International Journal of Information Technology and Knowledge Management
July-December 2011, Volume 4, No. 2, pp. 545-547

SENTRY: TRANSFORMING PHISHING DETECTION INTO AUTOMATIC FRAUD PROTECTION

Inturi Sirisha[1] & C.Shoba Bindu[2]

Identity theft refers to the use of another person's identity, usually for the financial gain or for defamatory purposes. Phishing is one method used to perform an identity theft. Many anti-phishing mechanisms focused on the prevention based approaches or detection based approaches, those alone fail to suppress the phishing attacks. In this paper, SENTRY uses a combination of both the approaches and transforms the phishing detection into automatic fraud protection. It detects spoofed web pages where users are intended to submit their information and generates juke, meaningful credentials to hide the real information of the web users. Whenever a phisher tries to steal, it efficiently identifies the identity theft. It is made as an extension plug-in to the Firefox web browser. The efficiency is evaluated in the real environment and is compared with the existing anti-phishing proposals.

Keywords: Web Spoofing, Identity Theft, Phishing, Credential Theft, Security

1. Introuction

The term phishing was coined by crackers to refer to the act of tricking people into revealing the sensitive or private information. It relies on the fact that asking a large number of people for this information will always fool at least a small number of people. In a phishing attempt, the attacker would typically create a situation where people believe that they are dealing with an authorized party, such as their bank. The attacker will then ask the victim for sensitive information such as credit card details. Much of this activity is automated and the target is typically a large number of internet users. Phishing is considered as an opportunistic attack rather than a targeted one.

Careful analysis of the phishing problem promises to shed light on a wide range of security problems. To defend against phishing attacks, a number of client-side and server-side defenses have been proposed and developed. Neither of them are successful in preventing vulnerable users from deceived. These different approaches are prevention based approaches, which alone fail to shield the vulnerable users.

The proposed system, SENTRY is developed as an extension to the Bogus Biter [5]. Bogus Biter is turned on once a login web page is classified as a phishing page by a web browser's built in phishing detection component or a third party toolbar. Sentry with its stronger algorithm can find whether a web page is genuine or not by verifying it on the browser itself, reducing the time for genuinity identification of the web pages. Sentry is developed as operating system independent and can run on unix, linux based machines. Phisher can explore the limitations of the Bogus Biter to circumvent it. The bogus credentials that are generated by the Bogus Biter can be analyzed by the phisher using local username filtering techniques, meaningful credential filtering techniques or statistical filtering techniques. Sentry uses phonetics, which helps in producing all meaningful juke credentials rectifying the problem in Bogus Biter.

The paper is organized as follows: Section 2 describes the working of Sentry, the implementation, results of Sentry, Section 3 draws some conclusions and Section 4 discusses future work.

[1,2]JNTUCEA, Anantapur

E-mail: [1]sirisha.inturi@gmail.com , [2]shobabindhu@gmail.com

2. Working of Sentry

In this section, we describe how the Sentry differentiates fake web pages and how the meaningful juke credits are generating. The results are produced and compared at the end of the section.

2.1 Identifying Fake Pages

Sentry is developed as an extension to the web browser. The very first stage of Sentry is identifying the fake web pages. For a webpage to be safe to log into it must be genuine and secure. To find whether the webpage is genuine and/or secure the test must be as follows:

- Check the <title> and look for word before .tld, example: Check for the word "Facebook" in <title> when, www.facebook.com is opened. Anything before .com, .net or any .tld, that word needs to be in the <title>
- The above criteria hold only, if the website is genuine or potentially genuine but, does not guarantee "secured". Thus, check security by understanding the encryption.

- Check for " https://", it is secured http, running on port 443 for Apache and has static IP, thus, assures prevention to eavesdropping.
- If website has name in <title>, i.e. domain name in title and has HTTPS, then, it is Genuine + Secured.
- If website has name in <title>, i.e. domain name but, does not guarantee security, then, it is Genuine but maybe not secured. Sentry just displays a warning.
- If website <title> does not match, anything before .tld, then site is forged and Sentry will bombard fake bites.
- Along with the genuinity test and verifying SSL certificate, Sentry sorts out the spoofed anchor,domain name inconsistency,cross site scripting for identifying fake web pages.

2.2 Generating Juke Credentials

After successfully identifying the fake web page Sentry hides the actual credentials by generating fake credentials using a strong algorithm in which the number of fake credentials that are to be generated is a positive real number. When the number of credentials generated reaches the maximum limit specified, the generation algorithm stops execution. A part of the juke credential creation algorithm is shown below:

```
Global var Gen_credential_num=0;
Generate(credential)
{If(charAt(credential(i)=='a'))
Then
    { Replace it with 'ea'
      New_credential:= Replaced credential;
      Gen_credential_num=Gen_credential_num+1
       Generate(new_credential)
     }
If(charAt(credential(i)=='i'))
Then
    { replace it with 'ee'
      New_credential:= Replaced credential;
      Gen_credential_num=      Gen_credential_num+1
       Generate(new_credential)
     }
.
.
.
}
```

Fig.1: Sample Template to Generate Juke Credentials

For every credential having number as a part of it, Sentry will generate credentials having altered number. The text is modified using phonetical alterations. E.g., let sri12 is the actual credential, then sentry will generate sri11, sri12, sri13, sree11, sree12, sree13 etc. Thus local username filtering techniques fails in identifying the meaningful credentials making it difficult for phisher to succeed.

When the user login page is classified as a fake page then the sentry bombards juke credentials and directs all the N set size number of credentials to the phisher. Phisher's local filtering techniques can't identify the actual credentials; as a result he has to check all the credentials in legitimate website to sort out the actual credentials. The Defensive Sentry Algorithm identifies the credentials that are being produced by the phisher as the juke credentials it had generated.

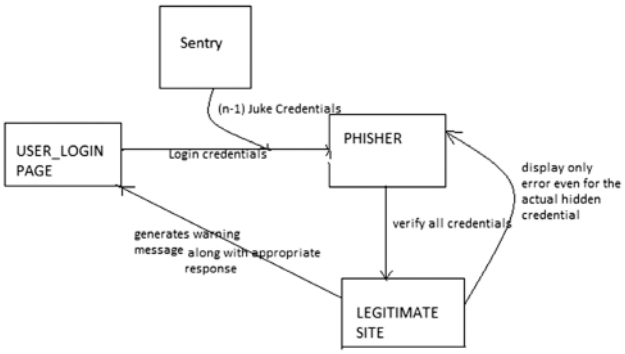

Fig. 2: Steps Involved in Working of Sentry

It then sorts the actual credentials among the juke credentials and intimates the User that he is safe as of now and has to take care of the phisher. It also generates only one message "error" to the phisher for any credential he produce to the legitimate site. The Diagrammatical representation is provided in Fig1.

2.3 Results and Comparisons

If the set size S is 4 or 8, for over 85% of phishing sites, the delay is less than 4 seconds and for the legitimate sites the delay is less than 3 seconds, and for over 85% of legitimate sites the delay is less than one second in case of bogus biter [5].

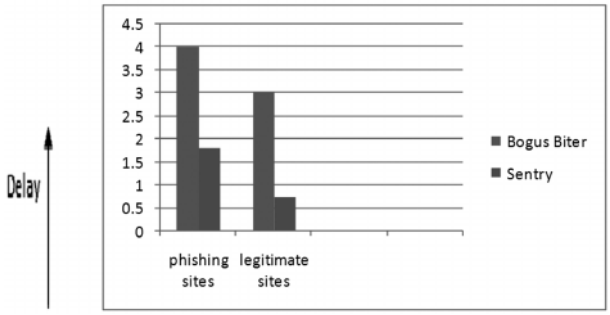

Fig.3: Indicating Delay in Sentry and Bogus Biter for Legitimate Websites, Phishing Sites

Coming to the Sentry when the N value is 4 or 8 for more than 80% of phishing sites, the delay is 1.78 seconds and for 90% of the legitimate sites it is 0.73 seconds.

3. CONCLUSION

Sentry is developed using ideas from Bogus Biter. The result is a Firefox extension overcoming the offline evasions like local username filtering, meaningful credential filtering that are in Bogus Biter. The effectiveness of Sentry is more, comparatively in terms of time delay. We believe that it will make a useful contribution to the Internet-Technology research.

4. FUTURE WORK

Sentry can have larger reach. Multi-browser standalone system can be implemented, not just limited to Firefox and Internet Explorer. Sentry has the limitations of online evasions these can be reduced in the later versions.

ACKNOWLEDGMENTS

We gratefully acknowledge suggestions from Rajasekhar. Thanks to Vivek and Kevel for their valuable services and help.

REFERENCES:

[1] BIRK, D., DORNSEIF, M., GAJEK, S., AND GR¨OBERT, F. 2006, "Phishing Phishers— Tracing Identity Thieves and Money Launderer", Tech. rep. Horst-G¨ortz Institute of Ruhr- University of Bochum.

[2] BORTZ, A., BONEH, D., AND NANDY, P. 2007, "Exposing Private Information by Timing", Web Applications, In Proceedings of the International World Wide web Conference, (WWW).621–628.

[3] CHIASSON, S., VAN OORSCHOT, P. C., AND BIDDLE, R. 2006, "A Usability Study & Critique of two Password Managers", In Proceedings of the USENIX Security Symposium 1–16.

[4] CHOU, N., LEDESMA, R., TERAGUCHI, Y., AND MITCHELL, J. C. 2004, "Client-side Defense Against Webbased Identity Theft", In Proceedings of the Network and Distributed System Security Symposium(NDSS).

[5] CHUAN YUE, HAINING WANG. 2010, "Bogus Biter: A Transparent Protection Against Phishing Attacks", In ACM Transactions on Internet Technology, 10, No.2, Article 6.

[6] DHAMIJA, R. AND TYGAR, J. D. 2005, "The Battle Against Phishing: Dynamic Security Skins", In Proceedings of the Symposium on Usable Privacy and Security (SOUPS). 77–88.

[7] DHAMIJA, R., TYGAR, J. D., AND HEARST, M. 2006., Why phishing works. In Proceedings of the Conference on Human Factors in Computing Systems (CHI), 581–590.

[8] FELTEN, E. W., BALFANZ, D., DEAN, D., AND WALLACH, D. S. 1997. Web Spoofing:, "An Internet Con Game", In Proceedings of the 20th National Information Systems Security Conference.

[9] JAKOBSSON, M. AND RATKIEWICZ, J. 2006., Designing Ethical Phishing Experiments: A study of (ROT13) rOnl Query Features", In Proceedings of the International World Wide Web Conference(WWW). 513–522.

[10] JAKOBSSON, M. AND YOUNG, A. 2005., "Distributed Phishing Attacks. In Proceedings of The Workshop on Resilient Financial Information Systems. KANDULA, S., KATABI, D.,

[11] JACOB, M., AND BERGER, A. W. 2005. Botz-4-Sale: "Surviving Organized DDoS Attacks that Mimic Flash Crowds", In Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI). 287–300.

[12] KIRDA, E. AND KRUEGEL, C. 2005., Protecting Users Against Phishing Attacks with AntiPhish, In Proceedings of the Annual International Computer Software and Applications Conference (COMPSAC). 517–524.

[13] KLEIN, D. V. 1990., "Foiling the Cracker—A Survey of, and Improvements to, Password Security, "In Proceedings of the 2nd USENIX Workshop on Security", 5–14.

[14] KUMARAGURU, P., RHEE, Y., ACQUISTI, A., CRANOR, L. F., HONG, J., AND NUNG, E. 2007, "Protecting People from Phishing: The Design and Evaluation of an Embedded Training Email System", In Proceedings of the Conference on Human Factors in Computing Systems (CHI). 905–914.

[15] PARNO, B., KUO, C., AND PERRIG, A. 2006, "Phoolproof Phishing Prevention", In Proceedings of the Financial Cryptography. 1–19.