Web Browser Independent Mechanism for Preventing Malware based on Layered Service Provider

[1]Ho-mook Cho, [2]Sang-Yong Choi, [3]Yong-Min Kim
[1, First Author]*Interdisciplinary of Information Security, Chonnam National University /KAIST CSRC, Korea, chmook79@kaist.ac.kr*
[2]*KAIST Cyber Security Research Center, Korea, csyong95@kaist.ac.kr*
[*3,Corresponding Author] *Dept. of Electronic Commerce, Chonnam National University, Korea, ymkim@jnu.ac.kr*

Abstract Web based Malware attacks and distribution techniques become more complex and intellectual nowadays. But most of user defense techniques rely on web-browsing and this is not enough for effective defense to cover all malware attacks. In this paper, we propose web-browser independent method based on Layered Service Provider. Our evaluation result shows that the proposed method successfully filters and blocks URLs without any intervention of web-browsers.

Keywords*: Malicious URL detection, Web-based Malware, Layered Service Provider*

1. Introduction

With the advance of web technology, attackers can choose variety of offense strategies. One of most serious attacks is Drive-by download attack and this strategy become more intellectual nowadays [1]. Variety of defense techniques were developed and currently commercialized to cover these attacks. But, most of defense techniques operate solely with specific web-browser. Thus, there are limitations to effectively defense all malware attacks since the attacks are not web-browser dependent and distribution flows without any limitations.

In this paper, we propose Layered Service Provider (LSP) [7] based method to overcome and improve from current limitations.

2. LSP-based URL blocking method

2.1 URL block technology

Currently, URL blocking solutions include Google's Safe Browsing [2], Smart Screen Filter for Microsoft [3], McAfee’s Site Advisor [4], Symantec's Safe Web [5], and Web of Trust [6]. As shown in the Table 1, these solutions are implemented as web-browser dependent components such as plug-ins or built-in features. However, attackers can exploit vulnerabilities of web-browser independent third-party applications such as Java and Adobe Flash Player. Such attacks can evade existing solutions. Thus, a web-browser independent URL blocking technique would be a more effective countermeasure.

Table 1. Existing URL blocking solutions

Software	**IE**	**Chrome**	**Firefox**	**Safari**
Safe Browsing		○	○	
Smart Screen Filter	○			
Site Advisor	○	○	○	
Safe Web	○	○	○	○
Web of Trust	○			

Web-browser independent URL blocking techniques can be implemented based on client\-side web proxy, Network Driver Interface Specification (NDIS), and LSP methods. The

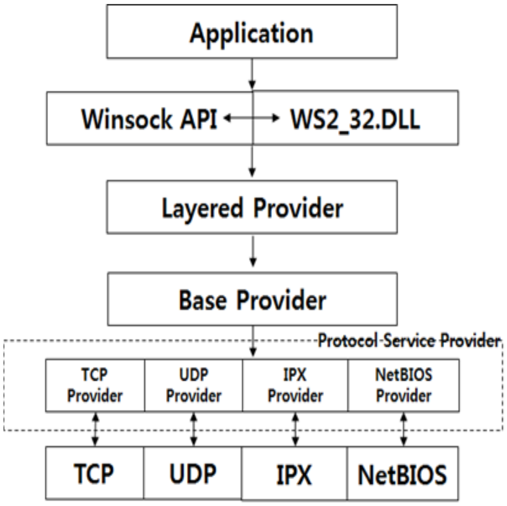


Figure 1. Winsock Architecture

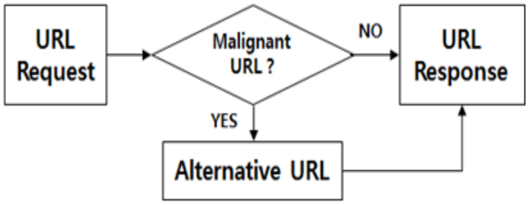


Figure 2. Mechanism for Preventing Malware based on LSP

proxy-based method requires a user to configure a network proxy setting in an operating system which degrades end-user convenience. NDIS-based method is much rely on a network device and it also needs to analyze HTTP contents in a separate procedure. In contrast, LSP-based method is transparent to the OS configuration and supports various application layer protocols.

2.2 Proposed method

In this paper, we propose LSP-based web-browser independent URL blocking method. Showing on Figure 1, LSP is located between Protocol Service Provider and Application Layer, so it can intercept application's network contents.

Procedure of the actual operation in LSP is shown on Figure 2. The proposed LSP-based URL blocking method checks if a requested URL exists in the malicious URL list. If the URL is detected as malicious one, it responses with an alternative webpage, otherwise it allows users access to an original webpage. This method can handle all application-generated HTTP requests without any web-browser dependencies.

3. Evaluation

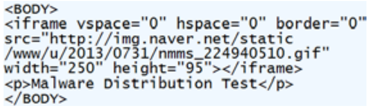
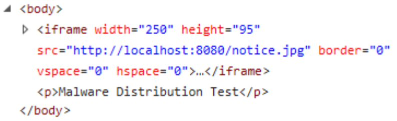
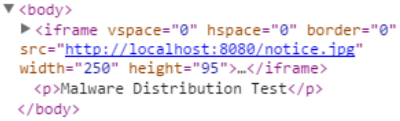
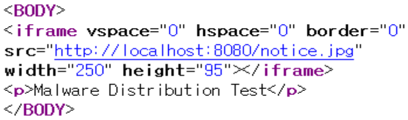
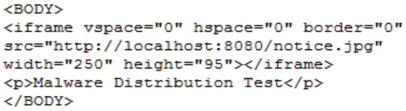
3.1 Methodology

To evaluate the proposed method, we tested the most widely used 4 web-browsers including IE, Chrome, Firefox, Safari. First, we started with malicious URL to initiate connection with testing browser and confirmed that the connection is normal. Next, we enabled the proposed method and let the browsers visit the same URLs again. Then, we check whether malignant URL is successfully filtered or blocked.

3.2. Result

Table 2 shows that the proposed method extracts and blocks the malicious URL for all 4 web-browsers and it replaces the malicious URL with the URL of alternative webpage. Thus, the proposed LSP-based URL blocking method was confirmed independently applicable to a web-browser.

Table 2. Evaluation result with 4 web-browsers

Malware Code	 <BODY> <iframe vspace="0" hspace="0" border="0" src="http://img.naver.net/static /www/u/2013/0731/nmms_224940510.gif" width="250" height="95"></iframe> <p>Malware Distribution Test</p> </BODY>
IE	 ◢ <body> ▷ <iframe width="250" height="95" src="http://localhost:8080/notice.jpg" border="0" vspace="0" hspace="0">…</iframe> <p>Malware Distribution Test</p> </body>
Chrome	 ▼ <body> ▶ <iframe vspace="0" hspace="0" border="0" src="http://localhost:8080/notice.jpg" width="250" height="95">…</iframe> <p>Malware Distribution Test</p> </body>
Firefox	 <BODY> <iframe vspace="0" hspace="0" border="0" src="http://localhost:8080/notice.jpg" width="250" height="95"></iframe> <p>Malware Distribution Test</p> </BODY>
Safari	 <BODY> <iframe vspace="0" hspace="0" border="0" src="http://localhost:8080/notice.jpg" width="250" height="95"></iframe> <p>Malware Distribution Test</p> </BODY>

4. Conclusion

In this paper, we propose LSP-based URL blocking method to overcome the limitations of current web-browser dependent blocking schemes. Our evaluation result shows that the proposed method successfully prevents users from visiting malicious URLs without any web-browser dependencies.

In the future work, we plan to extend the method to analyze HTTPS contents encoded at the application layer.

References

[1] ENISA, "Threat Landscape report," "http://www.enisa.europa.eu/activities/riskmanagement/evolving-threat-environment/ENISA_Threat_Landscape," Jan. 2013.

[2] Safe Browsing, https://developers.google.com/safe-browsing/

[3] Smart Screen Filter, http://windows.mic-rosoft.com/ko-kr/internet-explorer/products-/ie-9/features/smartscreen-filter

[4] Site Advisor, http://www.siteadvisor.com/

[5] Safe Web, http://safeweb.norton.com/

[6] Web of Trust, https://www.mywot.com/

[7] Hua, Wei, Jim Ohlund, and Barry Butterklee. "Unraveling the mysteries of writing a winsock 2 layered service provider." Microsoft Systems Journal-US Edition: 51, May, 1999.