Asian Journal of Engineering and Applied Technology (AJEAT)
Vol.2.No.1 2014pp 6-8
available at: www.goniv.com
Paper Received :05-03-2014
Paper Published:28-03-2014
Paper Reviewed by: 1. John Arhter 2. Hendry Goyal
Editor : Prof. P.Muthukumar

AN IMPROVED EFFICIENT SOLUTION FOR THE AUTOMATED DETECTION OF CLICK-JACKING ATTACKS

M.Jayakumar[1], Saveetha.D[2]
[1]Department of IT- Information Security and Cyber Forensics,
SRM University,
Kattankulathur, Chennai, T.N., INDIA

ABSTRACT

Click-Jacking is a web attack which is used to make a malicious page by masking the original content and creating a fake page to lure the user to click on the malicious page. So that the user clicks on the page without knowing the true content of the page, by this the attacker gets access to the victim. The attacker can get access to his Social webpage or also install malicious software into his system, without the knowledge of the victim. These web attacks cause much damage to the victim.

Keywords: Click-Jacking, Likejacking, UI redressing, Web Security, Click IDS, HTML I-FRAME, Browser Plug-In.

1. INTRODUCTION

A typical Click-Jacking [2] [3] attack uses two nested I-FRAMEs to crop and position an element from a target website. The inner I-FRAME contains the target page and must be large enough to display it in entirety, such that the element in which the user will click is visible without scrolling. The outer I-FRAME is much smaller and acts as a window into the page loaded in the inner I-FRAME. This type of attack is prevented by NoScript add-on. But Attackers have found another new way to do Click-Jacking via using the saved cached pages from the browser. By using "call history_back" function they retrieve the saved pages and load into a new pop-up window and start the attack by framing it into some non-malicious page.HTML5 is has many drawbacks compared to HTML 4. HTML5 gives more features which make it vulnerable to attackers.

.

2. EXISTING SYSTEM

To prevent Click-Jacking attack a NoScript add-on for Mozilla browser was created to mitigate the attacks. This add-on detects the overlapping clickable elements in the webpage. An extra module called ClearClick was also added to resolve this attack. This add-on also provides cross-site scripting attacks. It filters all the browser side downloadable contents like Java, Flash and Silverlight. This anti Click-Jacking feature is used to find the hidden CLICK-JACKING in the webpage. This add-on also features to find, when a single mouse click loads two events it alerts the user. So when the victim clicks the page on a single point and loads two events, it means a hidden frame webpage is behind and it also loads. By this way the attack is found.

3. MICHAEL ZALEWSKI METHOD

This method is based upon generating a pop-up window in browser and immediately loads the attacker controlled site and lure the victim to click on that page, shortly before the click the target page is loaded from the cache and it is immediately rendered. The click hits the target page and it navigates away or closes the pop up window.

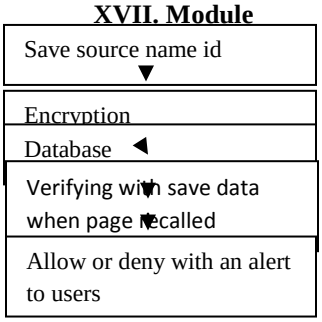
4. LIMITATIONS

The old methods are not 100% effective and the new Click-Jacking [2] [3] via History Navigation method is vulnerable to the existing tools. The cached pages in the browser are not encrypted, and it can be easily rendered by anyone into the new window.

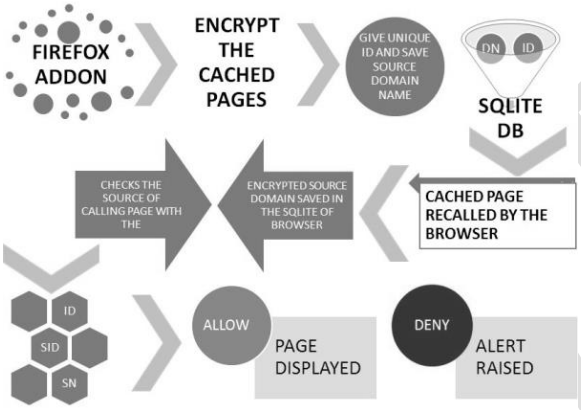
5. PROPOSED SYSTEM

The proposed system is to develop a browser add-on which keeps track of the Cached pages in the browser and saves the originated domain name in the database in an encrypted format. So whenever the cached page is recalled the add-on checks the source of the request, if the request matches to the true website or originas before the add-on allows the request to allow or deny according to the user option. This add-on will be supported by all latest browsers and can be installed easily.

XVII. Module



Data Flow Diagram



Likejacking

Likejacking [16] is an attack which is similar to Click-Jacking but it is used to steal the Like from social network Facebook. In common both the methods are same in a way. By this attack the victim gets more number of clicks to his post or page in Facebook or he can also manipulate the victim's status message to post as he likes.

6. CLICK-JACKING TOOL

For the reason that most web sites still have no implemented security mechanism against Click-Jacking, the company "Context Information Security Limited" published a program for Click-Jacking called "Click-Jacking Tool" [10] [15]. This tool was introduced by Paul Stone at the Black Hat Europe in 2010. It was primarily built due to the fact that it seems to be difficult to visualize Click-Jacking techniques in practice. It can be run in a web browser like Mozilla Firefox and allows for experimentation and trying out different techniques with point-and-click operations without a detailed knowledge of JavaScript or e.g. CSS. So it is possible to use point-and-click operations to select areas of a web page to be targeted just by using the mouse. It also supports techniques that are introduced in this paper that allow it to be easy used to try out the attacks. Functionalities like a "Visible mode" for replaying attacks to see how they work and the "hidden mode" for simulating a real Click-Jacking attack are helpful to get a better understanding of Click-Jacking.

Truncated screenshot of the "Click-Jacking Tool" displayed with the web browser Mozilla Firefox

A screenshot of the "Click-Jacking Tool" [10] [15]. First, the web browser opens the "google.com" web page, after that it types "nds.rub.de" into the search engine input field, and finally it clicks on "Google Search". Moreover, the condition for these steps to work is that the user must disable the automatic AJAX-based search function of "google.com". There are also things, or rather functionalities that the program does not offer. The first is that it is not possible to export a demo page to get malicious code to perform the attack directly. Secondly, the source code of the simulated Click-Jacking web page cannot be edited to see new effects concerning each attack variant by changing some parts of it. However, it is possible to check quickly whether a web page can be attacked or not. It would be also desirable to integrate SVG masking features so that one can, forexample, place the mask over an element with the mouse directly.

7. STORAGE

Local shared objects contain data stored by individual websites. With the default settings, the Flash Player does not seek the user's permission to store local shared objects on the hard disk. By default, a SWF application running in Flash Player may store up to 100 Kb of data to user's hard drive. If the application attempts to store more data than the allotted

default, the user is shown a dialog to allow or deny the request for more storage space. Adobe Flash Player does not allow 3rd-party local shared objects to be shared across domains. For example, a local shared object from "www.example.com" cannot be read by the domain "www.example2.com".However, the first party website can always pass data to a third party via some settings found in the dedicated XML file and passing the data in the request to the third party. Also, third party LSOs are allowed to store data by default. By default LSO[13] data is shared across browsers on the same machine. As an example.A visitor accesses a site using their Firefox browser, then views a page displaying a specific product, then closes the Firefox browser, the information about that product can be stored in the LSO [13].If that same visitor, using the same machine now opens an Internet Explorer browser and visits any page from the site viewed in Firefox, the site can read the LSO value(s) [13] in the Internet Explorer browser, and display dynamic content or otherwise target the visitor.This is unique from cookies which have directory isolated storage paths for saved cookies while LSOs use a common directory path for all browsers on a single machine.

8. CRITICISM

Many web based Flash games use LSO [13] files to store the user's personal game data, such as user preferences and actual game progress. Backing up files such as these requires a more technical understanding of software, and would be considered by most average users to be a difficult task. However, both browser updates and programs designed to remove unused files may delete this data. To help combat cheating, game developers may render LSO [13]files unusable if moved or uploaded from another location or backup. This has been criticized, however, as it may cause users to lose data despite backups.

Consequences

To date, there have been two kinds of widespread attacks in the wild: Tweet bomb [19] and Like-jacking. In both attacks, an attacker tricks victims to click on Twitter Tweet or Facebook Like buttons using hiding techniques, causing a link to the attacker's site to be reposted to the victim's friends and thus propagating the link virally. These attacks increase traffic to the attacker's site and harvest a large number of unwitting friends or followers.

9. CONCLUSION

Click-Jacking is a web attack which is used to make a malicious page by masking the original content and creating a fake page to lure the user to click on the malicious page. So that the user clicks on the page without knowing the true content of the page, by this the attacker gets access to the victim. The attacker can get access to his Social webpage or also install malicious software into his system, without the knowledge of the victim. These web attacks cause much damage to the victim.

REFERENCES

[1]. Security in the browser - Thomas Wadlow, Consultant VladGorelik, AVG Technologies.
[2]. Click-Jacking: Attacks and Defenses- Lin-Shung Huang, Alex Moshchuk, Helen J. Wang, Stuart Schechter, Collin Jackson.
[3]. Click-Jacking - OWASP, https://www.owasp.org/index.php/Click-Jacking
[4]. HTML5 Top 10 Threats - Stealth Attacks and Silent Exploits-By Shreeraj Shah, Founder & Director, Blueinfy Solutions.
[5]. HTTP Header Frame Options draft-ietf-websec-frame-options-00.
[6]. Busting Frame Busting: a Study of Click-Jacking Vulnerabilities on Popular Sites- Gustav Rydstedt, ElieBursztein, DanBoneh Collin Jackson.
[7]. Introduction to HTML 5- Brad Neuberg Developer Programs, Google.
[8]. Frame Busting:a Study of Click-Jacking Vulnerabilities on Popular Sites Gustav Rydstedt, ElieBursztein, Dan BonehStanford University, Collin Jackson Carnegie Mellon University.

Web links

[9]. Colin Ihrig (http://www.sitepoint.com/author/cjihrig/).

[10]. File API- http://www.w3.org/TR/2013/WD-FileAPI -20130912/.

[11]. http://msdn.microsoft.com/en-us/library/windows/apps/hh464959.aspx.
[12]. Local shared object - Wikipedia, the free encyclopedia
[13]. http://en.wikipedia.org/wiki/Local_shared_object
[14]. R.Hansen.Click-Jacking details. http://ha.ckers.org/blog/20081007/Click-Jacking-details/
[15]. Mozilla Foundation. https://bugzilla.mozilla.org/show_bug.cgi?id=154957
[16]. http://en.wikipedia.org/wiki/Click-Jacking.
[17].http://help.adobe.com/en_US/FlashPlayer/LSM/WS6aa5ec234ff3f285139dc56112e3786b68c-7fff.html
[18].http://noscript.net/
[19].http://en.wikipedia.org/wiki/Twitter_bomb
[20].https://www.schneier.com/blog/archives/2009/08/security_vs_usa.html