International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 5, 2017 ISSN: 2348-6600
Reference ID: IJCS-178 PAGE NO: 1084-1088
Sri Vasavi College, Erode Self-Finance Wing 3rd February 2017
National Conference on Computer and Communication NCCC'17
http://www.srivasavi.ac.in/ nccc2017@gmail.com

WEB SPY DEFENSE SYSTEM

P.SATHYA#1, P.PARAMESHWARI#2, JULIET ROZARIO#3

#*Assistant Professor, Department of Computer Science*
Sree Saraswathi Thyagaraja College, Pollachi – 642107, TN, India
1sathyakamalrajkr@gmail.com
2 params2005@yahoo.co.in
3 rozario.juliet@gmail.com

ABSTRACT- Click jacking aims in securing web based applications against click jacking attacks. Web applications have evolved from simple collections of static HTML documents to complex, fullfledged applications containing hundreds of dynamically generated pages. The combined use of client and server-side scripting allows developers to provide highly sophisticated user interfaces with the look-and-feel and functionalities that were previously only reserved to traditional desktop applications.

This unit is responsible for detecting and logging any click jacking attacks that are contained in the web page under analysis. The detection is handled by two browser plug-ins. The first component is a solution that we developed to detect when multiple clickable elements co-exist and overlay in the region of the page where the user has clicked. We call our detection solution ClickIDS. The second plug-in is the modified version of the NoScript open-source tool that saves the generated alerts into a database instead of displaying popups to the user.

Keywords— ClickIDS,NoScript, Click jacking Attack, Stopper.

INTRODUCTION

Web spy is a web-based attack that has recently received wide media coverage. In a click jacking attack, a malicious page is constructed such that it tricks victims into clicking on an element of a deferent page that is only barely (or not at all) visible. By stealing the victim's clicks, an attacker could force the user to perform an unintended action that is advantageous for the attacker (e.g., initiate an online money transaction). Although click jacking has been the subject of many discussions and alarming reports, it is currently unclear to what extent click jacking is being used by attackers in the wild, and how significant the attack is for the security of Internet users.

Click IDs:

ClickIDS is the browser plug-in that we implemented. It intercepts the mouse click events, checks the interactions with the elements of a web page, and detects click jacking attacks.

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 5, 2017 ISSN: 2348-6600
Reference ID: IJCS-178 PAGE NO: 1084-1088
Sri Vasavi College, Erode Self-Finance Wing 3rd February 2017
National Conference on Computer and Communication NCCC'17
http://www.srivasavi.ac.in/ nccc2017@gmail.com

The basic idea behind ClickIDS is simple. A suspicious behavior is reported when two or more clickable elements of different pages overlap at the coordinates of the mouse click. As clickable elements, we consider links (or, more precisely, the area enclosed between HTML <A> tags), buttons, and form inputs fields such as checkboxes, radio buttons, menu, and text fields. In addition, we also take into account Adobe Flash content, embedded in HTML with <EMBED> tags and associated with the application-type x-shockwave-flash.

We motivate the consideration of Flash content in two ways. First, when click jacking was first reported in October 2008, it gained interest fast mainly because a click jacking exploitation technique against the Adobe Flash Player Setting Manager would have permitted to modify the web-cam and microphone security settings. Basically this exploit allowed an attacker to remotely turn the user's computer into an eavesdropping device. Second, for some advanced attacks to be successful, an attacker would need to steal multiple user-clicks, and therefore, would prefer to overlay the click jacked site with flash content (e.g., a game) that persuades the user to perform several clicks. When our plug-in is loaded, it registers to the document event load. Each time a new page is loaded, the page-handler routine is executed. This routine registers the loaded page and attaches a second click-handler to it. At this point, every click of the user in the context of the web page is intercepted, and handled by the click-handler routine.

If the clicked element is clickable (according to our previous definition), we register the current mouse coordinates. Then, we scan the main page and the contained FRAMEs and IFRAMEs to check if they contain clickable elements at the same position. If there exists at least one element that overlays the clicked one, we generate an alert. ClickIDS is more precise in identifying attacks based on overlapping elements. However, unlike NoScript, it is not able to detect attacks based on partially obstructed pages. Nevertheless, the combination of the two different techniques can effectively reduce the number of false positives generated by the tools individually. Note that we also developed a third component that we call Stopper, which drops mouse events after all the registered listeners have been successfully executed. This prevents the browser from actually opening a new page, submitting a form, or downloading a file in response to the mouse clicks.

No Script:

NoScript is a Firefox add-on that provides protection against common security vulnerabilities such as cross-site scripting. It also features a URL access-control mechanism that filters browser-side executable contents such as Java, Adobe Flash, and Microsoft Silverlight. In October 2008, an anti-click jacking feature was integrated into NoScript. This feature protects users against transparent IFRAME-based attacks. Starting from version 1.8.2, the protection has been extended to cover also partially

Obstructed and disguised elements. The implemented technique, denoted Clear Click, resembles one proposed by Zalewski , and is

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 5, 2017 ISSN: 2348-6600
Reference ID: IJCS-178 PAGE NO: 1084-1088
Sri Vasavi College, Erode Self-Finance Wing 3rd February 2017
National Conference on Computer and Communication NCCC'17
http://www.srivasavi.ac.in/ nccc2017@gmail.com

based on the analysis of the click's neighborhood region. An alert is triggered when a mouse click is detected in a region where elements from different origins overlap.

The coordinates of the web page's clickable elements arereceived from the element extractor, a custom extensionthat we installed in the browser. This component is registered to the page-open event such that each time a pageis loaded, a callback function is called to parse the page'sDOM, and to extract all the information about the clickableelements. The plug-in also extracts information concerningall the FRAMEs and IFRAMEs included in the visited page,including their URL and opacity values.

EVALUATION

To test the effectiveness of our prototype tool in detecting clickjacking attacks, we first created five different testpages, based on the examples published on the Internet,that contained clickjacking attacks,. In all cases, the system correctly raised an alert message to report the attack.Having initially validated our approach on these testpages, we set out to test the effectiveness of our system in identifying real-world websites containing similar,previously-unknown clickjacking attacks. We combined different sources to obtain an initial list of URLs that is representative of what an average user may encounter in hereveryday web browsing experience. More precisely, we included the top 1000 most popular websites published byAlexa over 20,000 profiles of the MySpace social network, and the results of ad-hoc queries on popular searchengines. In particular, we queried Google and Yahoo withvarious combinations of terms such as "porn," "free download," "warez," "online game," "ringtones," and "torrents."

We ran each query in different languages including English,German, French, Italian, and Turkish.

Framebreaker script:

To prevent malicious scripts embedding frames in the web page. A frame breaker script is embedded in every web page of the application.

```
<script>if (top != self) top.location=location</script>
```

In web application server script has a frame checker which would break any clickjack attacks.

Following is the configuration file for the servlet in the java web application.**Web.xml**

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app id="WebApp_ID" version="2.4"
xmlns="http://java.sun.com/xml/ns/j2ee"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee http://java.sun.com/xml/ns/j2ee/web-app_2_4.xsd">
<display-name>OWASP ClickjackFilter</display-name>
<filter>
<filter-name>ClickjackFilterDeny</filter-name>
<filter-class>org.owasp.filters.ClickjackFilter</filter-class>
```

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 5, 2017 ISSN: 2348-6600
Reference ID: IJCS-178 PAGE NO: 1084-1088
Sri Vasavi College, Erode Self-Finance Wing 3rd February 2017
National Conference on Computer and Communication NCCC'17
http://www.srivasavi.ac.in/ nccc2017@gmail.com

```
<init-param>
<param-name>mode</param-name>
<param-value>DENY</param-value>
</init-param>
</filter>

<filter>
<filter-name>ClickjackFilterSameOrigin</filter-name>
<filter-class>org.owasp.filters.ClickjackFilter</filter-class>
<init-param>
<param-name>mode</param-name>
<param-value>SAMEORIGIN</param-value>
</init-param>
</filter>

<!-- use the Deny version to prevent anyone, including yourself, from framing the page -->
<filter-mapping>
<filter-name>ClickjackFilterDeny</filter-name>
<url-pattern>/*</url-pattern>
</filter-mapping>

<!-- use the SameOrigin version to allow your application to frame, but nobody else
<filter-mapping>
<filter-name>ClickjackFilterSameOrigin</filter-name>
<url-pattern>/*</url-pattern>
</filter-mapping>
  -->
```

```
</web-app>
```

DISCUSSION

Around 5% of the alerts raised during our experimentsinvolved a frame pointing to the same domain of the mainpage. Since it is very unlikely that websites would try totrick the user into clicking on a hidden element of the siteitself, we marked all these messages as being false positives.

However, we decided to manually visit some of these pagesto have an insight into what kind of conditions tend to causea false positive in the two plug-ins.We then carefully analyzed the pages containing crossdomain frames. In this set, we identified a number ofinteresting cases that, even though not corresponding toreal attacks, matched our definition of web spy defense system. Wedecided to divide these cases in two categories: The truepositives contain real clickjacking attempts, while theborderline cases contain pages that were difficult to classifyas being clickjacking.

CONCLUSION

"Web Spy Defense System" was mainly developed to make it convenient for the user to search data or word in a website. This unit is responsible for detecting and logging any click jacking attacks that are contained in the web page under analysis. The detection is handled by two browser plug-ins. The first component is a

International Journal of Computer Science
Scholarly Peer Reviewed Research Journal - PRESS - OPEN ACCESS
ISSN: 2348-6600
http://www.ijcsjournal.com Volume 5, Issue 1, No 5, 2017 ISSN: 2348-6600
Reference ID: IJCS-178 PAGE NO: 1084-1088
Sri Vasavi College, Erode Self-Finance Wing 3rd February 2017
National Conference on Computer and Communication NCCC'17
http://www.srivasavi.ac.in/ nccc2017@gmail.com

solution that we developed to detect when multiple clickable elements co-exist and overlay in the region of the page where the user has clicked. We call our detection solution *ClickIDS*. The second plug-in is the modified version of the NoScript open-source tool that saves the generated alerts into a database instead of displaying popups to the user.

By using this software, they can take quick decisions and preventive actions based on the details given by the system. Due to the software, I hope quality will be improved, problems will be solved.It is user friendly system provided with options, which can be utilized by the desired operations. The new system overcomes the problems encountered with the old system.

6. REFERENCES

[1] Acutenix web security scanner.
http://www.acunetix.com/.
[2] Alexa top sites.
http://www.alexa.com/topsites.
[3] Malware domain blocklist.
http://www.malwaredomains.com.