Research and Design on Web Application Vulnerability Scanning Service

Wu Qianqian
Department of Information and Communication Engineering
North China Electric Power University
Beijing, China
beata19900118@163.com

Liu Xiangjun
Department of Information and Communication Engineering
North China Electric Power University
Beijing, China
lxjun@ncepu.edu.cn

Abstract **Web application has got a remarkable change in the past few years, many new technologies are reshaping the pattern of Web applications. Since many manufacturers' promotion on HTML5 technology, more and more websites are using HTML5 gradually. The new technology provides users with a variety of Internet applications, but introduces new security problems at the same time. Currently, most Web application scanners can not detect the security problems with HTML5 features, which make HTML5 security issues become blind spots in security vulnerability scanning process. The paper focuses on a research among the existing Web application scanners firstly. Then we selected W3af(Web Application Attack and Audit Framework) as a basic platform for transformation, and by customizing scanning modules and scripts, we designed a Web application security scanning service. The practical scan results show that it can not only detect the Clickjacking vulnerabilities brought by HTML5, but also provide efficient Web application security scanning and evaluation services for the websites.**

Keywords: Clickjacking; HTML5; W3af; Web application vulnerability scanning

I. INTRODUCTION

Web application technology is one of the fastest growing technologies in the past 20 years. With the continuous development of Web application technology, Web application security has attracted researchers' attention.

Although a variety of Web application scanners can not only greatly facilitate penetration testers to work in actual penetration tests , but also assess a certain site's safety. However, the current Web application scanners can not completely replace the manual way of penetration to do comprehensive assessment in websites, mainly because of the following several significant issues:

1) There exists false positive results of scanning, which requires professionals to authenticate.
2) Web applications scanning are generally time-consuming and inefficient. Especially some open source scanners are inadequate to determine the target systems.
3) Open source Web application scanners do not update themselves in time, so they lack of approaches to do security testing on new technologies in Web applications. (e.g. Clickjacking in HTML5)
4) Limited degree of automation, especially in the use of open source tools are unfriendly, they all need certain operators who have a background knowledges in information security to do the configuration.

Based on this, the thesis will give a reasonable solution for the detection, and improve the efficiency of Web application security vulnerability scanning especially for Clickjacking in HTML5.

II. RESEARCH ON OPEN SOURCE WEB APPLICATION VULNERABILITY SCANNER

Open source Web scanners appeared frequently in recent years. According to incomplete statistics, the current open source and free version Web scanners have reached more than 100 kinds. These Web application vulnerability scanners are quite different in terms of functionality and technology. The following table lists some of the more well-known open source vulnerability scanners.

TABLE I: THE FAMOUS OPEN SOURCE WEB VULNERABILITY SCANNER

Name	Function	Type
Arachni	Integrated scanner	automatic
Grendel-Scan	Integrated scanner	automatic
Skipfish	Integrated scanner	automatic
ZAP proxy	Integrated scanner	Semi-automatic
Sqlmap	SQL vulnerability scanner	automatic
W3af	Integrated scanner	automatic

978-1-4799-3279-5 /14/$31.00 ©2014 IEEE

Xsser	XSS vulnerability scanner	automatic

For the integrated scanner, currently can be divided into two types: automated and semi-automated. Automated scanner is simple to configure, and it only needs to enter the URL of the target site, so the scanner can crawl every page of the site, then scan automatically and give a result. This tool has certain requirements on web crawling. For semi-automatic scanner, generally it needs to set the proxy in the browser, the user's requests for each target site will be analyzed through the scanner, also you can choose whether to be crawling on the web page or not. And each time the requests and responses of the browser can capture more details , it requires the user to determine whether to request further explorations. The scanner requires the user to have more systematically knowledge on the Web security system, so semi-automated scanners are much more suitable for professionals to use.

Due to the differences in the technical realizations of the scanners, the scanning accuracy of various vulnerabilities vary widely. In all of the open source Web application scanners, W3af is highly-modular, and it is easy to expand. So we choose W3af as the basic platform to transform so that it can provide a Web application security vulnerability scanning service.

III. WEB APPLICATION ATTACK AND AUDIT FRAMEWORK

W3af (Web application attack audit framework) is an open source project created by Argentine Andres Riancho. And the goal is to become a Web application attacks and statistical platform. Currently, W3af is divided into two main parts: the core modules and plug-ins section. The former is responsible for scheduling and using of the plug-ins. The later is responsible for finding and attacking Web security vulnerabilities. According to different functions of the plug-in part, it is divided into eight modules, including : discovery module (discovery), audit module (audit), search module (grep), attack module (attack), the output module (output), modify module (mangle), escape module (evasion), break module (bruteforce). The relations between them are shown in Fig. 1 :

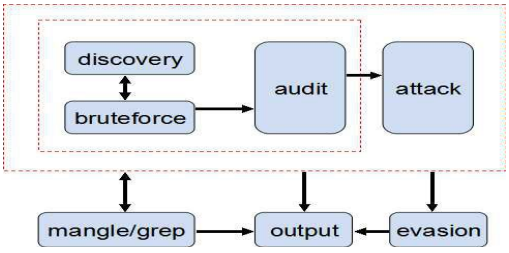

Fig. 1. Inner Structure of W3af

However, either the command line or the UI methods of the W3af client requires a certain understanding of Web application security technology. The configuration is complicated, too. The default scan options not only can not meet the needs of remote security assessments, but also because of so many audit modules, it takes a long time for scanning. Also we should recognize that W3af is not just a Web application assessment tool, it also has a certain penetration testing auxiliary capacity, so the function of many modules can generate a larger role in penetration testing, but they are not significant for security assessment.

In the meantime, many modules in W3af only work for individual languages or frameworks. In fact, most sites only use a certain kind of Web service, framework and development language, too many unrelated configurations in some modules will not affect the scan efficiency, but may interfere with the scan results. So the modules in W3af need to be sorted and classified.

IV. RESEARCH ON CLICKJACKING IN HTML5

In recent years, the major Internet browser vendors have strongly supported the development and promotion of HTML5 protocol. The good interactive performance, simple-understanding code, and powerful features, making HTML5 become the next generation of Web application technology standards. However, the introduction of new technology security issues can not be ignored.

One of the popular ways to attack is Clickjacking, also known as Interface Camouflage Hijacking. It means an attacker use a transparent cover in a normal web page to entice users to click on a button in a normal web page, the user unknowingly click the button on that transparent page at the same time, thereby triggering malicious code attacks which were preset. As the popular of social network, the effect of such attacks becomes more obvious .

Because of the introduction of a new label and a new Cascading Style Sheets (CSS-level3), making it easier to attack through clickjacking in HTML5. Since the introduction of sandbox attribute in iframe in HTML5, it allows non-homologous domain L to be loaded in iframe. If it is not strictly set while allowing scripts to execute, there will be a possibility of cross-site scripting attacks.

In order to solve the problem, many websites are using a frame-busting technique : using some Java scripts to check whether there are cover pages in "form". However, the sandbox attribute prohibits Java scripts to run, resulting in using a sandbox can be bypassed by a frame-busting checks:

In the past years, there are a lot of excellent commercial or open source Web application scanners. Through our research, we choose a higher rank and widely acclaimed scanners to check for whether these security tools are supportive of Clickjacking attacks in Table Ⅱ.

TABLE Ⅱ: THE SUPPORT OF CLICKJACKING IN DIFFERENT SCANNERS

Name	Type	Detection for Clickjacking
IBM AppScan	Foreign commercial	no
Sandcat	Foreign commercial	no
Acunetix	Foreign commercial	no
Netsparker	Foreign commercial	no
Burp Suite Pro	Foreign commercial	yes

W3AF	Open source	underdevelopment
OWASP ZAP	Open source	no
SkipFish	Open source	no
Nikto	Open source	no
WebRavor	Domestic commercial	no
MatriXay	Domestic commercial	no

As can be seen from TABLE Ⅱ , almost all open source tools can not detect Clickjacking security issues. Domestic Web application scanning tools are also lack of detection, which makes Clickjacking becoming a blind spot without due attention.

V. W3AF INTEGRATED WEB SCANNING SERVICE DESIGN

As a comprehensive automated scanning tool, W3af is equipped with scanning capabilities for almost all mainstream Web threats, while supporting a variety of modules to be loaded for attacking. The web-spider in it also supports proxy authentication function, you can input in advance in order to do a back-page scanning. There are many advantages even better than many commercial tools, too. But it also has some disadvantages : not suitable for people who are not familiar with Web security knowledge; unfriendly user interface; difficult in command line configuration; complex configure in graphical interface; untimely to find vulnerabilities in new technologies for Web applications(such as HTML5). According to the above shortcomings, combined with the specific circumstances of university services, this part is committed to develop detection module related to Clickjacking, and support for the latest vulnerabilities to find.

A. Design module for HTML5 security vulnerability scanning

In W3af, the audit module provides a detection for all kinds of vulnerabilities, such as injection vulnerabilities, cross-site scripting attacks, etc. These audit modules are all stored in the form of Python scripts under the directory W3af/plugins/audit. This modular approach brings convenience to add new detection modules.

Clickjacking security issues become more serious after the introduction of the new technologies in HTML5. Considering Clickjacking is strengthened in hand touch devices, it is necessary to check whether the server has set up protective measures in terms of Clickjacking.

The best way to prevent Clickjacking currently is to deploy X-FRAME-OPTION parameter in HTTP header. This option allows the developer to specify which pages can not be added with frame boxes. The X-FRAME-OPTION parameter has two operating modes: first, deny all users to add frames on the current page; second , homologous users can add frames on the page.

The method of detecting Clickjacking attack in w3af is to determine whether the response of HTTP packets support X-FRAME-OPTIONS option. if supports, it means it can withstand Clickjacking attack, otherwise there exists the risk of Clickjacking attacks. Achieving detection on Clickjacking is as the following:

- Calculate HTTP headers in all responses
- Calculate whether HTTP headers has set the X-FRAME-OPTION option
- If all of the URL have deployed X-FRAME-OPTION, then there is no vulnerability
- If the X-FRAME-OPTION deployment option is not present, then there is Clickjacking vulnerability
- If the number of recorded HTTP response headers are not equal with the number of headers which are not configured with X-FRAME-OPTION, it shows that part of the pages have cross-site clicking vulnerabilities.

B. Customize script in W3af for scanning

Either the command line client or the UI method client of the W3af requires a certain understanding of Web application security technology. The configuration is complicated, too. The default scan options not only can not meet the needs of remote security assessments, but also because of so many audit modules, it takes a long time for scanning. Also we should recognize that W3af is not just a Web application assessment tool, it also has a certain penetration testing auxiliary capacity, so the function of many modules can generate a larger role in penetration testing, but they are not significant for security assessment.In the mean time, many modules in W3af only work for individual languages or frameworks. In fact, most sites only use a certain kind of Web services, frameworks and development languages. Too many unrelated configurations in some modules will not only affect the scan efficiency, and may interfere with the scan results, so the modules in W3af need to be sorted and classified.

Through the research on W3af, Web Application Framework types and characteristics of Web applications, we make a custom script from the four aspects: operating system, Web framework, detection module, and output format. The script is convenient for users to choose for Web security assessment. TABLE Ⅲ only lists the grep module in the safety assessment process for Clickjacking detection.

TABLE Ⅲ: W3af script configuration module program

W3af module	options	remark
Operating system	Windows Unix Unknow	Default is unknown
Web framework	Unknow, PHP, Asp Asp.net, Java Jsp, Ruby, Perl	Default is unknown

Grep module	Clickjacking	Detect whether the target can defense Clickjacking
Output module	HtmlFile	Configure the output format

Through the above described combination of several modules for different Web frame and system, there are different scripts. By comparing the default scan options with the customed script can we describe the advantages of the custom script.

C. Practical test on a website

After a brief configuration on our testing environment, we selected the relevant modules to configure scripts in w3af. The information can be show as follows:

```
plugins
    grep error500, clickJacking
    back
plugins
    output htmlFile, console
    output config htmlFile
        set fileName test01.html
        set verbose False
        back
    back
plugins
    discovery serverHeader, allowedMethods
back
```

After scanning the target : http://202.204.65.103, we got the following results:

Vulnerability	tcp/80	An unidentified web application error(HTTP response code 500) was found at:"http://www.ncepu.edu.cn/SiteFiles/inner/dynamic/output.aspx". Enable all plugins and try again, if the vulnerability still is not identified, please verify manually and report it to the w3af developers. This vulnerability was found in the request with id 51924. URL:http://www.ncepu.edu.cn/SiteFiles/inner/dynamic/output.aspx Severity : Medium
Vulnerability	tcp/80	An unidentified web application error(HTTP response code 500) was found at:"http://www.ncepu.edu.cn/SiteFiles/inner/dynamic/output.aspx". Enable all plugins and try again, if the vulnerability still is not identified, please verify manually and report it to the w3af developers. This vulnerability was found in the request with id 14693.

Fig. 2. Part of the scan result for a certain website

As can be seen from the test results, through human intervention to verify the vulnerabilities, the scan confirmed the information about leakage, robots.txt, and Clickjacking vulnerabilities. Practice has proved that through the packaging and transformation, W3af is able to detect the target site' s vulnerabilities in the actual operation. At the same time, it has increased dramatically in the detection time than the original tool.

VI. CONCLUSION

On the basis of the research on existing Web application security technologies, the paper firstly discusses the open source Web application vulnerability scanners. By comparing we selected W3af for Web interface package, and by designing audit module, we increased the Clickjacking vulnerability scan especially in HTML5. Finally, we made a custom script and used it in actual scanning test, the result shows the relevant vulnerabilities can be detected and the scan efficiency can be greatly improved without affecting the scan results.

With the development of Web application technology, Web application security issues will increasingly occur. Traditional security vulnerabilities still require manual intervention to detect. Web application scanners can not completely replace the penetration testers in Web applications vulnerability scanning. The researchers still need to keep track of new security threats, and add check technologies to Web application scanners through effective mechanisms.

REFERENCES

[1] W3af. http://w3af.sourceforge.net/.

[2] A. Doupe, M. Cova, G. Vigna. Why Johnny Can't Pentest: An Analysis of Black-box Web Vulnerability Scanners. Proceedings of the Conference on Detection of Intrusions and Malware and Vulnerability Assessment (DIMVA) Bonn, Germany July 2010.

[3] Web Application Vulnerability Scanner Evaluation Project(Vulnerability Scanner Evaluation Project) [2012] http://code.google.com/p/wavsep/

[4] Michael Schmidt, Thomas Röthlisberger ,HTML5 web , security[EB] Compass Security AG , December 6th, 2011

[5] Gordon Lyon.Top 125 Network Security Tools [EB][2011] .http:// sectools.org/

[6] Shay-Chen The Web Application Vulnerability Scanner Evaluation Project [EB][2012] http://www.sectoolmarket.com/