

Cross-Site Scripting

Getting Developers to Take XSS Seriously

Use Social Engineering to Enhance Your Vulnerability Reporting



XSS

Contact Information

Jason Pubal

Website

www.intellavis.com/blog

E-mail

jpubal@gmail.com

Social

<http://www.linkedin.com/in/pubal>

<http://www.twitter.com/pubal>

Cross-Site Scripting

Outline

What is XSS?

XSS History

Detecting XSS

Preventing XSS

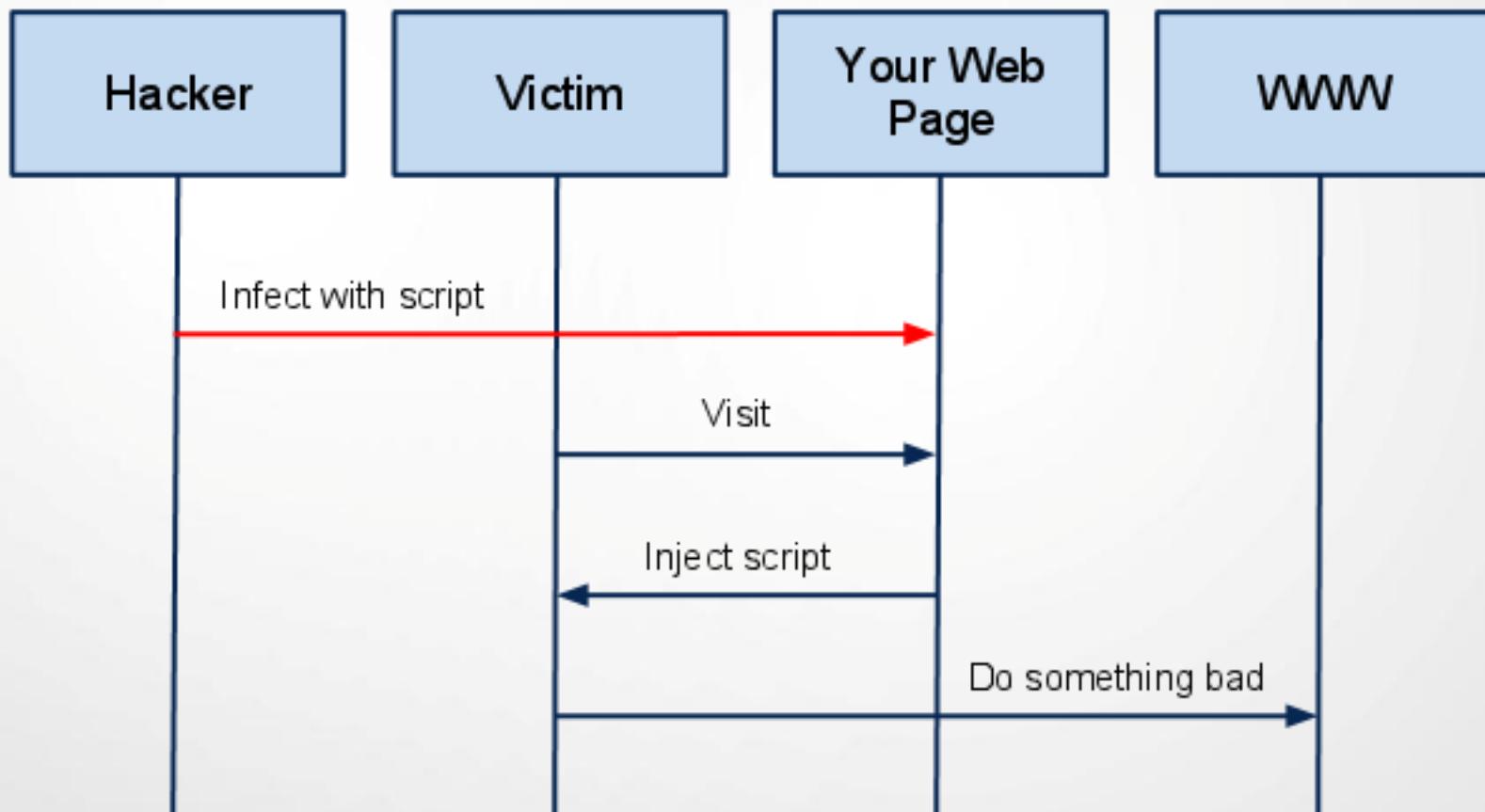
Reporting Tricks

Cross-Site Scripting

Cross Site Scripting (XSS) is an attack against the user of a website. It is a technique that forces a website to display malicious code, which then executes in the user's web browser. The attacker uses a vulnerable website to send malicious code to another end user of the site. The vulnerability arises when the website takes data in some way from a user and dynamically includes it in a web page without first validating that data.

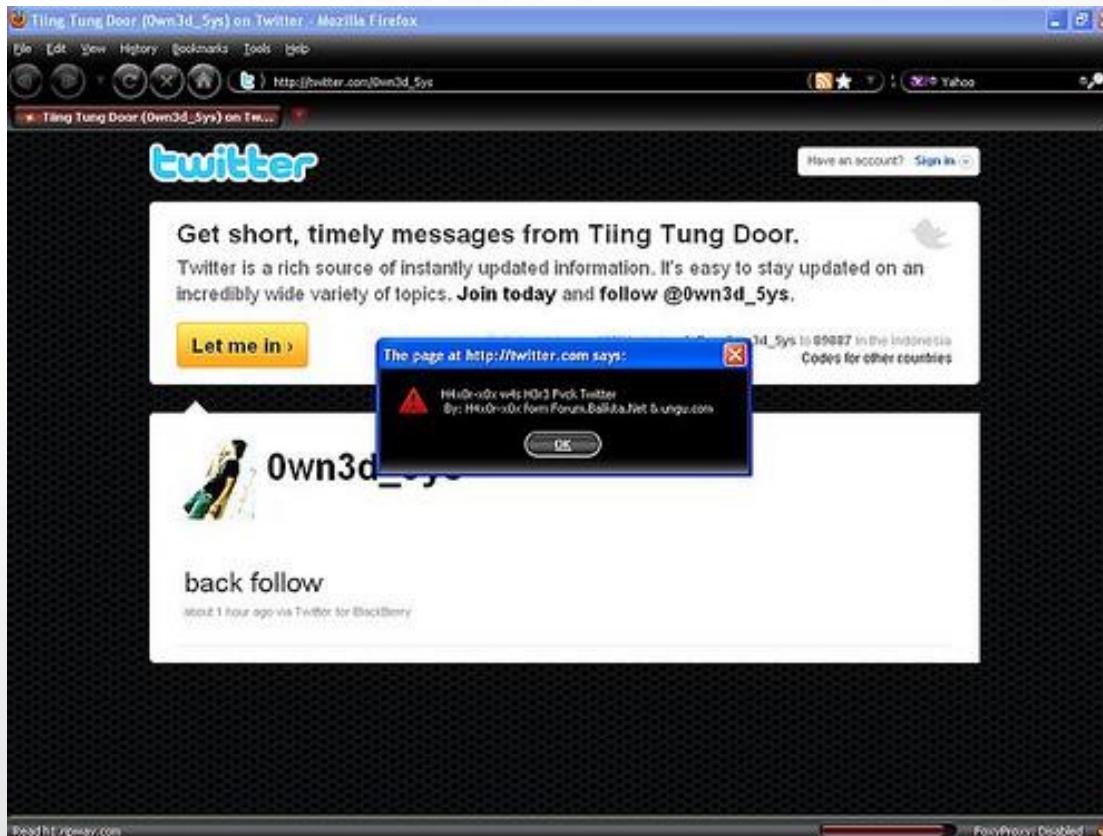
- account hijacking
- rewrite portions of the page
- log keystrokes
- steal browser information
- Steal client machine data
- attack the user's network

XSS



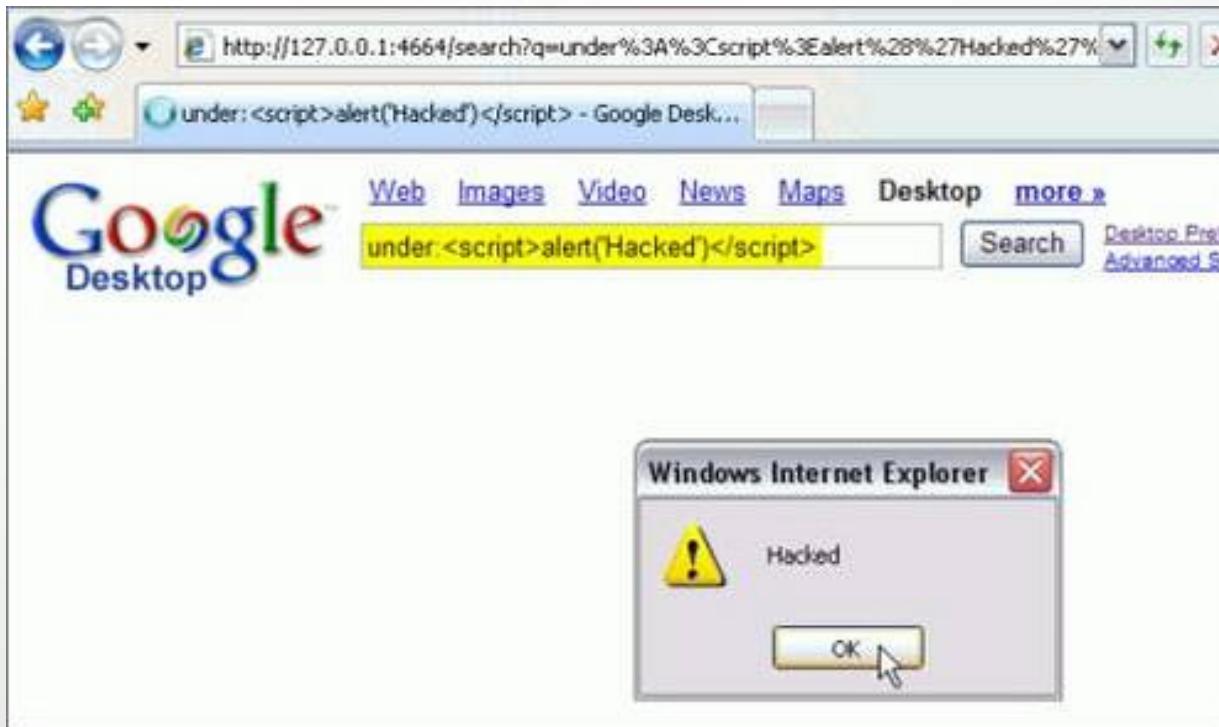
.....

Persistent Cross-Site Scripting



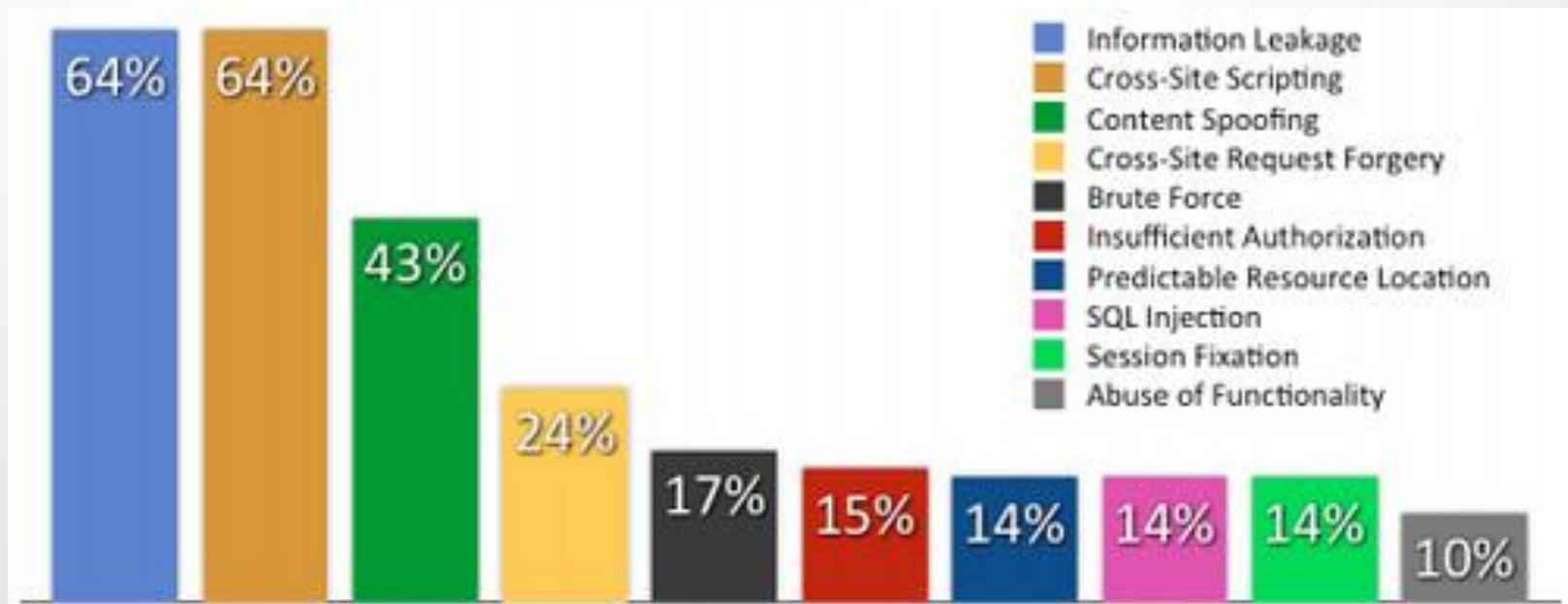
XSS

Reflected Cross-Site Scripting



Websites with Cross-Site Scripting

WhiteHat Website Security Statistic Report, Winter 2011



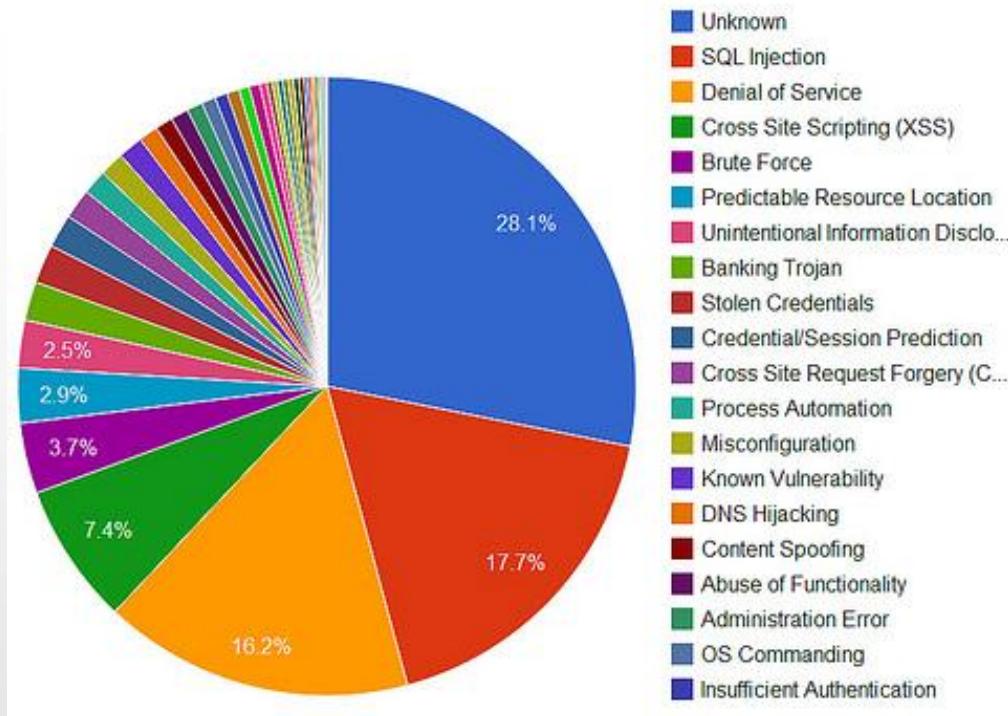
.....

XSS

Attacks Using Cross-Site Scripting

Web Hacking Incident Database

Top Attack Methods (All Entries)



Real World Examples

Hacker Redirects Barack Obama's site to hillaryclinton.com

During the 2008 democratic primaries, XSS in Obama's website was exploited to redirect visitors to Hillary Clinton's website. Users who went to Obama's community blog were instead taken to www.hillaryclinton.com.

Apache.org hit by targeted XSS attack, passwords compromised

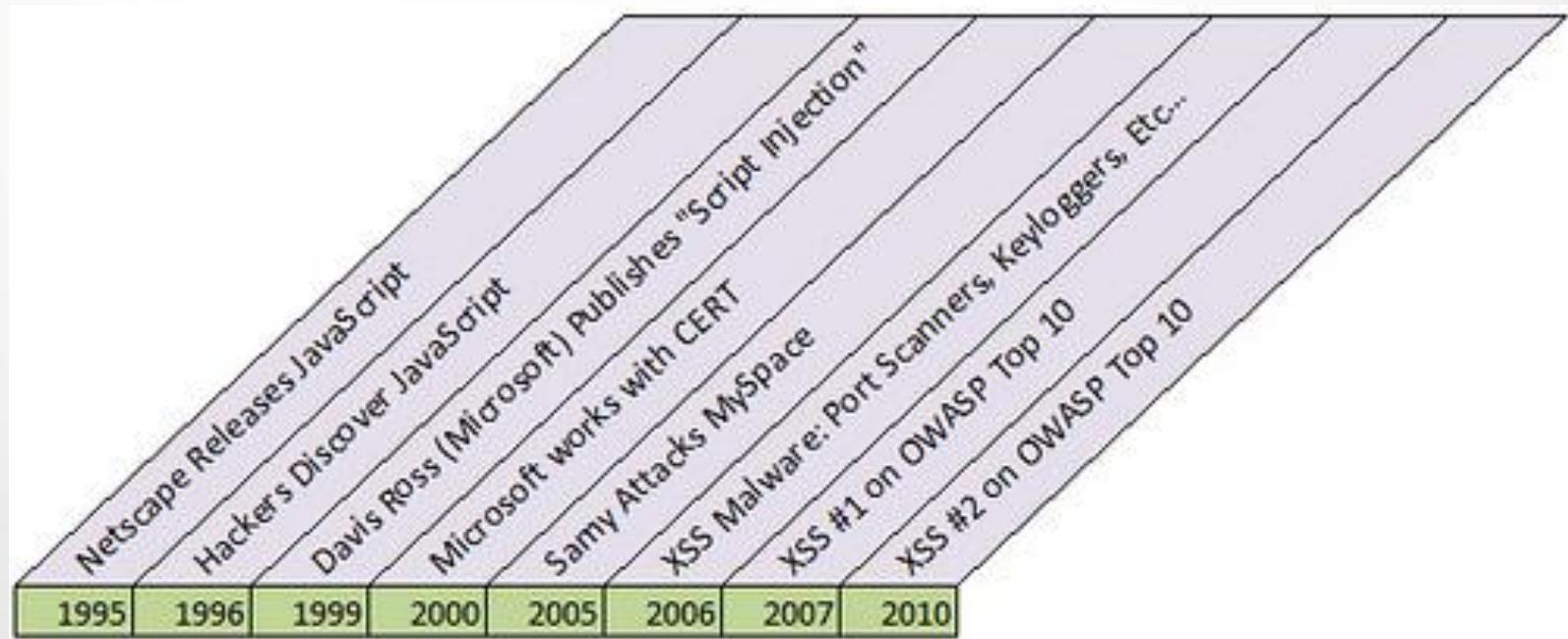
A targeted attack against JIRA admins used XSS to steal administrative cookies. Using those privileges, they installed backdoors and scripts to collect passwords at login. Thanks to people's tendency to use the same password on several websites and applications, the attacker was able to use those credentials get root access to other servers.

New XSS Facebook Worm Allows Automatic Wall Posts

An XSS in the Facebook's mobile API allowed a maliciously prepared iframe element containing JavaScript to post to user's walls.

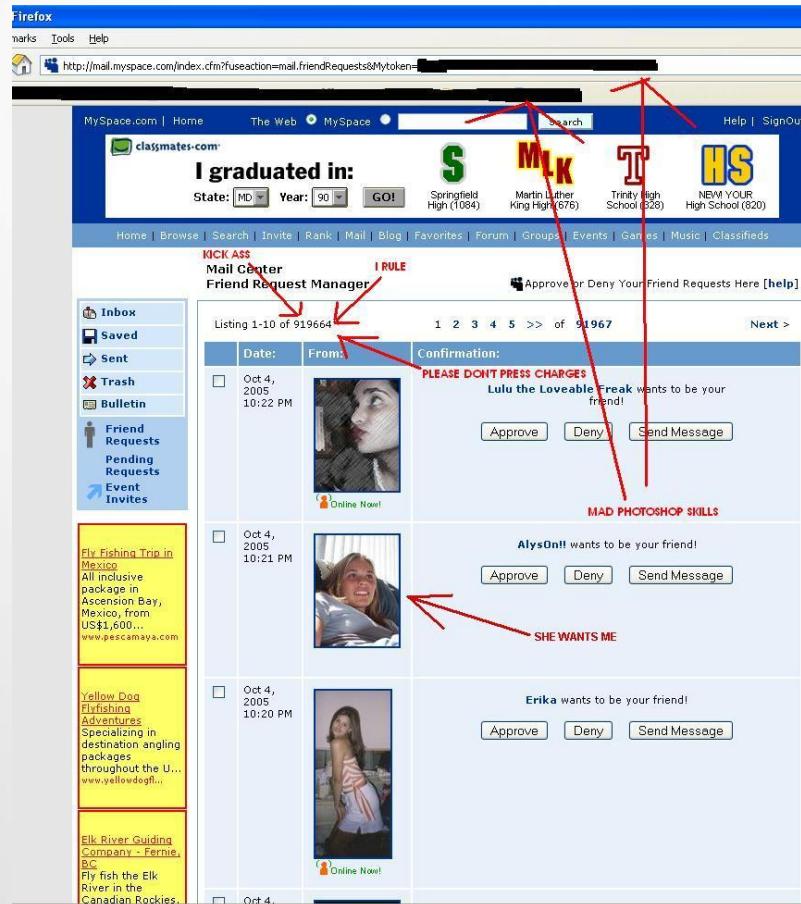
.....

History of Cross-Site Scripting



XSS

Samy



A screenshot of a Firefox browser window showing a MySpace friend request inbox. The URL in the address bar is [http://mail.myspace.com/index.cfm?useaction=mail.friendRequests&Mytoken=\[REDACTED\]](http://mail.myspace.com/index.cfm?useaction=mail.friendRequests&Mytoken=[REDACTED]). The page title is "MySpace.com | Home". The inbox lists 919664 friend requests, with the first few shown in detail:

- Listing 1-10 of 919664**
- From:** PLEASE DON'T PRESS CHARGES (Oct 4, 2005 10:22 PM)
Confirmation: Lulu the Loveable Freak wants to be your friend!
Buttons: Approve, Deny, Send Message
Text: MAD PHOTOSHOP SKILLS
- From:** AlysonH! (Oct 4, 2005 10:21 PM)
Confirmation: AlysonH! wants to be your friend!
Buttons: Approve, Deny, Send Message
Text: SHE WANTS ME
- From:** Erika (Oct 4, 2005 10:20 PM)
Confirmation: Erika wants to be your friend!
Buttons: Approve, Deny, Send Message

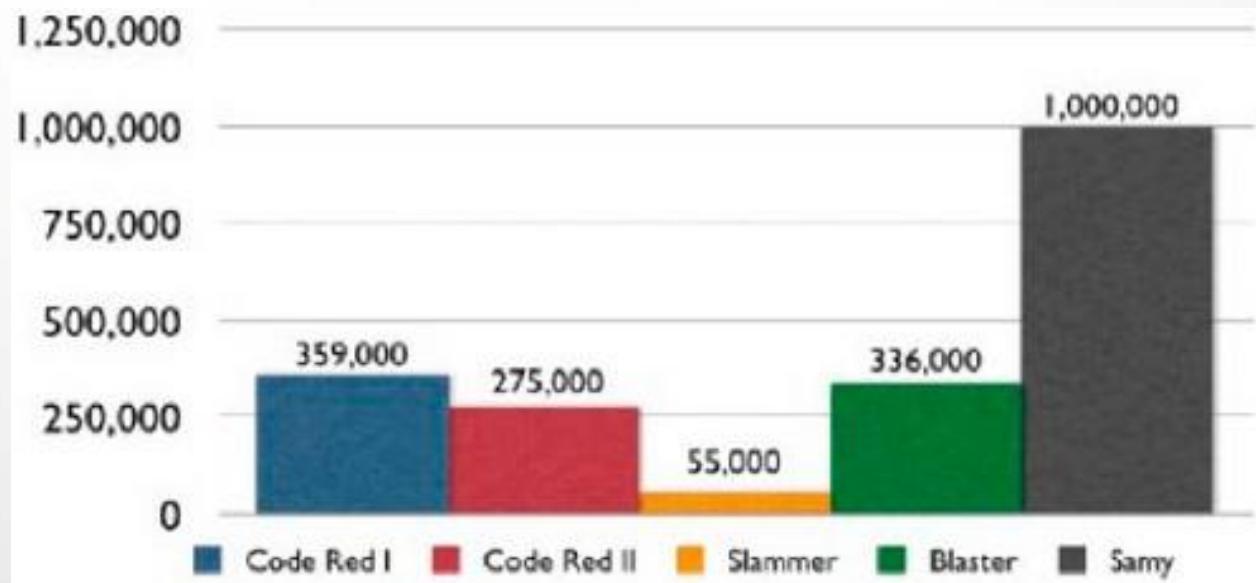
The sidebar on the left shows a navigation menu with links: Inbox, Saved, Sent, Trash, Bulletin, Friend Requests, Pending Requests, and Event Invites. There are also three advertisement boxes: "Fly Fishing Trip in Mexico", "Yellow Dog Flyfishing Adventures", and "Elk River Guiding Company - Fernie, BC".

"I'm sorry MySpace and FOX. I love you guys, all the great things MySpace provides, and all the great shows FOX has, my favorite being Nip/Tuck. Oh wait, Nip/Tuck is FX? My bad, but FOX, I'm sure you still have some good stuff. But maybe you should start picking up Nip/Tuck reruns? Just a thought. I'm kidding! Please don't sue me."

XSS

Samy

Fastest Spreading Worm in History



JavaScript Malware

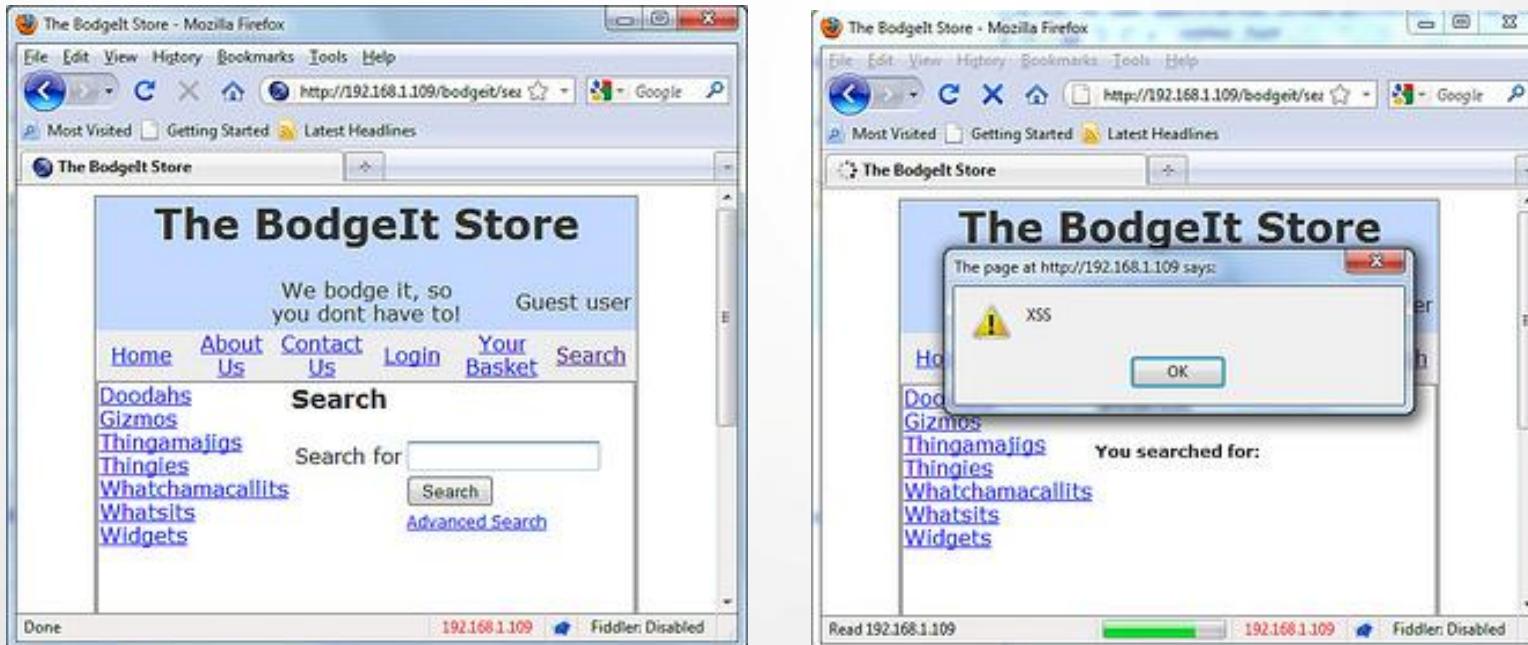
Cross Site Scripting (XSS) is an attack against the user of a website. It is a technique that forces a website to display malicious code, which then executes in the user's web browser. The attacker uses a vulnerable website to send malicious code to another end user of the site. The vulnerability arises when the website takes data in some way from a user and dynamically includes it in a web page without first validating that data.

- account hijacking
- rewrite portions of the page
- log keystrokes
- steal browser information
- Steal client machine data
- attack the user's network

- ANYTHING A USER CAN DO OR ACCESS FROM THE BROWSER!**

Manual Testing

```
<SCRIPT>alert('XSS')</SCRIPT>
```



XSS Cheat-Sheet: <http://ha.ckers.org/xss.html>

OWASP Broken Web Applications (Vulnerable Applications to Hack): https://www.owasp.org/index.php/OWASP_Broken_Web_Applications_Project

Browser Plugins

XSS Me

Security Compass

XSS Me

XSS-Me is a tool to aid in testing for Cross-Site Scripting vulnerabilities in the current page.

Each tab in the sidebar represents a form on the current page and lists all of the fields in the form.

query Change this to the value you want tested

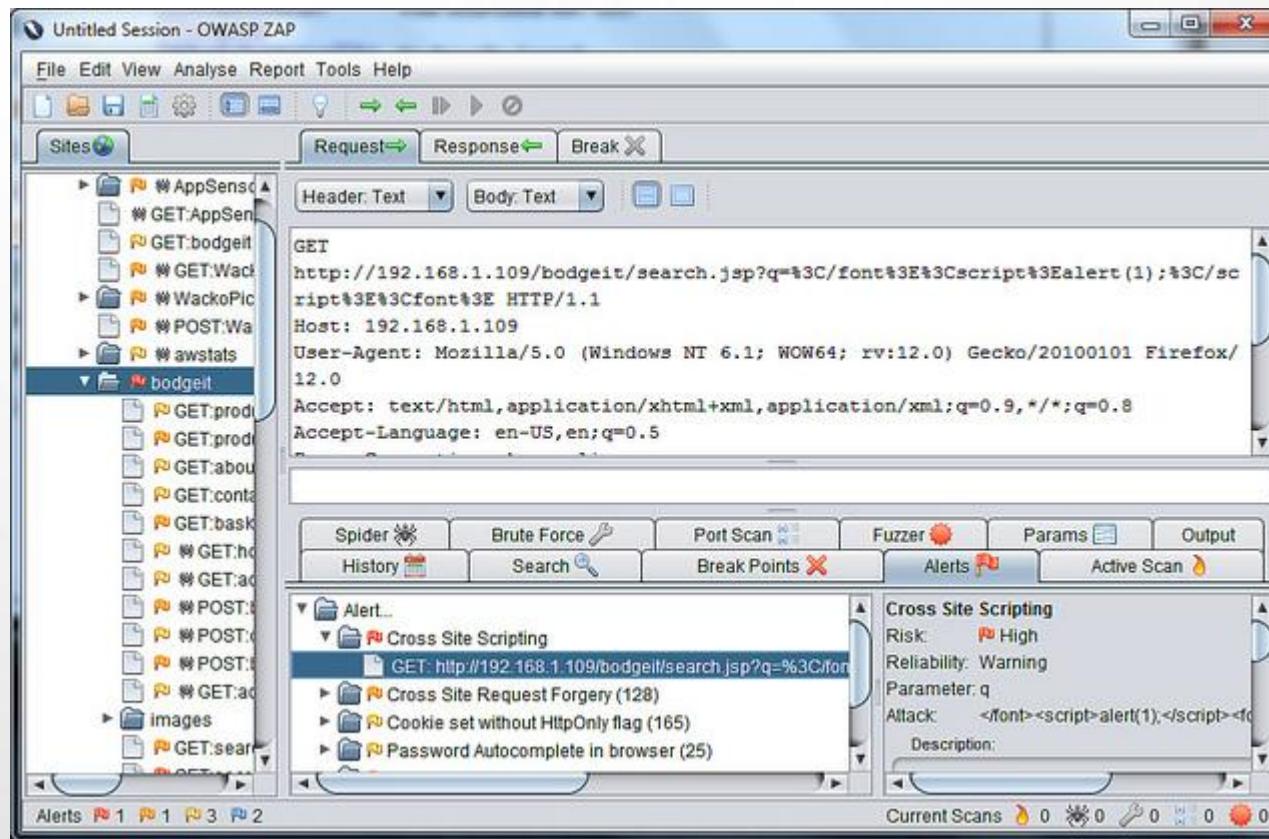
Temper Popup

http://192.168.1.339/WackoPicks/users/login.php

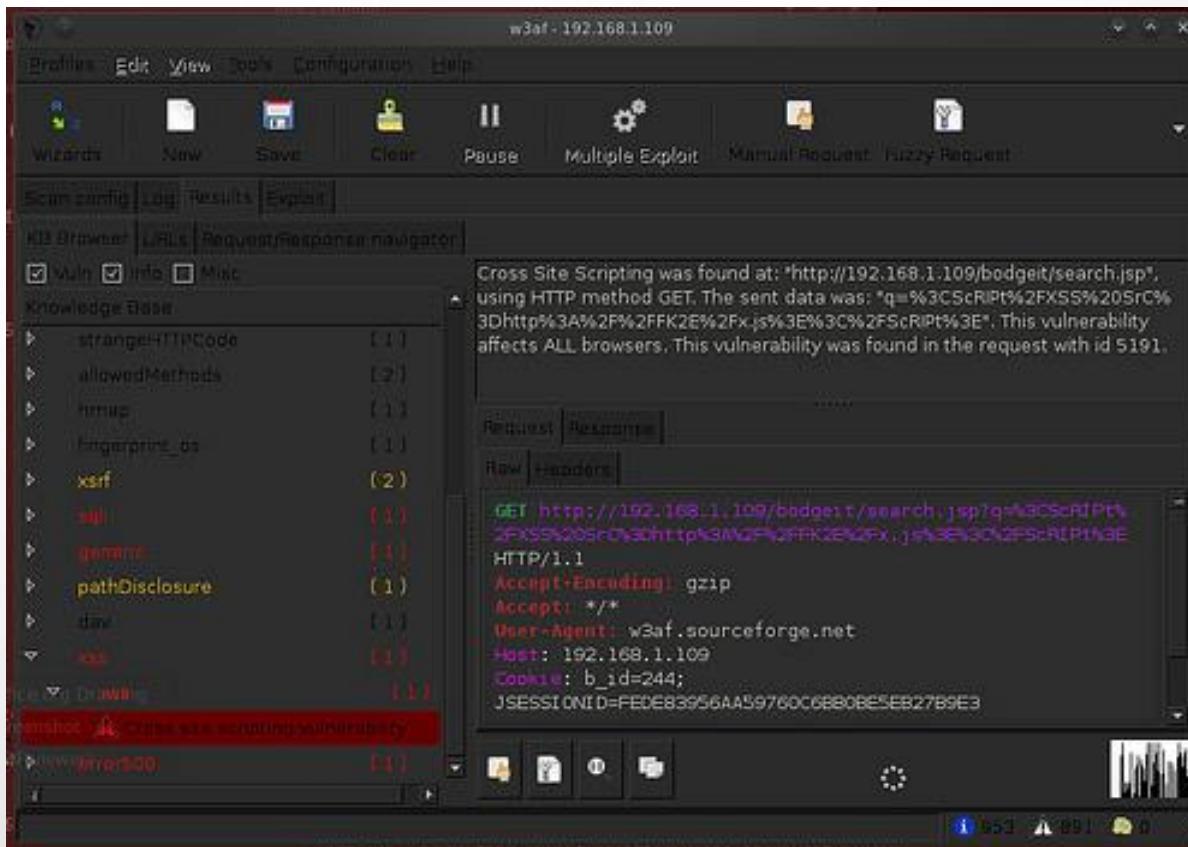
Request Header Name	Request Header Value	Post Parameter Name	Post Parameter Value
Host	192.168.1.339	username	Alert;pub1
User-Agent	Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:0.9.1) Gecko/2009062401 Fx/3.0.13	password	<input type="button" value="Add element:inline"/> <input type="button" value="Add elements from file"/> <input type="button" value="Add"/> <input type="button" value="Delete Element"/> <input type="button" value="Encode"/> <input type="button" value="Decode"/> <input type="button" value="Encode Base 64"/> <input type="button" value="Decode Base 64"/> <input type="button" value="Decimal HTML"/> <input type="button" value="Hex HTML"/> <input type="button" value="un-HTML"/> <input type="button" value="data"/> <input type="button" value="xss"/> <input type="button" value="sql"/>
Accept	text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8		
Accept-Language	en-US,en;q=0.5		
Accept-Encoding	gzip, deflate		
Connection	keep-alive		
Referer	http://192.168.1.339/WackoPicks/users/login.php		
Cookie	JSESSIONID=8885		

» Alert
object:Alert
Alert
%2B Alert
onload Alert
%22 Alert
no angle brackets alert
table Alert
image Alert
background Alert

Web Application Vulnerability Scanners



Web Application Vulnerability Scanners



Preventing Cross-Site Scripting

[https://www.owasp.org/index.php/XSS_\(Cross_Site_Scripting\)_Prevention_Cheat_Sheet](https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet)

Input Validation

Accept known good (whitelist)

Reject known bad (blacklist)

Sanitize (change input to acceptable format)

Output Encoding / Escaping

Characters will still render in a browser correctly; escaping simply lets the interpreter know the data is not meant to be executed.

& → &
< → <
> → >
" → "
' → '
/ →

.....



XSS

Preventing Cross-Site Scripting

Use Libraries

ESAPI – <https://www.owasp.org/index.php/ESAPI>

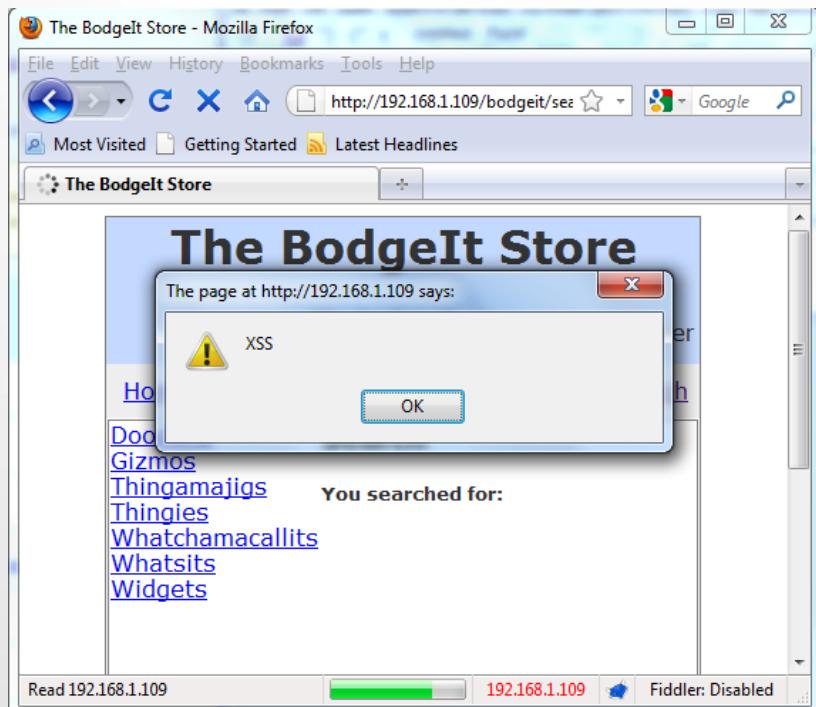
MS Anti-XSS Library - <http://wpl.codeplex.com>



• • • • •

Cross-Site Scripting Reporting

Seriously?



The value of the `search_txt` request parameter is copied into the value of an HTML tag attribute which is not encapsulated in any quotation marks. The payload `<script>alert(XSS)</script>` was submitted in the `search_txt` parameter. This input was echoed unmodified in the application's response.

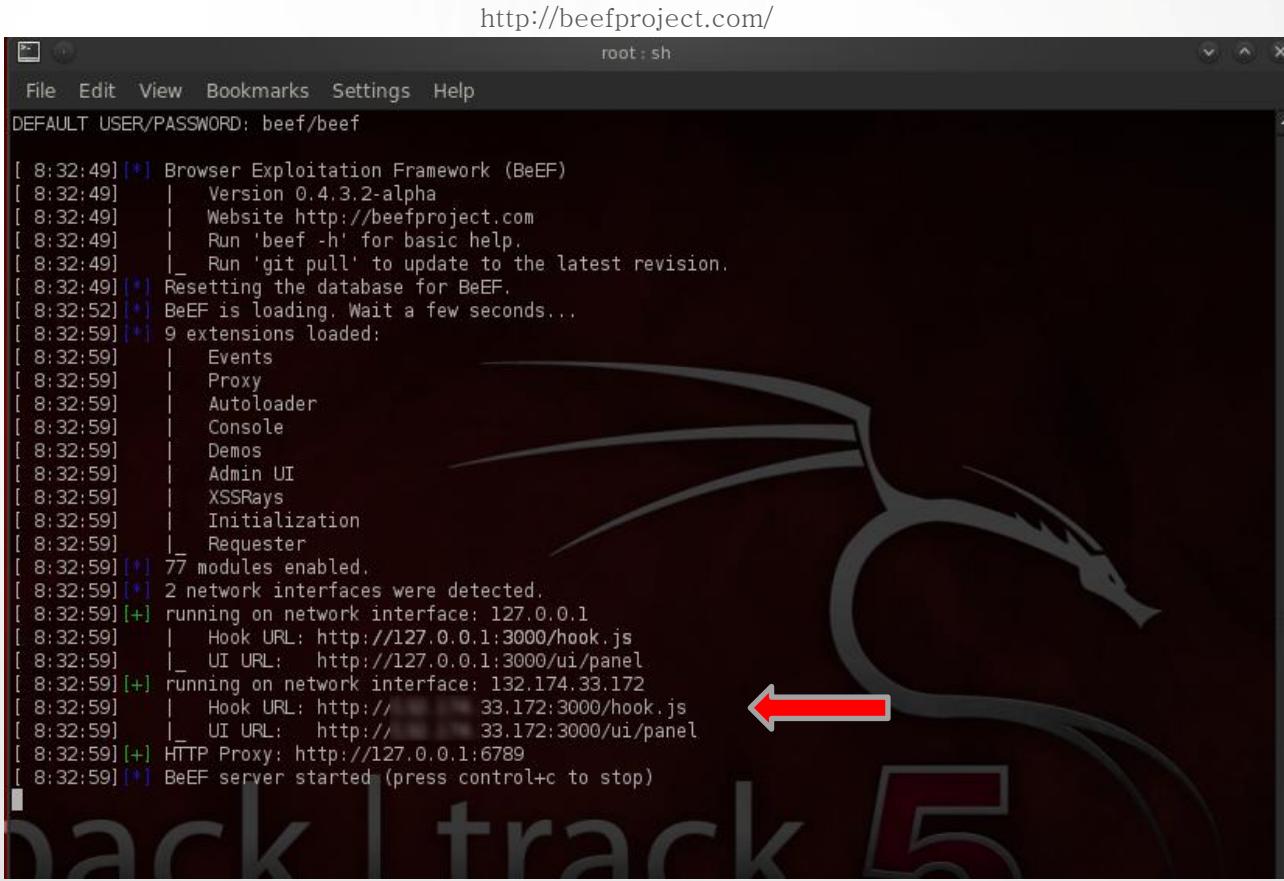
This proof-of-concept attack demonstrates that it is possible to inject arbitrary JavaScript into the application's response.

Browser Exploitation Framework (BeEF)

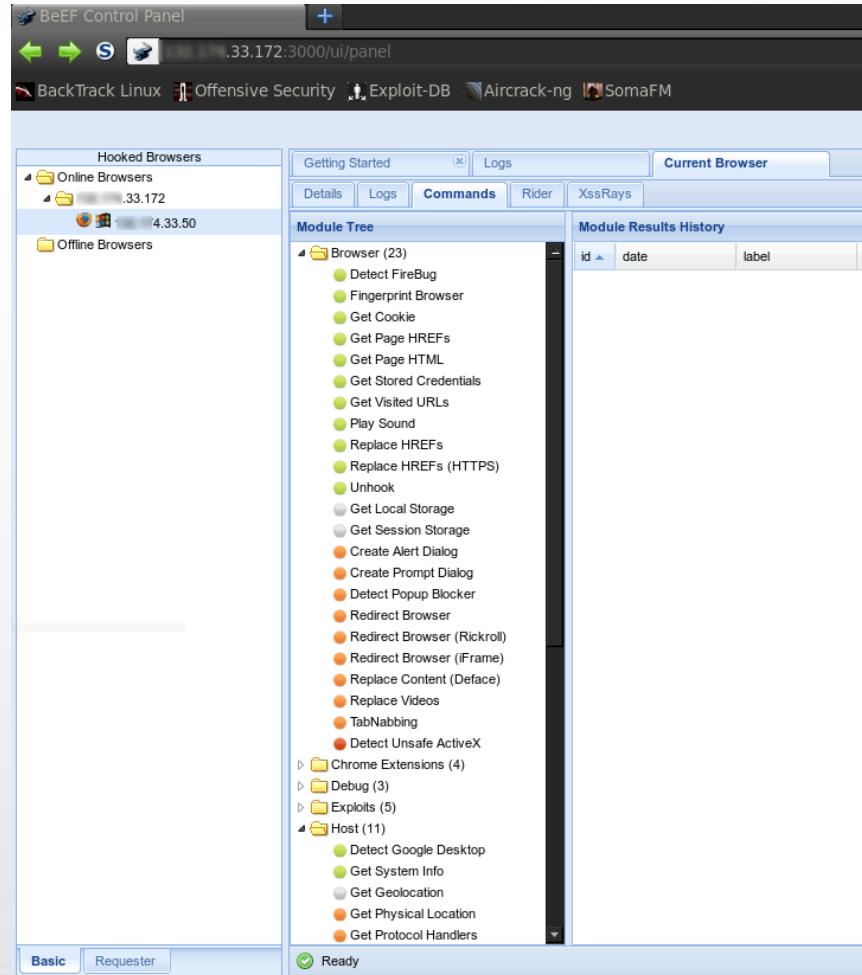
http://beefproject.com/

```
root : sh
File Edit View Bookmarks Settings Help
DEFAULT USER/PASSWORD: beef/beef

[ 8:32:49] [*] Browser Exploitation Framework (BeEF)
[ 8:32:49] | Version 0.4.3.2-alpha
[ 8:32:49] | Website http://beefproject.com
[ 8:32:49] | Run 'beef -h' for basic help.
[ 8:32:49] | Run 'git pull' to update to the latest revision.
[ 8:32:49] [*] Resetting the database for BeEF.
[ 8:32:52] [*] BeEF is loading. Wait a few seconds...
[ 8:32:59] [*] 9 extensions loaded:
[ 8:32:59] | Events
[ 8:32:59] | Proxy
[ 8:32:59] | Autoloader
[ 8:32:59] | Console
[ 8:32:59] | Demos
[ 8:32:59] | Admin UI
[ 8:32:59] | XSSRays
[ 8:32:59] | Initialization
[ 8:32:59] | Requester
[ 8:32:59] [*] 77 modules enabled.
[ 8:32:59] [*] 2 network interfaces were detected.
[ 8:32:59] [*] running on network interface: 127.0.0.1
[ 8:32:59] | Hook URL: http://127.0.0.1:3000/hook.js
[ 8:32:59] | UI URL: http://127.0.0.1:3000/ui/panel
[ 8:32:59] [*] running on network interface: 132.174.33.172
[ 8:32:59] | Hook URL: http:// 33.172.3000/hook.js
[ 8:32:59] | UI URL: http:// 33.172.3000/ui/panel
[ 8:32:59] [*] HTTP Proxy: http://127.0.0.1:6789
[ 8:32:59] [*] BeEF server started (press control+c to stop)
```



Cross-Site Scripting Exploitation



XSS

Socially Engineering your Report

Exploit the Vulnerability! Report the Impact!

Company
Wrong Company

User Name
user

Password
*****|

LOGIN

Remember me on this
computer

Socially Engineering your Report

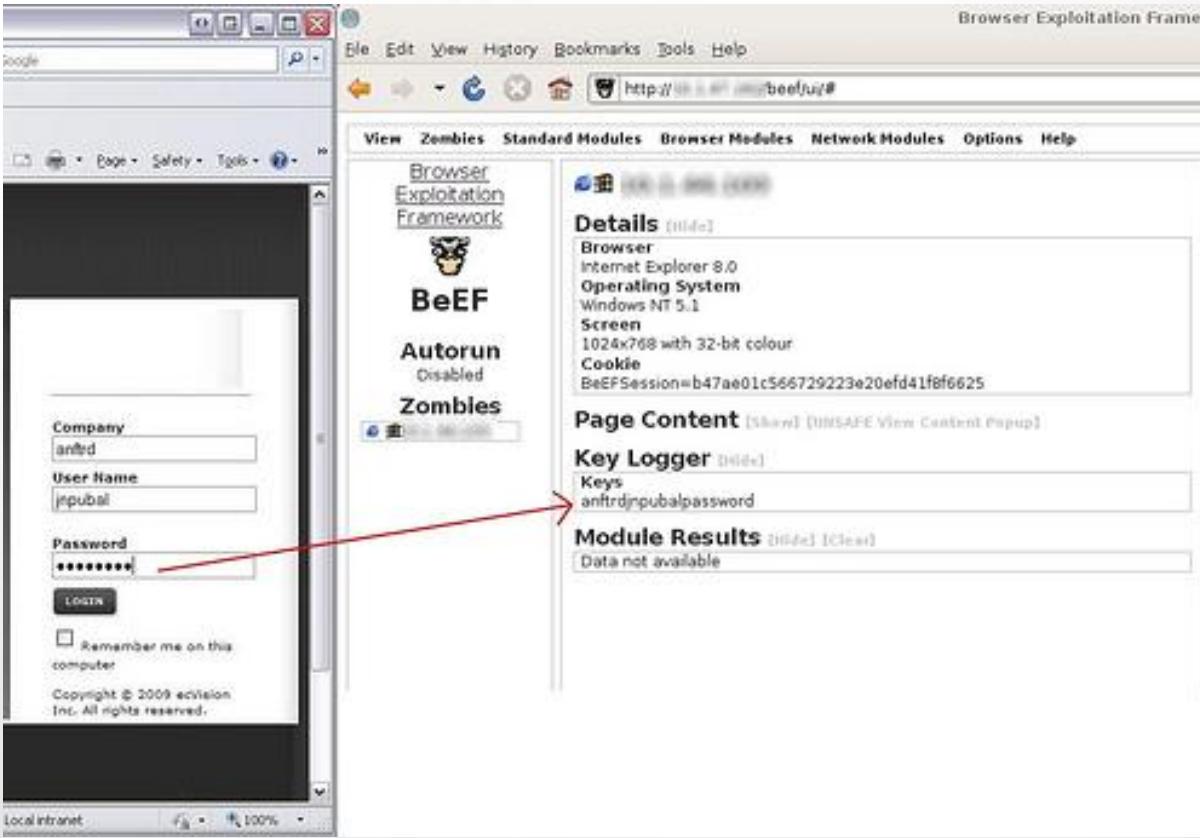
Exploit the Vulnerability! Report the Impact!



.....

Socially Engineering your Report

Exploit the Vulnerability! Report the Impact!



The image shows a screenshot of the BeEF (Browser Exploitation Framework) interface. On the left, a browser window displays a login page for 'ecvision Inc.' with fields for 'Company', 'User Name', 'Password', and a 'LOGIN' button. A red arrow points from the 'Password' field in the browser to the 'Key Logger' section in the BeEF interface on the right. The BeEF interface includes tabs for 'View', 'Zombies', 'Standard Modules', 'Browser Modules', 'Network Modules', 'Options', and 'Help'. The 'Zombies' tab is selected, showing details for a target browser: Internet Explorer 8.0 on Windows NT 5.1 with a 1024x768 resolution. The 'Page Content' section shows the captured password 'anfrdjinpubalpassword'. The 'Key Logger' section also displays this password. The 'Module Results' section shows 'Data not available'.

Browser Exploitation Framework

File Edit View History Bookmarks Tools Help

http://192.168.1.100:8080/beef/u/1#

View Zombies Standard Modules Browser Modules Network Modules Options Help

Browser Exploitation Framework

BeEF

Autorun Disabled

Zombies

Details [Hide]

Browser: Internet Explorer 8.0
Operating System: Windows NT 5.1
Screen: 1024x768 with 32-bit colour
Cookie: BeEFSession=b47ae01c566729223e20efd41f8f6625

Page Content [Share] [Unsafe View Content Popup]

Key Logger [Hide]

Keys: anfrdjinpubalpassword

Module Results [Hide] [Clear]

Data not available

Company: anfrd
User Name: jnpubal
Password: anfrdjinpubalpassword

Copyright © 2009 ecvision Inc. All rights reserved.

Local Intranet 100% • • • • •

Copy Machine Experiment

The Power of “Because”



“May I use the Xerox machine?”

Giving no reason - 60%

“May I use the Xerox machine, because I have to make copies?”

Giving no real reason - 93%

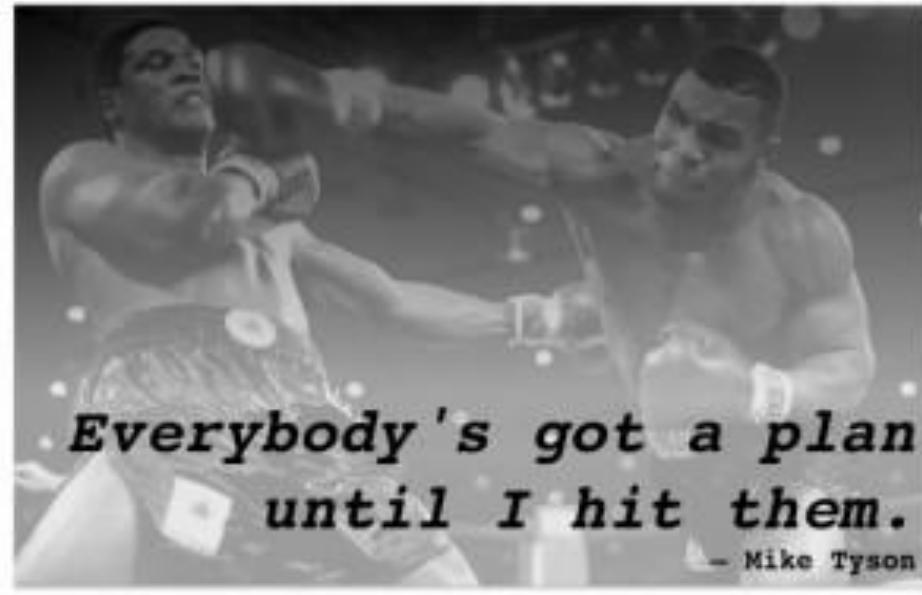
“May I use the Xerox machine, because I’m in a rush?”

Giving a reason - 94%

.....

Commander's Intent

Give Them a Reason!



.....

Bystander Apathy

Assign a JIRA Ticket!



.....

Contrast Frame

NLP

The Ponemon Institute puts the cost per record of a breach at \$214, with an average cost of 7.2 million dollars. By contrast, a week of development time seems cheap.

Options

1 - \$5,000	95% Effective
2 - \$500	80% Effective



.....

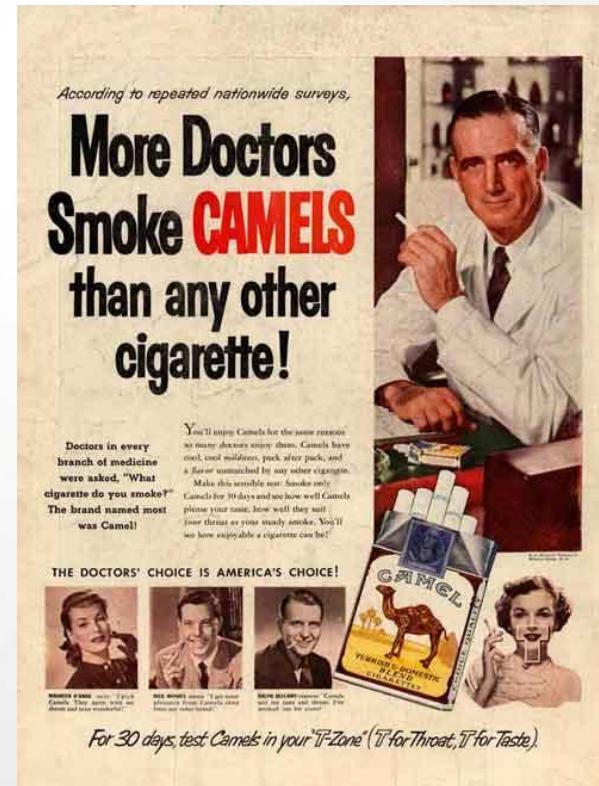
Herd Effect

You're all sheep.

Best Practices

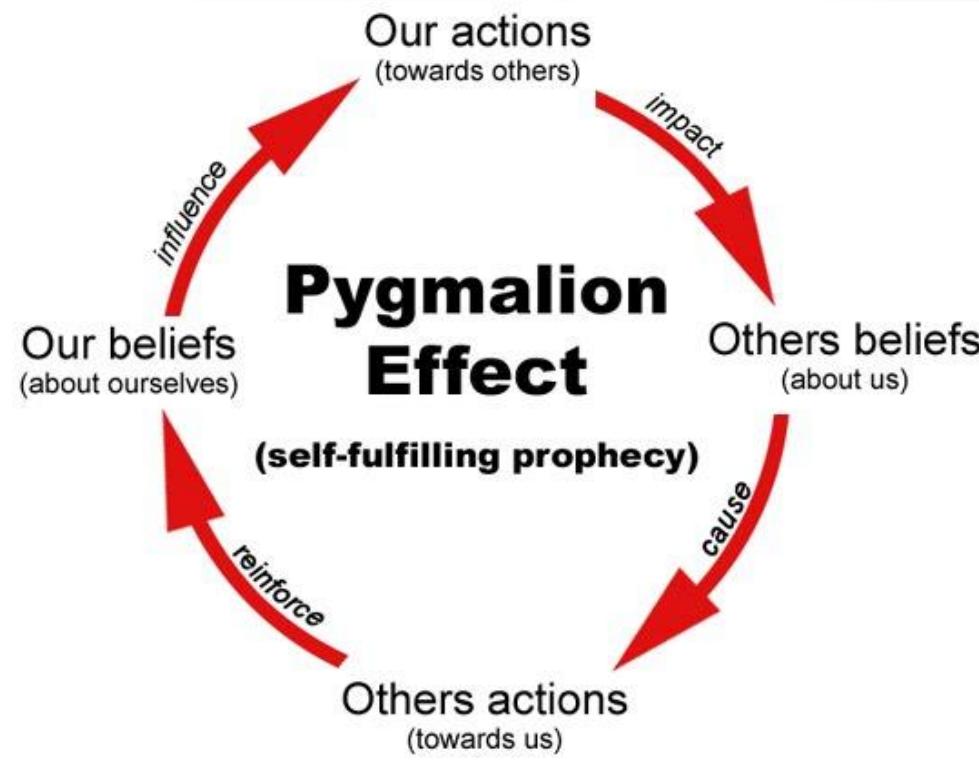
Amazon and Facebook employ CAPTCHA

93% of Websites in our Industry use Input Validation



Pygmalion Effect

Clearly Communicate Expectations



.....

XSS

Metrics

If you want to improve something, measure it.

Measure to see if what you're doing is working. If not, try something else.

• • • • • • •

THANK YOU

Questions?!

