International Journal of Engineering & Technology, 7 (3.4) (2018) 42-46

SPC

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Phishfort – Anti-Phishing Framework

Eric Abraham Kalloor[1]*, Dr. Manoj Kumar Mishra[2], Prof. Joy Paulose[3]

[1,3] Department of Computer Science, Christ University,
Bengaluru, India – 560029
[2] Department of Computer Science, Banaras Hindu University,
Varanasi, India – 221005
**Corresponding author E-mail: eric.abraham@mca.christuniversity.in*

Abstract

Phishing attack is one of the most common form of attack used to get unauthorized access to users' credentials or any other sensitive information. It is classified under social engineering attack, which means it is not a technical vulnerability. The attacker exploits the human nature to make mistake by fooling the user to think that a given web page is genuine and submitting confidential data into an embedded form, which is harvested by the attacker. A phishing page is often an exact replica of the legitimate page, the only noticeable difference is the URL. Normal users do not pay close attention to the URL every time, hence they are exploited by the attacker. This paper suggests a login framework which can be used independently or along with a browser extension which will act as a line of defense against such phishing attacks. The semi-automated login mechanism suggested in this paper eliminates the need for the user to be alert at all time, and it also provides a personalized login screen so that the user can to distinguish between a genuine and fake login page quite easily.

Keywords: *Anti-Phishing; Cyber Security; Identity Theft; Phishing; Social Engineering*

1. Introduction

Phishing is a type of social engineering attack which exploits the human nature of blindly believing in what they see, the attacker presents a fake web page to the user which is exactly the same as the genuine one. Creating a phishing page is very easy and setting up an attack takes very less effort. The attacker visits the genuine page and clones the entire source code of the webpage and hosts it on a malicious server. The link to the malicious server hosting the phishing page is then sent to the user via various means such as email, SMS, social networking, etc. A carefully crafted malicious link can be masked, i.e. the actual link will be different from what is visible. When clicked the user will be redirected to the malicious server hosting the phishing page which will render a page on user's browser exactly as same as the genuine page. The only observable difference is in the URL which is rarely noticed by an average user, especially the users who do not have computer science background and do not realize the importance of address bar. If the users enter their sensitive data in such phishing pages, the data is stored into the malicious server or sent directly to the attacker in plaintext, and the user is redirected to either a genuine server or some other location as specified by the attacker. Most of the victims even at this point have no clue that they have been phished. After a successful phishing attack the attacker has access to the login credentials and other sensitive data which can be used to login to the website as the genuine user. Once the attacker has gained full access to the account as a normal user, the access is maintained by both attacker and victim until the password to the website is changed. If the victim changes the password first then the attacker no longer has access to it, and if the attacker manages to change the password first then the victim is locked out from his/her account and will not be able to access it. If the attacker changes the password recovery option then the users might never be able to reset the password and get back the access to their account ever again.

A successful phishing attack is potentially capable of causing serious damage depending on the attacker's intention. Any other linked account can also be compromised by the attacker easily. For example, if the attacker gets access to a victim's email account, then any other linked account can be compromised just by resetting the password using the password reset link which is sent to email. Further, the plaintext password of the victim will help the attacker to study the victim. The attacker can then use this type of information to perform more attacks; about what kind of password is used by them and either try the same login credentials or generate a wordlist with similar type of passwords and try to brute force other accounts of the victim. The attacker has the capability to defame the user socially, cause financial damage to the victim or cause some serious damage to the user's organization. Although there are various solutions such as firewalls, antivirus software, browser add-ons, they all need to be set up in user's environment to detect phishing attacks.

2. Related Work

Singh, Akhilendra Pratap, et al. [1] proposed a method of dynamic watermarking into a login page in order to prevent phishing attack. The work suggests including an image in the login page. The position of the image is dynamic and must be changed by the user while logging out. The changes made by the user are visible during the next login.

Juan Chen and Chuanxiong Guo [2] proposed a new method to detect phishing pages, called LinkGuard. LinkGuard contains of a program which analyses received emails or messages and looks for links in it, if any of the link is masked, i.e. the anchor text and the hyperlink URL are different then the program alerts the user.

"A Review on Phishing Attacks and Various Anti Phishing Techniques" by V. Suganya [3] presented a comprehensive study on how phishing attack can be carried out in various ways. It gives a wide range of possibilities to launch a simple yet fatal phishing attack.

Sharifi, Mohsen, et al. [4] have proposed a method against real-time phishing attacks based on mutual authentication technique. Their proposed method is effective against both traditional as well as real time phishing attacks. As per this technique the user is validated by the server via a password and similarly the user also validates the server.

A machine learning based method for detection of phishing site is proposed by Daisuke Miyamoto, et al. [5]. They have quantified several parameters such as age of domain, suspicious URL, known images, dots in URL, suspicious links, IP address and HTML forms, etc. Based on these parameters a machine learning algorithm trained on a dataset of over 3,000 URLs, classifies the given URL as phishing or safe.

Michael Atighetchi and Partha Pal [6] have developed a proxy called PhishBouncer which records the webpage and analyses from its attributes if it is a phishing site or not. It is an automated framework which crawls to different URLs and checks its authenticity based on the algorithm.

Mutual authentication techniques is proposed by Bryan Parno, et al. [7]. According to their method, the login process is initiated by the website only if the request originated from a trusted device. Trusted devices are identified by a key pair stored in the device. This setup is useful when a user has a personal device and only the user has access to it. In order to exploit this method the attacker will first have to compromise the trusted device or the key pair value.

A brief survey on different techniques is given by Minal Chawla and Siddarth Singh Chouhan [8]. They have discussed the various practices and procedures through which an attacker can send the phishing page URL to the user.

Greg Aaron and Rod Rasmussen [9] published a report on Anti-Phishing Working Group (APWG), which presents a brief overview of how bad the current scenario of cybercrime is with respect to phishing attacks and its constant growth.

Various phishing techniques were studied by Dmytro Iliev and Yong Bin Sun [10] and they suggest mutual authentication as a tool to safeguard users from phishing attacks. According to their proposed method the user has to select the correct avatar from a displayed list of avatars in order to enter the password.

Rachna Dhamija and J.D. Tygar [11] have suggested an interesting method to prevent phishing attack, which is based on visual cryptography. They have a dynamic skin which generates an image on login page which is only revealed if the user enters correct password, the user then has to verify the background image visually to be the correct one.

An anti-phishing framework is proposed by Divya James and Mintu Philip [12] based on visual cryptography, in which an image is split into two shares, and one is stored on the server while other remains with the user. During login both user and server share of image is combined, if a meaningful image captcha is visible to the user then they can trust the website.

P P N G Phani Kumar and Dr. R. John Mathew [13] propose a method based on visual cryptography, splitting an image in to two different shares, one stored at the server while other remains with the user, during login both shares are combined to generate a meaningful captcha. In addition to the captcha they use two-tire validation using an OTP, which adds additional layer of security.

A recent report [14] from APWG, an international consortium for phishing attacks, states 291,096 individual phishing website and 592,335 phishing emails were reported by the users from January to June 2017. Attackers used a total of 108,680 domain names and targeted 2660 brands over the same period.

Engin Kirda and Christopher Kruegel [15] suggest an anti-phishing application named AntiPhish which is integrated into the browser, it analyses the data entered on a webpages' HTML form element and the corresponding domain, if the user has visited same domain and entered data before as per history stored on the application then it is considered to be a legitimate website, else the user is alerted.

Gastellier-Prevost, et al. [16] have proposed a toolbar named Phishark which uses 20 heuristic tests such as checking for short URLs, classification on the basis of IP address, testing form fields, testing page title against domain name, spelling check of domain names and checking the use of suspicious characters in the URL to determine the authenticity of a webpage.

DOMAntiPhish is an extension proposed by Rosiello, et al. [17] which analyses a webpage based on its layout. If the given webpage's DOM structure exceeds a certain threshold, then phishing alert is triggered.

A different LinkGuard algorithm is suggested by U.Naresh, et al. [18] which classifies a phishing page and a legitimate page based on the anchored text and the actual hypertext link. If the visual link is different from the hypertext link then the algorithm extracts and compares both the DNS names, if they don't match it is categorized as a phishing page and the algorithm flags the email as one of 5 categories of phishing based on the type of hyperlink in the email

It is found that Wombat Security's latest report [19], 76% of professionals in information security revealed that their organization experienced phishing attacks in 2017. Those campaigns represent a slight increase over the previous year. Additionally, the number of InfoSec professionals whose organizations weathered a USB-based social engineering attack declined by a quarter from 2016 to three percent. Finally, more than half (53%) of respondents witnessed spear phishing attacks in 2017, as compared to the 66% of professionals who did so in 2016.

3. The Proposed Framework

The suggested framework changes the traditional approach of sign-up/sign-in to a website which asks for username and password as input on the same login page. The framework must be able to safeguard the users form various kinds of phishing attacks. The framework consists of the browser extension as well as few changes in the way login pages are used now-a-days. Since phishing attacks are possible only because of human error, using the extension eliminates the chance of the attacker fooling the user. The framework will also work to automated login procedure implemented by new way of logging into a website. Assume that the channel of communication between the server and the client is secure during registration, these credentials are only with the user and the legitimate server during registration.

This framework requires few changes in the traditional approach to sign up and log in of a website:

3.1. Sign up Phase

These are the one-time setup steps which must be done by the user while registering on the website for the first time;

- Step 1: Client opens the website and starts with registration phase. A few essential attributes such as username, password, a passphrase (confidential and must be treated as a secondary password), watermark image (any image whose digital copy is not available to the public is recommended) must be provided by the user during registration.

- Step 2: All the data related to the particular user is stored at the website's hosting server. The username combined with the passphrase will be unique for every user.

After the user completes the above-mentioned steps correctly, he/she is then notified of successful registration on the website. The user also needs to enter username and passphrase in a browser extension if it is installed on the system so that the login process becomes semi-automated partially and the extension starts monitoring for any login activities on a configured domain from this point of time.

3.2. Sign in Phase

After the user has signed-up for a website account he/she need to log in to the website to access it. This is very acute stage because the phishing attack is performed in this phase. This phase requires the user to provide username and passphrase combination with a special symbol "~" in between, it acts as a separator between the username and the passphrase. A tilde (~) symbol is chosen to be a separator due to the rarity of its use [20]. When the server receives this data, it treats the first part before "~" as username and the part after tilde symbol as a passphrase.

- Step 1: When the user starts the login procedure on a website, he/she will be asked for username with the combination of passphrase separated by "~" symbol. This combination is unique for each user. In this phase, if the browser extension is installed then it will be done automatically by extension, and the user will not be presented with the username screen.

- Step 2: A database query is executed at the server side to check if the input is valid.

- Step 3: After getting a right match from the database, a watermark image will be returned to the browser as a personalized password page. (User gets this page directly if the browser extension is installed and properly configured, as soon as the user visits the login page of the website.)

- Step 4: After a visual approval of watermark picture in the background the user will be able to confirm the genuineness of the website. Only then the user should continue to enter login password over the webpage.

- Step 5: Another database query will be executed by the server to validate the password provided by the user.

- Step 6: If the credentials match, access token will be granted to the browser and the user will be logged in to the website.

If the browser extension is installed then the process becomes semi-automated and as soon as the user visits the domain of a website, the extension is triggered and the username with passphrase combination is directly entered by the extension. The user directly gets the personalized password screen. This action will only be triggered if the current domain matches the preconfigured domain, thus making the login mechanism safe from a phishing attack. Even if the browser extension is not installed and user has entered the "username~passphrase", the 2nd step of authentication (missing or incorrect watermark image on the background of the password page) assists the user to sense a possible phishing attack. If the user senses a phishing attack, he/she must then log in to the website from any other system by carefully entering the URL manually in the browser and change the passphrase (as the attacker can only get access to username and passphrase which was entered on the 1st step of authentication).

The data flow as well as the working of the entire framework with and without the extension installed on the browser is shown in Figure 1. The diagram represents how the suggested framework can protect the user from a potential phishing attack in any scenario. The diagram below helps to understand how efficiently the framework can defend users against the phishing attack.

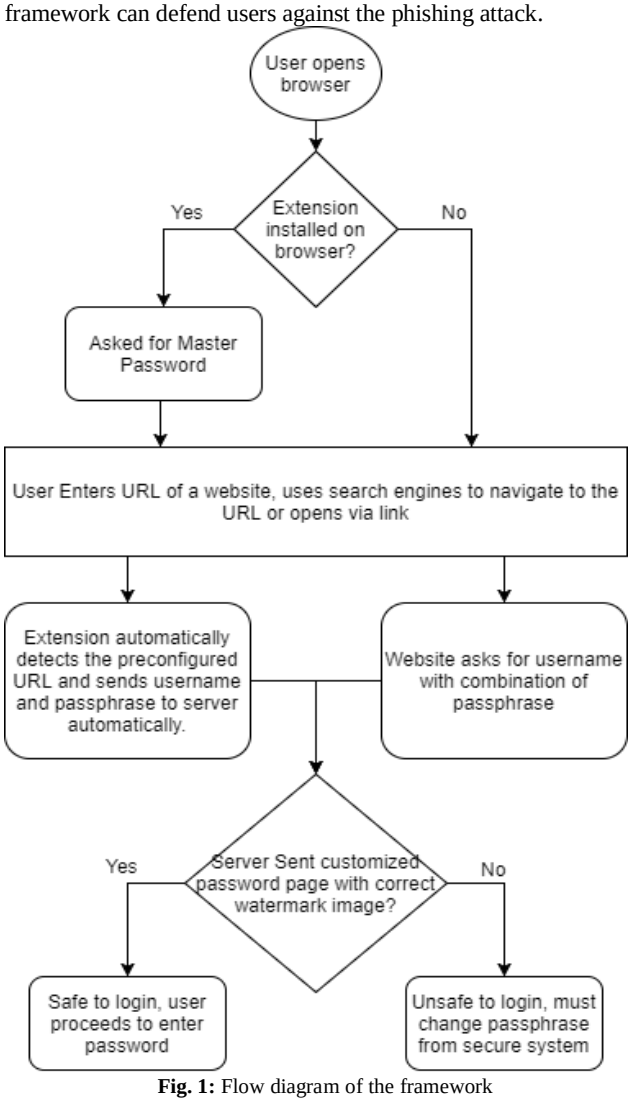

Fig. 1: Flow diagram of the framework

4. Using the Browser Extension

The browser extension is intended to make the 2-step login process semi-automated by completing the 1st step on behalf of user. The extension continually monitors the domain name in the URL and is only triggered if the user is visiting a page served from a genuine domain address (which is preconfigured at the time of sign-up phase). If somehow the user visits a malicious server hosting the phishing page the extension will remain inactive and user will see the username page which will ask for the username and passphrase. At this point of time the user gets the first hint of a

potential phishing attack, as the user must directly get the password page with personalized watermark image because the 1st step is automated by the extension. This eliminates the need for the user to continuously monitor the URL of the webpage. The domain, username, and passphrase stored in the extension are encrypted, and is decrypted with the master password which is used to activate the extension. The password is not stored and remains only with the user. This provides a very strong mechanism for security as an attacker will have to hack the user's password as well as the extension at the same time to gain any useful information. The extension can access any stored data only if the visited domain matches the preconfigured domain list. The activated browser extension backs up the user by checking the URL continuously (where they are most likely to make mistake). The user must see the personalized password page directly if the extension is preconfigured and is active. If the user sees the username page (the extension does not auto-complete this step) then it is an indication of a potential phishing attempt.

5. Results and Discussion

Implementation of the suggested login framework with or without the browser extension will show different behavior in different cases. Table 1 discusses different test cases and behavior in both the cases, the remarks show whether it is a phishing attack or user is safe to proceed. It also describes various actions recommended to the user in different situations.

Table 1: Test Case

CASE	With Browser Extension	Without Browser Extension	Remarks
Opens a registration page	Can configure username, passphrase and URL[a] in the extension	It is just like a normal registration process on the webpage	User is recommended to confiure the username and passphrase into extension if it is installed
Sign-in (not configured extension)	User gets username page asking for username and passphrase	User gets username page asking for username and passphrase	User must verifythe URL carefully befor entering username and passphrase
Sign-in (configured extension)	Extension autofills username and passphrase, redirected to personalized password page with users watermark image	User gets username page asking for username and passphrase	If extension does not trigger automatacly, user must not proceed as it may be potential phishing attack
Phishing page opened	User gets username page asking for username and passphrase (extension remains inactive)	User gets username page asking for username and passphrase	If extension does not trigger automatacly, user must not proceed as it may be potential phishing attack
User enters username & passphrase (Phishing page)	Does not get personalized password page with correct watermark image	Does not get personalized password page with correct watermark image	User must change passphrase from a secure system
User enters username & passphrase	Gets personalalized password page with correct watermark image	Gets personalalized password page with correct watermark	User is visiting genuin server and is safe to proceed
		image	

URL is taken automatically.

The page asks for username and passphrase combination in the first step to login as shown in Figure 2.

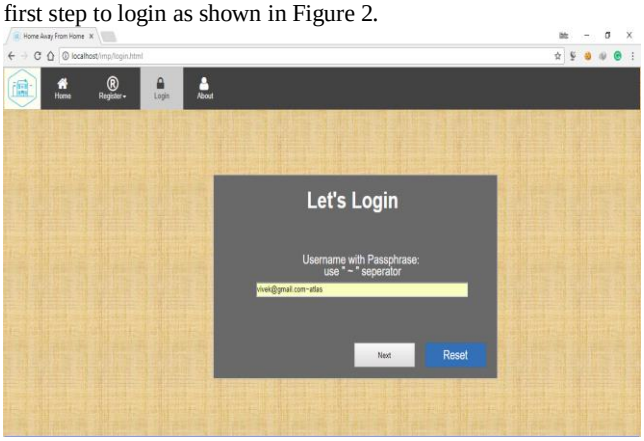

Fig. 2: Username and passphrase page.

After entering the correct username and passphrase combination the user is redirected to a personalized login page as shown in Figure 3. It has a custom watermark image in the background, as set by the user during his/her registration.

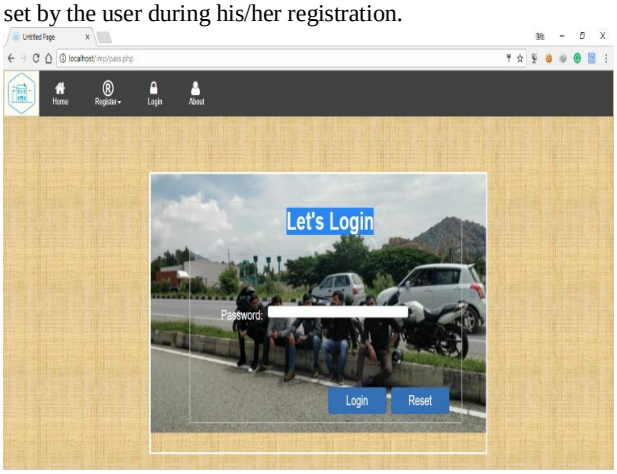

Fig. 3: Personalized password page with custom background image.

The user must automatically be redirected to the password page, as the first step is automatically performed by the browser extension if it is installed and configured. The configuration process is shown in figure 4, in which the user must enter username and passphrase in separate textboxes, the URL will be captured automatically by the extension. If the extension does not act automatically and user is on the passphrase page despite browser extension being installed and configured indicates that the user is on an incorrect domain, which means it may be a phishing page.

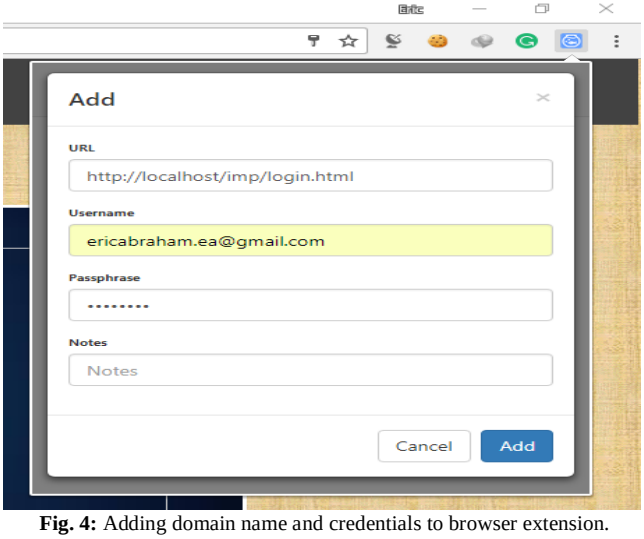

Fig. 4: Adding domain name and credentials to browser extension.

If the background watermark image on the password page is incorrect then the user must not proceed to enter the password in the page as it may be a phishing attempt.
pointed in (1) the…).

6. Conclusion

The suggested framework for login is able to alert the user of any potential phishing attack without the need of the user to unremittingly monitor the URL of the visited page. The semi-automated nature of the browser extension is able to reduce the effort needed to log in to a website by automating the process of URL checking and input of username and passphrase. The framework can work with or without the browser extension and is independent of any pre-requisite in the user environment as it is a mechanism implemented on the server. Hence it works for any user accessing the website from any device irrespective of the platform or the browser. Future scope of work includes replacing single watermark image with multiple images. Those images will be displayed based on various parameters such as date, time, day of the week etc. The user will be able to decide the pattern and sequence of images to be displayed as watermark. This will make the framework even more secure.

References

[1] Singh, Akhilendra Pratap, et al. "Detection and prevention of phishing attack using dynamic watermarking." Information Technology and Mobile Communication. Springer, Berlin, Heidelberg, 2011. 132-137.
[2] Chen, Juan, and Chuanxiong Guo. "Online detection and prevention of phishing attacks." Communications and Networking in China, 2006. ChinaCom'06. First International Conference on. IEEE, 2006.
[3] Suganya, V. "A Review on Phishing Attacks and Various Anti Phishing Techniques." International Journal of Computer Applications (0975–8887) Volume.
[4] Sharifi, Mohsen, et al. "A zero knowledge password proof mutual authentication technique against real-time phishing attacks." International Conference on Information Systems Security. Springer, Berlin, Heidelberg, 2007.
[5] Miyamoto, Daisuke, Hiroaki Hazeyama, and Youki Kadobayashi. "An evaluation of machine learning-based methods for detection of phishing sites." International Conference on Neural Information Processing. Springer, Berlin, Heidelberg, 2008.
[6] Atighetchi, Michael, and Partha Pal. "Attribute-based prevention of phishing attacks." Network Computing and Applications, 2009. NCA 2009. Eighth IEEE International Symposium on. IEEE, 2009.
[7] Parno, Bryan, Cynthia Kuo, and Adrian Perrig. "Phoolproof phishing prevention." Financial Cryptography. Vol. 4107. 2006.
[8] Chawla, Minal, and Siddarth Singh Chouhan. "A survey of phishing attack techniques." International Journal of Computer Applications 93.3 (2014).
[9] Greg Aaron and Rod Rasmussen "Unifying the Global Response To Cybercr ime" Global Phishing Survey 2015-2016 by APWG published 26 June 2017.
[10] Iliyev, Dmytro, and Yong Bin Sun. "Website forgery prevention." Information Science and Applications (ICISA), 2010 International Conference on. IEEE, 2010.
[11] Dhamija, Rachna, and J. Doug Tygar. "The battle against phishing: Dynamic security skins." Proceedings of the 2005 symposium on Usable privacy and security. ACM, 2005.
[12] James, Divya, and Mintu Philip. "A novel anti phishing framework based on visual cryptography." Power, Signals, Controls and Computation (EPSCICON), 2012 International Conference on. IEEE, 2012.
[13] Kumar, P. P. N. G., and R. John Mathew. "An Advanced Anti Phishing Approach Based On Two-Tier Validation." IJRCCT 3.9 (2014): 1015-1017.
[14] Anti-Phishing Working Group. Phishing Activity Trends Report. [https://www.antiphishing.org/resources/apwg-reports/], 2017.
[15] Kirda, Engin, and Christopher Kruegel. "Protecting users against phishing attacks with antiphish." Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International. Vol. 1. IEEE, 2005.
[16] Gastellier-Prevost, Sophie, Gustavo Gonzalez Granadillo, and Maryline Laurent. "Decisive heuristics to differentiate legitimate from phishing sites." Network and Information Systems Security (SAR-SSI), 2011 Conference on. IEEE, 2011.
[17] Rosiello, Angelo PE, Engin Kirda, and Fabrizio Ferrandi. "A layout-similarity-based approach for detecting phishing pages." Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on. IEEE, 2007.
[18] Naresh, U., U. VidyaSagar, and C. V. MadhusudanReddy. "Intelligent Phishing Website Detection and Prevention System by Using Link Guard Algorithm." IOSR Journal of Computer Engineering (IOSR-JCE) 14 (2013): 28-36.
[19] Organizations Experienced Phishing Attacks 2017. "State of the Phish™ Report 2018". [https://www.tripwire.com/state-of-security/security-data-protection/three-quarters-organizations-experienced-phishing-attacks-2017-report-uncovers/], 2018.
[20] Samuel Arbesman, "The Rarity of the Ampersand: Frequencies of Special Characters", [https://www.wired.com/2013/08/the-rarity-of-the-ampersand/], 2013.