Phishing Attack Recognition by End-Users: Concepts for URL Visualization and Implementation

V. Erbenich, D. Träder, A. Heinemann and M. Nural

User-Centered Security Research Group, Department of Computer Science
Hochschule Darmstadt – University of Applied Sciences, Germany
e-mail: vivian@erbenich.eu · {daniel.traeder | andreas.heinemann | meltem.nural}@h-da.de

Abstract

Social engineering, through means of phishing, is a very popular entry point for a targeted attack in order to obtain further data on a company or private individual, e.g. by injecting malware on the victim's machine. A phishing attack that leads to a malicious website can usually be identified by the HTTP link with expert knowledge. However, only very few users pay attention to the link or have the necessary knowledge to recognize a threat as such. This work addresses the question of how current link visualization could be improved so that a user can better identify whether the link points to a phishing site or a legitimate site. Additionally, we also address the question of how our proposed link concepts can be put into practice. As an improvement, the outer shape of a link will be adapted by content-based formatting, trimming and other features. The user will thus be able to interactively explore a URL and its components in order to gain a better decision. As a next step, we plan to evaluate our concepts in a controlled lab environment with few test persons as well as by a large-scale online user-study.

Keywords

Phishing, URL Pruning, URL Visualization, Security Warning, Usable Security

1 Motivation

Nowadays, phishing attacks are one of the most important cybersecurity threats Internet users are exposed to (APWG et al. 2019). Phishing attacks are still popular due to a lack of knowledge about phishing and technical context of the Internet in general, lack of attention and awareness, inefficient anti-phishing tools and the – often falsely – feeling of security and control while surfing the web (Erkkila 2011; Dhamija et al. 2006; Alsharnouby et al. 2015). Further, Erkkila suggests, that users need to be trained about the danger of phishing attacks as well as tools that are easier to use and that support the user to make a save decision. In this sense, the focus of this work is to help users identify a phishing URL, when all other detection methods fail.

Phishing attacks are often recognizable by the domain of a URL in a message or the sender of an e-mail. However, this assumes, that the user carefully checks a URL and the domain in it with sufficient background knowledge and considers whether the message is real or fake. Therefore, the aim of this work is to accompany the user in the process of URL analysis through optimized visualizations of URLs and to make inconsistencies in URLs easily recognizable. This should be done directly in the place

where the user comes into contact with the URL at first place. The user should be made aware of possible dangers through eye-catching and informative visualizations. Our work is driven by two questions: a) How does a URL needs to be visualized so that a user can better identify whether the target of that URL is a phishing site or a legitimate site? and b) How must a URL be embedded in an application so that a user can better identify whether the target of that URL is a phishing site or a legitimate site? In order to give answers to these questions, we combine existing URL visualization proposals from literature ((Lin et al. 2011; Volkamer et al. 2016; Volkamer et al. 2017), extend these and combine them with our new ones.

The remainder of this work is structured as follows: Our starting point is URL pruning and URL highlighting, security warnings and interventions as introduced in Section 2. Next, we present our extensions and new concepts in Section 3. From an implementation point of view, we discuss design variants and integration issues in Section 4 and sum-up our work including a short outlook in Section 5.

2 Related Work

For the sake of completeness: First and current rolled out approaches to fight phishing attacks are technical means to filter out or at least mark suspicious messages for a user. Gupta et al. (2017) gives a short overview of current anti phishing methods and phishing taxonomies, example approaches include blacklists, whitelists, content-based heuristics, machine-learning approaches (Ozgur et al. 2019; Patil et al. 2018) and the verification of security features (Garera et al. 2007; Fette et al. 2007; Chhabra et al. 2011). The problem is, however, that if such tools and algorithms malfunction, the end-user is left on his own. Therefore, our work is based on and extends URL pruning and highlighting as well as security warnings and interventions as an aid for the end-user. We will discuss the most relevant corresponding work hereafter.

2.1 URL Pruning and URL Highlighting

URL highlighting was first proposed by Lin et al. (2011). The idea is to highlight the domain name of the URL in the URL address bar of a browser so that users can determine the legitimacy of a website. Lin et al. (ibid.) showed the benefit of URL highlighting but emphasized that this cannot be used as the only method to prevent phishing attacks.

Volkamer et al. (2016) proposed an improved approach to URL visualization through a two-stage measure. The basic idea is to draw the user's attention to the address line and to improve the understanding of the URL by cutting off irrelevant parts of the URL. Their result of an accompanying user-study showed an improvement in phishing detection by 96,7 percent. Further on, Volkamer et al. (2017) developed a browser and e-mail client plug-in to display the actual target of an URL via tooltip by hovering over the displayed URL. Making use of visual highlighting of an URL, alarm colors and a forced delay to open an URL, users gain more hints and time to check the URL and were able to detect more malicious URLs as compared to users without the tool at hand.

2.2 Security Warnings and Interventions

In order to accompany an end-user during checking an URL and support him to make a sound judgment on whether an URL is dangerous or not, acknowledged design criteria on security and safety warnings need to be taken into account. We name the most prominent: Warnings should be active in the sense that they interrupt a user's workflow and draw attention (Egelman 2009), warnings should be easily distinguishable from other dialogs and pop-up windows (Krol et al. 2012), warnings should be simple to understand (Yang et al. 2017) and prefer images over text (Christin et al. 2013), warnings should have a clear design and be simple to understand (Wilson et al. 2017), warnings should come with a minimal security effort and standard selection of dialog questions should be based on safe answers (Yang et al. 2017; Egelman 2009), warning frequency should be chosen with care in order to avoid a decrease in neuronal activity of the brain and therefore hindering the perception of a warning (Reeder et al. 2018; Felt et al. 2015; Sunshine et al. 2009).

In addition, security intervention plays an important role. The core idea is to temporary interfere with or take over a user's workflow in order to draw their attention to a present security issue (Egelman 2009; Volkamer et al. 2017). We will present our extensions and new concepts in the following section.

3 Extended and new URL Visualization and Interaction

In order to make the appearance of an URL easier to understand and support end-users without expert knowledge in identifying malicious URLs, we extend existing approaches and combine them with new ones. Table 1 gives an overview of our contribution. Altogether, our methods fall into one of five categories. We will provide details about each method in the course of the next sections.

Category	Method	Used	Extended	New
URL Pruning	Prune protocol	• *a*		
	Prune subdomains			•
	Prune arguments			•
	Show full URL on click			•
URL Highlighting	Highlight domain	• *b*		
	Wider letter-spacing on domain	• *b*		
	Soften subdomains			•
	Soften top-level-domain			•
URL Coloring and URL Explanation	Color numbers in domain			•
	Color typosquats in domain			•
	Color special character in domain			•
	Explaining of URL features			•
Security Warnings	Warning if no HTTPS	• *c*	•	
	Warning on short-URLs	• *b*	•	
	Warning on long URLs			•
	Warning on long subdomains			•
Security Interventions	Security-awareness delay	• *db*	•	
	Scoring based alarm levels	• *d*	•	

Table 2: Overview of used, extended and new URL Visualizations

[a] (Volkamer et al. 2016) [b] (Volkamer et al. 2017) [c] (Garfinkel 2005). [d] (Egelman 2009)

3.1 URL Pruning

As introduced by Volkamer et al. (2016), the truncation of some URL components serves to focus on individual areas of a URL. The domain, as well as the top-level domain, are the only important components that a user needs to recognize first in order to be able to check whether the link leads to the desired provider and its domain. The other components of the URL (protocol, subdomains, and parameters) do not contribute to the unambiguous recognition of the target domain. Therefore, we propose to truncate these parameters.

Figure 1: Full, truncated and highlighted URL

On the other hand, this approach might unsettle users. The users are no longer clear where the link really leads or might not recognize the link. To solve this problem, our URL pruning has been extended as follows:

When the URL is visualized, only the domain and the top-level domain will be shown from the beginning. If the user moves the cursor over the URL-visualization, the mouse changes to a hand symbol and makes clear to the user that he can click on the URL. If the user clicks on the URL visualization, the complete link becomes visible through an animation (Fig. 7b). Thus, the user can view the full link on request, but only sees the part needed for classification at the beginning. The animation should also help the user to understand the connection between the full URL and the abbreviated version (Fig. 1).

3.2 URL Highlighting

URL highlighting is another variant that can contribute to better recognition of a URL. As explained above, the URL can be displayed completely after clicking on it. In order to lead the focus in this view on the domain and the top-level domain as well, the different components of the URL are formatted in a differentiated way.

To ensure that the main focus remains on the domain, it is displayed in a higher font weight (three-quarter bold). The protocol, the subdomains as well as the parameters are reduced in lower font size (light). In addition, the contrast to the background is

reduced by half by reducing the opacity. The top-level domain remains in normal font size and full opacity but still stands out from the protocol and arguments.

Next, in order to make the individual letters of the domain appear on their own, the font size of the domain is increased by 0.1em. The font spacing (also called letter-spacing) determines the distance between individual characters. This should enable the user to better recognize dangerous character combinations in URLs such as m ≠ rn (Fig. 2).

3.3 URL Coloring and URL Explanation

In order to be able to enhance the recognition of likely phishing attempts, the domain in an URL is checked for special characters, numbers and suspicious letter combinations. These are highlighted in color and underlined (Fig. 2 last row).

amazon.com – *original domain*
arnazon.com – *fake domain with letter combination (m = rn)*
arnazon.com – *fake domain with letter combination + letter-spacing*
arnazon.com – *fake domain with letter combination + letter-spacing + highlighting*
arnazon.com – *fake domain with letter combination + letter-spacing + highlighting + coloring*

Figure 2: Letter combinations and letter spacing

If the user moves the cursor over these highlighted characteristics, they will be surrounded by a frame. A tooltip will appear above the corresponding characteristic. It contains a short text that explains why this characteristic was highlighted (Fig. 3a).

Another URL feature that may indicate a phishing attempt is the use of an IP address within an URL. A tooltip points out this potential danger (Fig. 3b).

Figure 3a: Mouseover tooltip with details

Figure 3b: IP address warning

3.4 Security Warnings

Whenever the length of an URL deviates too much from a typical length or the destination does not support HTTPS, we show a warning to the end-user. We classify a URL with the length of more than 300 characters as suspicious and pay special

attention to subdomains since it is a popular method to fake a domain by including the top-level domain in a subdomain. Therefore, if the subdomain has a length of more than 40 characters, the URL is classified as suspicious.

In addition, we provide a warning, if short URLs (as often provided by URL shortener services) are detected. Our current implementation works with a blacklist. In the future, we plan to integrate services like getlinkinfo.com.

3.5 Security Interventions

Besides URL-Visualization, we want to help the user to recognize a risk by intervening into the user's workflow. In order to force the user's attention to the link for a few seconds, an additional delay is used (Egelman et al. 2008). The delay should force the user to interrupt his work and pay attention to the URL. The delay is set to 5 seconds in our current implementation and only appears for new links, an end-user has not visited before.

Figure 4: Security-awareness delay

We propose to use a loading bar close to the URL itself to inform the user intuitively about the delay. By this design, we aim to steer a user's focus on the URL structure in order to detect suspicious ones (Fig. 4). At this point it must be noted that this only reflects our assumption. In an evaluation these suggestions would have to be examined for effectiveness.

We further propose three alarm levels based on a simple internal score to improve our security invention further. At level 1, we did not find any well-known suspicious phishing features. Since phishing might still be possible, we ask the user to check a link. At level 2 we have identified at least one phishing feature and at level 3 two or more phishing features are present (Fig. 5). The alarm level severity is further indicated using grey, yellow and red signal colors.

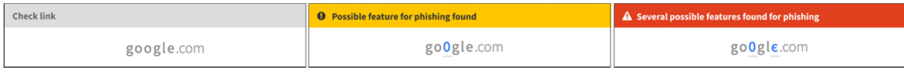


Figure 5: Scoring based alarm levels

4 Prototypical Implementation and Integration

State-of-the-art phishing attacks heavily rely on HTML-messages since these technologies provide easy mechanisms to hide a real URL target from a displayed one. Therefore, our first prototypical implementation realizes a tooltip, which is based on HTML5/JavaScript support in the client (Fig. 6). In general, it should be ensured that the end-user does not become accustomed to warnings. We want to achieve this, by

not displaying URLs that have already been marked as trustworthy by the end-user (Reeder et al. 2018; Felt et al. 2015; Sunshine et al. 2009).

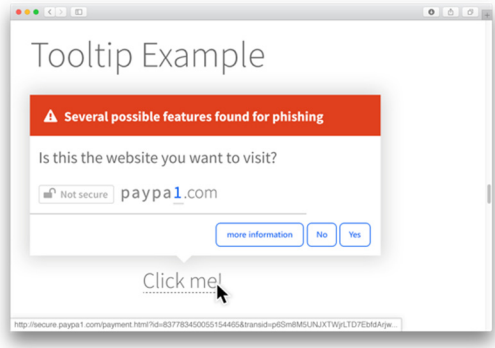
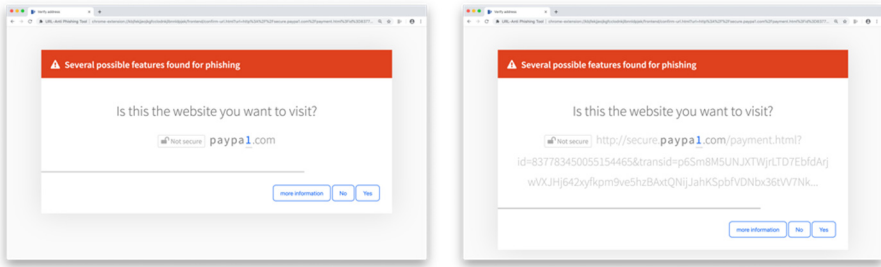


Figure 6: HTML5/JavaScript based prototype

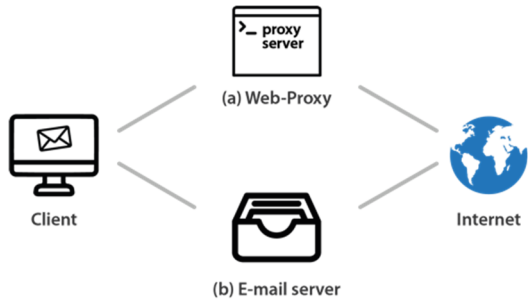
However, this approach assumes, that our client allows the modification of the message payload, i.e. the injection of additional HTML5/JavaScript code in the message itself. Since this is not always the case, e.g., the latest Microsoft Outlook E-Mail client does not offer such kind of API/hock, we have also implemented a browser extension that intercepts all URL opening requests from various clients installed on the end-user PC. As a prerequisite, the extension must be implemented for the default browser (Fig. 7). One could also think of an application on its own that registers as the default browser at the operating system. This would allow an implementation not to be bound to a browser's plugin capabilities.



(a)…showing a pruned URL. **(b)...showing the full URL**

Figure 7: Browser extension prototype

Finally, we want to mention, that both of our current prototypes run on the client side. Thinking of software distribution and update a more suitable integration spot might be located within a company's infrastructure. Fig. 8 illustrates two other possible spots: A Web-Proxy (a) and an E-mail server (b).



Icons taken from: The Noun Project Icons - Creative Commons CC-BY Arthur Shlain, Nauraicon, Rockicon, Gha Arizal, Arafat Uddin

Figure 8: Possible integration spots

The payload modification of an HTML5/JavaScript-based message could happen at an E-mail Server beforehand (as long as the message is not encrypted). The browser extension could be replaced by a Web-Proxy, that analyses the URL according to phishing features and returns our proposed visualization and warnings as an HTML page in a first step. Only if the user confirms to visit the URL, the URL is passed through the Web-Proxy and the content is fetched from the target address.

5 Conclusion and Outlook

Since the effectiveness of automatic phishing detection is limited and end-users are exposed to phishing messages containing malicious URLs, we propose ways to make phishing attack recognition easier for them. Therefore, we used and extended known URL visualizations methods and developed new ways for the visualization of URLs. In that sense, we break down URLs to their minimal form so that a user will see only the most critical parts of an URL for successful phishing detection. Nonetheless, the minimal form could be expanded to see the full URL on request. We pay special attention to common phishing attack features in URLs such as misleading character combinations or unusual length of domains or subdomains. All in all, this should help the end-user judge, if the URL is legitimate or not and therefore save to visit. In addition, we add security warnings and interventions.

We have realized our concepts as an HTML/JavaScript based tooltip and a browser extension and shortly discussed further integration possibilities in the backend of a company's IT infrastructure.

As a next step, we plan to evaluate our concepts in a controlled lab environment with few test persons as well as by a large-scale online user-study. Further on, we like to investigate whether and to what extent our concepts can be transferred to non-desktop devices (tablets, smartphones) with different interaction metaphors and resource limitations, especially with regard to the display.

Acknowledgement: This work was funded by the Hessian Ministry of the Interior and Sports (HMdIS), Germany, within the "Round Table Cybersecurity@Hessen".

6 References

Alsharnouby, Mohamed, Furkan Alaca, and Sonia Chiasson (2015). "Why phishing still works: User strategies for combating phishing attacks". *Int. J. of Human-Computer Studies* 82. Elsevier

APWG (2019). Phishing Activity Trends Report – 4th Quarter 2018. url: https://docs.apwg.org/reports/apwg_trends_report_q4_2018.pdf (Accessed 20 May 2019)

Chhabra, Sidharth and Aggarwal, Anupama and Benevenuto, Fabricio and Kumaraguru, Ponnurangam (2011). "Phi. sh/$ ocial: the phishing landscape through short urls". In: 8th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conf. CEAS '11. ACM

Christin, Delphine, Martin Michalak, and Matthias Hollick (2013). "Raising User Awareness About Privacy Threats in Participatory Sensing Applications Through Graphical Warnings". *Int. Conference on Advances in Mobile Computing & Multimedia*. MoMM '13. ACM

Dhamija, Rachna, J. D. Tygar, and Marti Hearst (2006). "Why Phishing Works". *SIGCHI Conference on Human Factors in Computing Systems*. CHI '06. ACM

Egelman, Serge (2009) Trust Me: Design Patterns for Constructing Trustworthy Trust Indicators. Ph.D. Dissertation. *Carnegie Mellon Univ.*, Pittsburgh, PA, USA.

Egelman, Serge, Lorrie Faith Cranor, and Jason Hong (2008). "You've Been Warned: An Empirical Study of the Effectiveness of Web Browser Phishing Warnings". *SIGCHI Conference on Human Factors in Computing Systems*. CHI '08. ACM

Erkkila, Jussi-Pekka (2011). "Why We Fall for Phishing". *SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. ACM, pp. 7–12.

Felt, Adrienne Porter and Ainslie, Alex and Reeder, Robert W. and Consolvo, Sunny and Thyagaraja, Somas and Bettes, Alan and Harris, Helen and Grimes, Jeff (2015). "Improving SSL Warnings: Comprehension and Adherence". *ACM Conference on Human Factors in Computing Systems*. CHI '15. ACM

Fette, Ian, Norman Sadeh, and Anthony Tomasic (2007). "Learning to Detect Phishing Emails". *16th International Conference on World Wide Web*. WWW '07. ACM.

Garera, Sujata and Provos, Niels and Chew, Monica and Rubin, Aviel D. (2007). "A framework for detection and measurement of phishing attacks". *2007 ACM Workshop on Recurring Malcode*. WORM '07. ACM.

Garfinkel, Simson (2005). "Design principles and patterns for computer systems that are simultaneously secure and usable". PhD thesis. *Massachusetts Institute of Technology*.

Gupta, B. B., Nalin A. G. Arachchilage, and Kostas E. Psannis (2018). "Defending against Phishing Attacks: Taxonomy of Methods, Current Issues and Future Directions". *Telecommunication Systems 67.2*,

Krol, Kat, Matthew Moroz, and M Angela Sasse (2012). "Don't work. Can't work? Why it's time to rethink security warnings". *Intl. Con. on Risks and Security of Internet and Systems* (CRiSIS). IEEE.

Lin, Eric and Greenberg, Saul and Trotter, Eileah and Ma, David and Aycock, John (2011). "Does Domain Highlighting Help People Identify Phishing Sites?" *SIGCHI Conference on Human Factors in Computing Systems*. CHI '11. ACM

Ozgur Koray Sahingoz, Ebubekir Buber, Onder Demir, Banu Diri (2019). "Machine learning based phishing detection from URLs" Expert Systems with Applications, Volume 117, Elsevier Ltd.

Patil V., Thakkar P., Shah C., Bhat T. and Godse S. P. (2018). "Detection and Prevention of Phishing Websites Using Machine Learning Approach. 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018

Reeder, Robert W and Felt, Adrienne Porter and Consolvo, Sunny and Malkin, Nathan and Thompson, Christopher and Egelman, Serge (2018). "An Experience Sampling Study of User Reactions to Browser Warnings in the Field". *2018 CHI Conference on Human Factors in Computing Systems*. CHI '18. ACM, pp. 512–524.

Sunshine, Joshua and Egelman, Serge and Almuhimedi, Hazim and Atri, Neha and Cranor, Lorrie Faith (2009). "Crying Wolf: An Empirical Study of SSL Warning Effectiveness." *USENIX security symposium*

Volkamer, Melanie, Karen Renaud, and Paul Gerber (2016). "Spot the Phish by Checking the Pruned URL". *Information and Computer Security 24.4*

Melanie Volkamer, Karen Renaud, Benjamin Reinheimer and Alexandra Kunz (2017). "User Experiences of TORPEDO: TOoltip-poweRed Phishing Email DetectiOn". *Computers & Security* 71

Wilson, Graham, Harry Maxwell, and Mike Just (2017). "Everything's Cool: Extending Security Warnings with Thermal Feedback*". 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*. CHI EA '17. ACM

Yang, Weining and Xiong, Aiping and Chen, Jing and Proctor, Robert W and Li, Ninghui (2017). "Use of phishing training to improve security warning compliance: evidence from a field experiment". *Hot Topics in Science of Security: Symposium and Bootcamp*. HoTSoS. ACM, pp. 52–61.