Detecting and Exploiting XSS with Xenotix XSS Exploit Framework

Ajin Abraham

ajin25@gmail.com

keralacyberforce.in

Introduction

Cross Site Scripting or XSS vulnerabilities have been reported and exploited since 1990s. XSS got listed as the top 2nd Vulnerability in the OWASP 2010 Web application Vulnerabilities list.

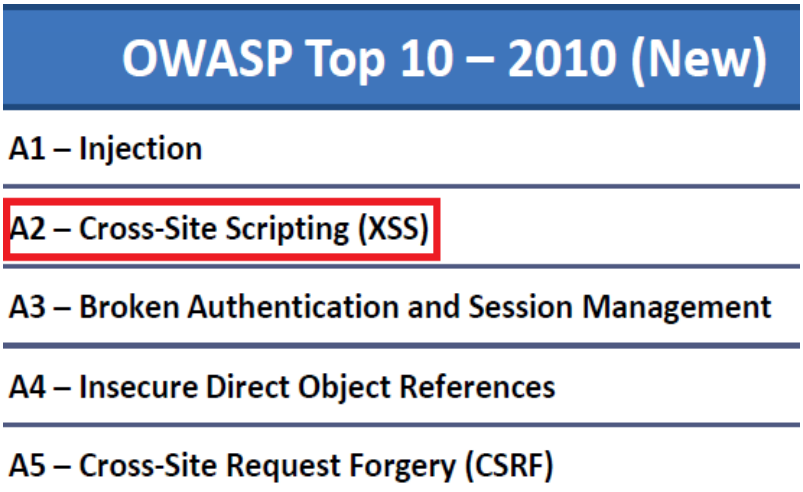

Figure 1: Top 10 Web Application Vulnerabilities OWASP

Cross-site scripting (XSS) is a type of security vulnerability typically found in web applications which allows the attackers to inject client-side script into web pages viewed by other users. The execution of the injected code takes place at client side. A cross site scripting vulnerability can be used by the attacker to bypass the Same Origin Policy (SOP). In the past, the potentials of XSS vulnerability were not known. XSS was mainly used for stealing cookies and for temporary or permanent defacements and was not considered as high risk vulnerability. But later XSS tunneling and Payload delivering showed us the potential of XSS Vulnerability. Most of the large websites like Google, Facebook, Twitter, Microsoft, and Amazon etc. even now suffers from XSS bugs. That's a brief introduction about XSS.

Threats due to XSS

XSS Tunneling: With XSS Tunnel a hacker will obtain the traffic between the victim and a webserver.

Client side code injection: A hacker can inject malicious codes and execute them at client side.

DOS: A hacker can perform DOS against a remote server or against the client itself.

Cookie Stealing: A hacker can obtain the session cookies or tokens of a victim.

Malware Spreading: A hacker can spread malwares with a website which is vulnerable to XSS.

Phishing: A hacker can embed or redirect to a fake page of the website to get the login credentials of the victim.

Defacing: Temporary or permanent defacement of web application is possible.

Need for a new Tool

Many tools are available for detecting XSS vulnerabilities in web applications. But most of these are not so easier to use or you should specify XSS payloads manually. So I thought of the possibility of a new user friendly tool with a payload list to test against XSS in a web application. After a 5 months research, I built a XSS payload database of over 350+ XSS payloads and implemented a tool in VB.NET and that is Xenotix XSS Exploit Framework.

What is Xenotix XSS Exploit Framework?

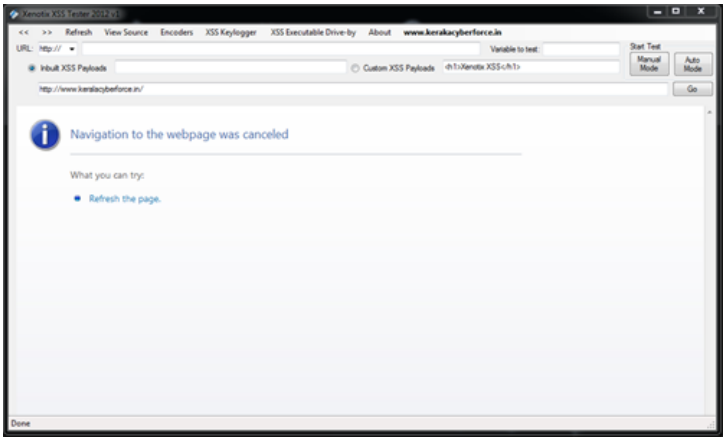

Figure 2: Xenotix XSS Exploit Framework

Xenotix XSS Exploit Framework is a penetration testing tool to detect and exploit XSS vulnerabilities in Web Applications. This tool can inject codes into a webpage which are

vulnerable to XSS. It is basically a payload list based XSS Scanner. It provides a penetration tester the ability to test all the possible XSS payloads available in the payload list against a web application with ease. The tool supports both manual mode and automated time sharing based test modes. It includes a XSS encoder, a victim side keystroke logger, and an Executable Drive-by downloader.

Features of Xenotix XSS Exploit Framework

The features of Xenotix XSS Exploit Framework are

- Built in XSS Payloads
- XSS Key logger
- XSS Executable Drive-by downloader
- Automatic XSS Testing
- XSS Encoder

Built in Payload List

It is having an inbuilt XSS payload list of above 350+ XSS payloads. It includes HTML5 compactable XSS injection payloads. Most of the XSS filters are implemented using *String Replace filter*, *htmlentities filter* and *htmlspecialcharacters filter*. Most of these weakly designed filters can be bypassed by specific XSS payloads present in the inbuilt payload list.

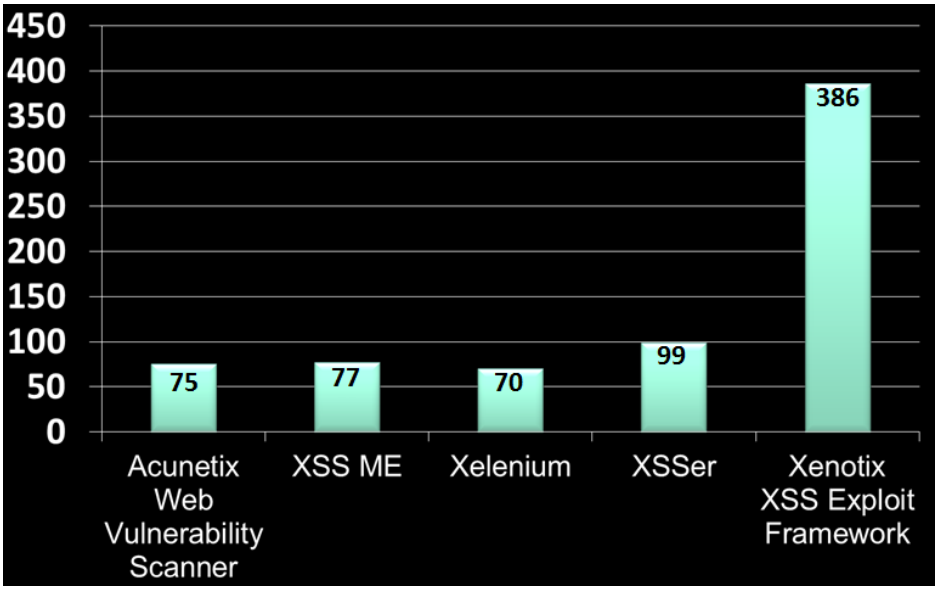

Figure 3: XSS Payload count in different Vulnerability Scanners

The above chart shows the number of XSS Payloads in different XSS Scanning tools available in market. So it's clear that Xenotix XSS Exploit Framework got the world's second largest XSS Payload list.

XSS Key logger

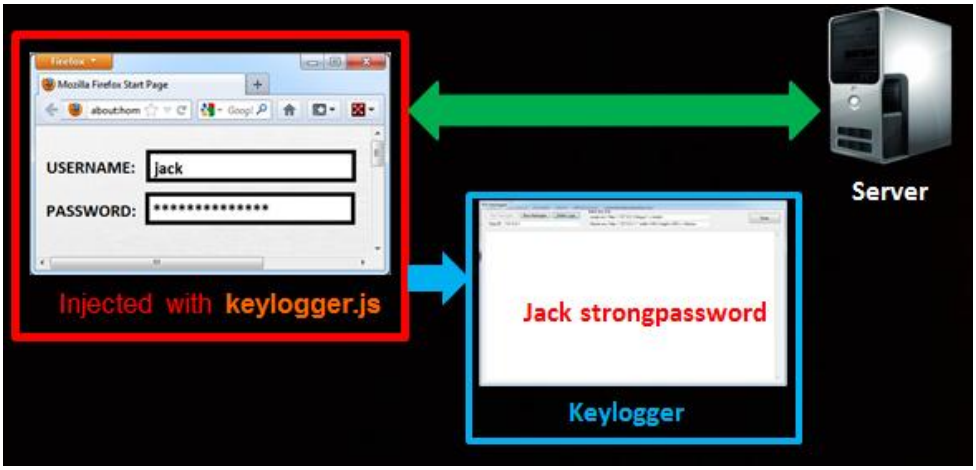

Figure 4: XSS Keylogger Working

The tool includes an inbuilt victim side Key logger which is implemented using JavaScript and PHP. PHP is served with the help of a portable PHP server named QuickPHP by Zach Saw. A JavaScript file is injected into the web application vulnerable to XSS and is presented to the victim. The script captures the keystrokes made by the victim and send to a PHP file which further write down the logs into a text file.

XSS Executable Drive-by Downloader

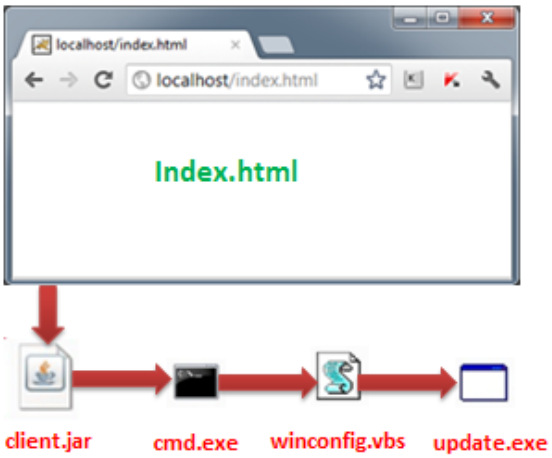

Figure 5: Executable Drive-by Downloader Working

Java Drive-by download can be implemented with Xenotix XSS Exploit Framework. It allows the attacker to download and run a malicious executable file on the victim's system without his knowledge and permission. You have to specify the URL for the malicious executable and then embed the drive-by implemented webpage into a XSS vulnerable page and serve your victim. When the victim view the injected page, the java applet client.jar will access the command

prompt and with the help of echo command, write down some scripts to a Visual basic script file named winconfig.vbs in the temp directory(%temp%) and then the cmd.exe will start winconfig.vbs. The winconfig.vbs will download the malicious executable specified by you in the URL to temp directory and rename it as update.exe and finally it will execute update.exe. The downloading and executing of the malicious executable happened without the knowledge and permission of the victim.

Automatic XSS Testing

The tool is having an automatic test mode based on a time interval. You have to specify the time interval according to the time taken by a webpage to load which depends on your bandwidth. It will test all the payloads one by one after the specified time interval. With this feature automated XSS testing can be done. You don't have to check all the 350+ payloads manually.

XSS Encoder

The inbuilt Encoder will allow encoding into different forms to bypass various filters and Web Application Firewalls. The encoder supports Base64 Encoding, URL Encoding, HEX Encoding, HTML Characters Conversion, Character Code Conversion and IP to Dword, Hex and Octal conversions.

Testing a website with Xenotix XSS Exploit Framework

To test a website URL, say **http://www.site.com/search.php?id=1&term=about**

You suspect that the variable '**term**' is vulnerable to XSS.

Figure 6: Testing a Website with Xenotix XSS Exploit Framework

For testing against XSS in Xenotix XSS Exploit Framework you should specify the protocol, which is http or https. Then give the website URL other than the suspected variable in the field after

the protocol and specify the suspected variable in the Variable to test field. Now select between Inbuilt XSS Payloads or Custom XSS payloads. You can select between Manual Mode and Auto Mode to start testing.

Features for the Next Build

Current version of XSS Exploit Framework is based on Internet Explorer's webpage rendering engine. Since XSS got slightly different behavior in different Web Browsers, the support for the Gecko (Used by Mozilla Firefox) and Webkit (used by Chrome, Opera, and Safari) Rendering engines will be added up in the next build. The support for XSS in POST Parameter will be included in the next build. XSS Proxy to tunnel the victim-server traffic will be added in future builds. Automatic detection of parameters or variables vulnerable against XSS and DOM Based XSS detection will be added up in next build.

Conclusion

XSS in popular website is a high security threat. Xenotix XSS Exploit Framework can be used by Security Analysts to perform Penetration test on Web Applications against XSS vulnerability. Google Vulnerability Reward Program, Facebook Bounty etc. are there. So go for XSS hunting and grab your bounty. ☺

References

Papers

- Our Favorite XSS Filters/IDS and how to Attack Them - Eduardo Vela and David Lindsay.
- Blackbox Reversing of XSS Filters - Alexander Sotirov.
- Advanced Cross-Site-Scripting with Real-time Remote Attacker Control - Anton Rager
- Bypass XSS filters - k3nz0
- XSS for Fun and Profit - Lord Epsylon
- Bypassing Web Application Firewalls (WAFs) - Ing. Pavol Lupták
- Abusing Internet Explorer 8's XSS Filters –Eduardo Vela Nava, David Lindsay

Websites

- OWASP's Cross-site Scripting (XSS)
  https://www.owasp.org/index.php/Cross-site_Scripting_(XSS)
- CGISecurity's Cross Site Scripting FAQ
  http://www.cgisecurity.com/xss-faq.html#
- Gunter Ollmann's XSS paper
  http://www.technicalinfo.net/papers/CSS.html

- PeterW's Cross Site Request Forgery (CSRF) Concept
  http://www.securityfocus.com/archive/1/191390
- CERT info on XSS
  http://www.cert.org/advisories/CA-2000-02.html
- Remote Scripting with IFRAMEs
  http://developer.apple.com/internet/webcontent/iframe.html
- Cross Site Scripting - XSS - The Underestimated Exploit
  http://www.acunetix.com/websitesecurity/cross-site-scripting.htm