as a feature in email filtering. Another approach is extension of email delivery protocols to exchange user trust scores between sender and receiver domains. Such scores can be incorporated in the phishing classifiers. Lastly, having a cyberspace resilient to social engineering is the duty of all parties. All platforms have to employ anti-phishing mechanisms for both outgoing and incoming messages.

The pilot experiment described in this paper and the reported results are based on a limited pool of subjects and scenarios. Therefore, more extensive experiments are required to provide deeper insight about the effect of X-Platform Phishing, as we plan to explore in future. Designing countermeasures also is a high priority.

In conclusion, we have identified the possibility of a targeted phishing attack with potentially high impact in real-world. Success of this attack can

be attributed to the trust between platforms to deliver messages and the trust of users on messages coming from reputable service providers. Leveraging this trust, an attacker can achieve high delivery and click-through rates. This calls for improved methods for detection of targeted phishing emails.

References

1. RFC 6376: DomainKeys Identified Mail (DKIM) Signatures. https://tools.ietf.org/html/rfc6376 (Accessed 20-Dec-2016).
2. Domain-based message authentication, reporting, and conformance (dmarc). https://tools.ietf.org/html/rfc7489, 2015. Accessed: 2016-04-17.
3. BISSON, D. Sony Hackers Used Phishing Emails to Breach Company Networks. https://www.tripwire.com/state-of-security/latest-security-news/sony-hackers-used-phishing-emails-to-breach-company-networks/ (Accessed 20-Dec-2016).
4. Brian Krebs. FBI: $1.2B Lost to Business Email Scams. https://krebsonsecurity.com/2015/08/fbi-1-2b-lost-to-business-email-scams/ (Accessed 20-Dec-2016).
5. CBS. The phishing email that hacked the account of John Podesta. http://www.cbsnews.com/news/the-phishing-email-that-hacked-the-account-of-john-podesta/ (Accessed 20-Dec-2016).
6. CISCO. Email Attacks: This Time Its Personal. http://www.cisco.com/c/dam/en/us/products/collateral/security/email-security-appliance/targeted_attacks.pdf (Accessed 20-Dec-2016).
7. Geek.com. Major Open Source code repository hacked for months, says FSF. https://www.geek.com/news/major-open-source-code-repository-hacked-for-months-says-fsf-551344/ (Accessed 20-Dec-2016).
8. Github. Mastering Markdown. https://guides.github.com/features/mastering-markdown/ (Accessed 20-Dec-2016).
9. Jung, J., and Sit, E. An empirical study of spam traffic and the use of dns black lists. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement* (2004), ACM, pp. 370–375.
10. Kitterman, S. Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1. https://tools.ietf.org/html/rfc7208 (Accessed 20-Dec-2016).
11. Metsis, V., Androutsopoulos, I., and Paliouras, G. Spam filtering with naive bayes-which naive bayes? In *CEAS* (2006), pp. 27–28.
12. Motherboard. The hack we can't see: All Signs Point to Russia Being Behind the DNC Hack. https://motherboard.vice.com/read/all-signs-point-to-russia-being-behind-the-dnc-hack (Accessed 20-Dec-2016).
13. Palka, S., and McCoy, D. Fuzzing e-mail filters with generative grammars and n-gram analysis. In *9th USENIX Workshop on Offensive Technologies (WOOT 15)* (2015).
14. Siadati, H., Jafrikhah, S., and Jackobsson, M. *Traditional Countermeasures to Unwanted Email*. Springer-Verlag New York.
15. Symantec. Internet security threat report (istr) 2016. https://www.symantec.com/content/dam/symantec/docs/reports/istr-21-2016-en.pdf (Accessed 20-Dec-2016).

16. Torres-Arias, S., Ammula, A. K., Curtmola, R., and Cappos, J. On omitting commits and committing omissions: Preventing git metadata tampering that (re) introduces software vulnerabilities. In *25th USENIX Security Symposium, USENIX Security* (2016), vol. 16, pp. 10–12.
17. Verizon. 2016 Data Breach Investigations Report. `http://www.verizonenterprise.com/resources/reports/rp_dbir-2016-executive-summary_xg_en.pd` (Accessed 20-Dec-2016).
18. Wikipedia. Github. `https://en.wikipedia.org/wiki/Github` (Accessed 20-Dec-2016).
19. Wikipedia. Half a million widely trusted websites vulnerable to Heartbleed bug. `https://news.netcraft.com/archives/2014/04/08/half-a-million-widely-trusted-websites-vulnerable-to-heartbleed-bug.html` (Accessed 20-Dec-2016).
20. ZDNet. Anatomy of the Target data breach. `http://www.zdnet.com/article/anatomy-of-the-target-data-breach-missed-opportunities-and-lessons-learned/` (Accessed 20-Dec-2016).