OWASP AppSec EU
August 20-23, 2013 Hamburg

The OWASP Foundation
http://www.owasp.org

OWASP OWTF
Summer Storm

Abraham Aranguren

OWASP OWTF Project Leader
@7a_ @owtfp
abraham.aranguren@owasp.org

Agenda

- GSoC Overview
- What is OWASP OWTF?
- Status update on OWTF GSoC projects
  - OWTF Reporting
  - OWTF Multiprocessing
  - OWTF MiTM Proxy
  - OWTF Testing Framework
- OWASP Testing Guide with OWTF
- Conclusion

Agenda

- GSoC Overview
- What is OWASP OWTF?
- Status update on OWTF GSoC projects
  - OWTF Reporting
  - OWTF Multiprocessing
  - OWTF MiTM Proxy
  - OWTF Testing Framework
- OWASP Testing Guide with OWTF
- Conclusion

Google Summer of Code (GSoC) Overview

GSoC Stats + Outcome

- OWASP got **11** slots from Google
- OWASP received **84** proposals
- **73** students (**87%**) could **not** be selected.
- Final slot breakdown:
  - **4** - OWASP ZAP
  - **4** - OWASP OWTF
  - **1** - OWASP Hackademic
  - **1** - OWASP ModSecurity
  - **1** - OWASP PHP Security Project

http://blog.7-a.org/2013/06/owasp-owtf-gsoc-selection-stats-and-poll.html

OWTF GSoC Overview

- **14** students showed interest (email)
- **11** (**79%**) students submitted a proposal
- **14** proposals were submitted (**16%** of 84)
- **5** OWTF proposals ended in the top 11
- **1** student was lost in de-duplication process (accepted by another org)
- **4** OWTF proposals were finally selected (**36%** of 11)

Why submit for OWTF?

OWTF GSoC student poll summary:

- “It’s python”
- “I like this project”
- “It’s a project I can do with my skills”
- “OWTF is the best project to learn about other tools/security”
- “Other mentors/org didn’t reply” (!)
- “Quick feedback/encouragement/advice”

http://blog.7-a.org/2013/06/owasp-owtf-gsoc-selection-stats-and-poll.html

Selected OWTF Proposals

- **Reporting**: Assem Chelli
- **Multiprocessing**: Ankush Jindal
- **MiTM Proxy**: Bharadwaj Machiraju
- **Testing Framework**: Alessandro Fanio González

http://blog.7-a.org/2013/06/owasp-owtf-gsoc-selection-stats-and-poll.html

Dedicated OWTF mentors

Without them **3** OWTF students would have been **lost** (GSoC 1 dedicated mentor x student rule):

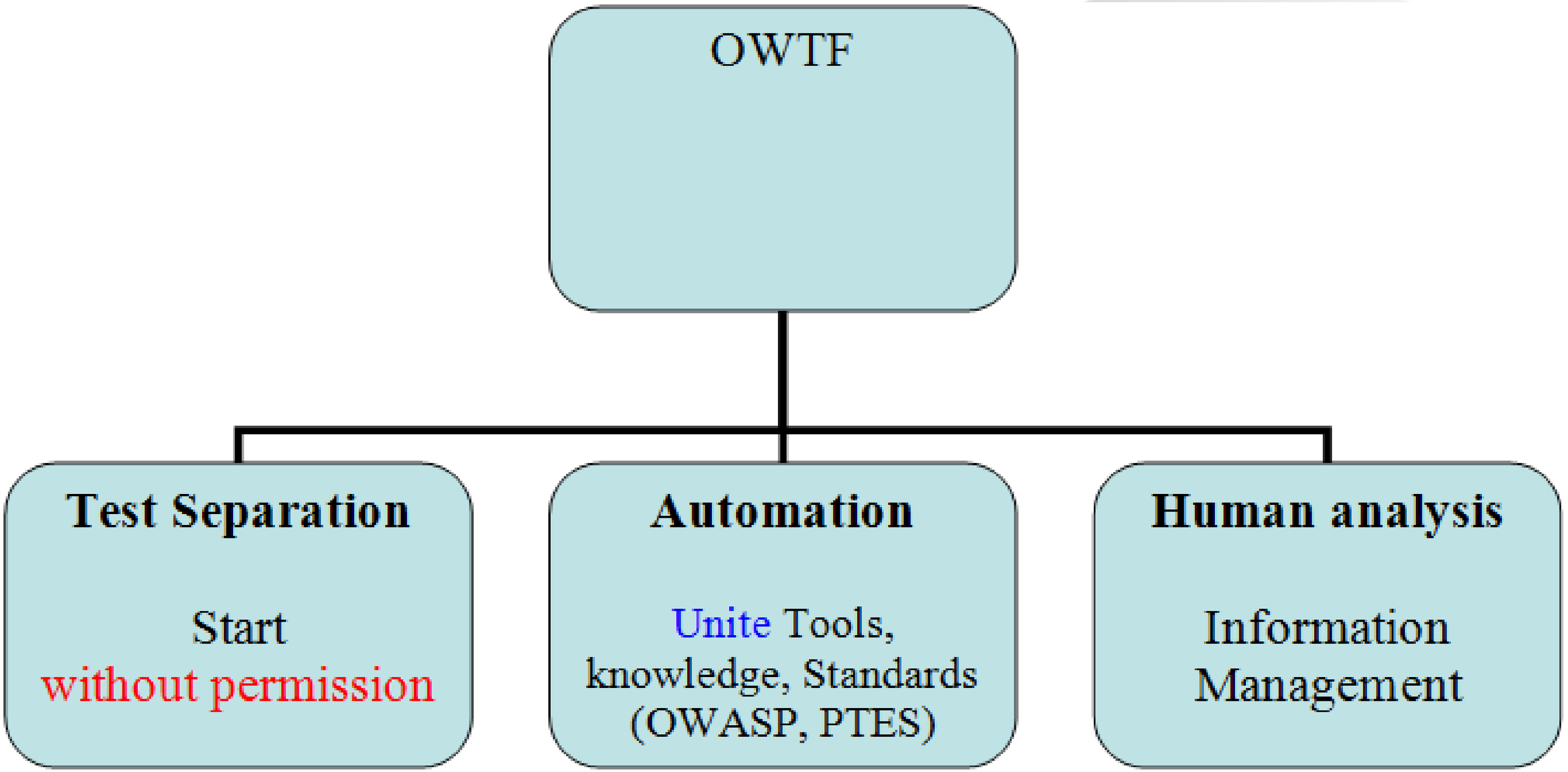
Andrés Morales, Andrés Riancho, Azeddine Islam Mennouchi, Gareth Heyes, Hani Benhabiles, Javier Marcos de Prado, Johanna Curiel, Krzysztof Kotowicz, Martin Johns

THANK YOU for stepping up!

What is OWASP OWTF?

aka *The Offensive (Web) Testing Framework*

OWTF = Test/Exploit ASAP



OWTF's Chess-like

Runs Tools

- theHarvester
- Nikto
- Arachni
- w3af, etc.

Runs Tests directly

- Header searches
- HTML body searches
- Crafted requests, etc.

Knowledge Repository

- PoC links
- Resource links
- OWASP mapping

Helps Human analysis

- Flag importance
- Tool output manager
- Screenshot manager
- Notes manager
- Report assistant

OWTF Plugin Groups (-g)

- **web**: Try to cover the OWASP Testing Guide

*owtf.py http://demo.testfire.net (-g web: optional) ← **web only***
owtf.py –l web ← List web plugins

- **net**: Somewhat like nmap scripts

*owtf.py demo.testfire.net (-g net: optional) ← **portscan + probe***
NOTE: if a web service is found, web plugins will also run
owtf.py –l net ← List net plugins

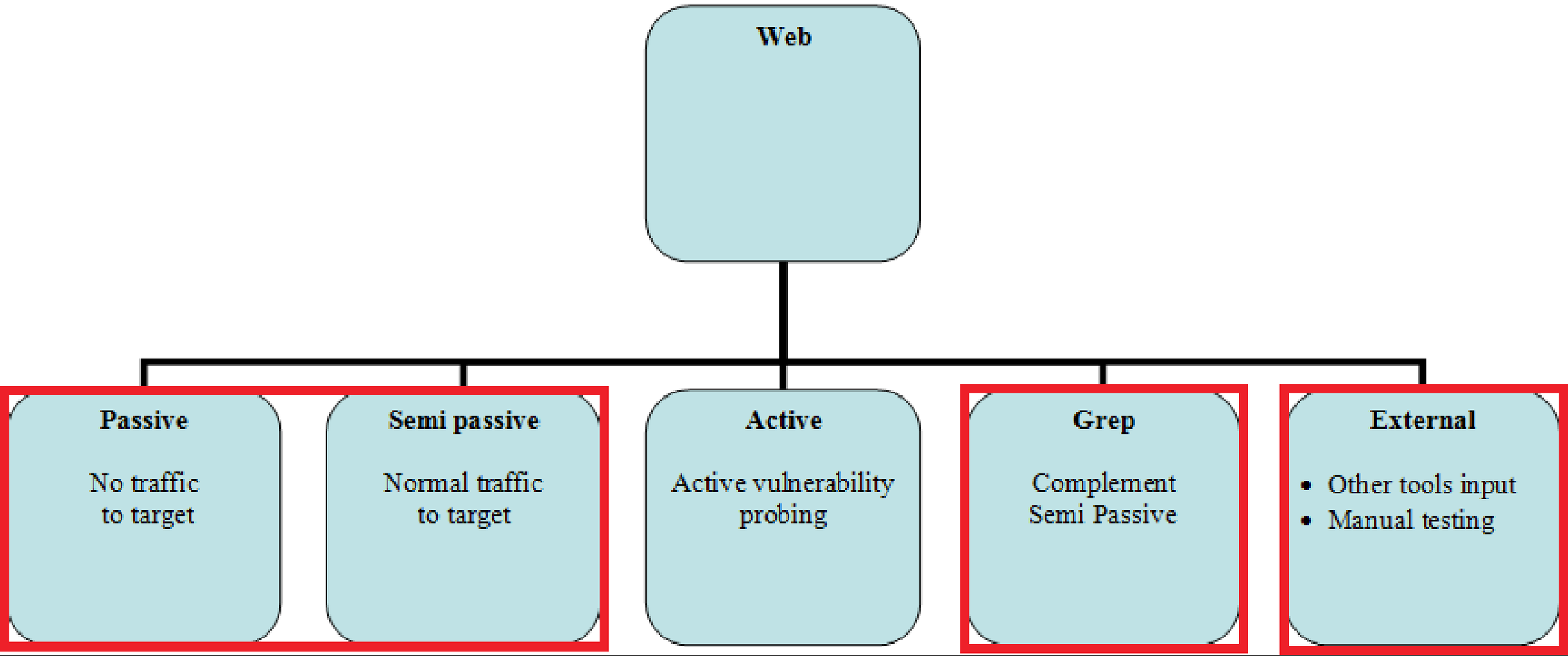
- **aux**: Somewhat like msfcli in metasploit

*owtf.py -f -o Targeted_Phishing SMTP_HOST=mail.pwnlabs.es SMTP_PORT=25 SMTP_LOGIN=victim SMTP_PASS=victim EMAIL_FROM=sevena@pwnlabs.es EMAIL_PRIORITY=no EMAIL_SUBJECT='Test subject' EMAIL_BODY='test_body.txt' EMAIL_TARGET='victim@pwnlabs.es' ← **Phishing via SET***
owtf.pl –l aux ← List aux plugins

Web Plugin Types (-t)

At least **50%** (32 out of 64) of the tests in the OWASP Testing guide can be legally* performed <u>to some degree</u> **without permission**

* Except in Spain, where visiting a page can be illegal ☺
* This is only my interpretation and not that of my employer + might not apply to **your** country!



OWTF Report = Chess-like Analysis

You need to understand this to use the OWTF report efficiently ☺

From Alexander Kotov - "Think like a Grandmaster":
1) Draw a list of candidate moves (3-4) ← ***1st Sweep (!deep)***
1) Draw up a list of candidate paths of attack = ***rank what matters***

2) Analyse each variation only once (!) ← ***2nd Sweep (deep)***
2) Analyse [tool output + other info] once and only once

3) After step 1 and 2 make a move
3) After 1) and 2) exploit the best path of attack

Ever analysed X in depth to only see “super-Y” later?

Demo 1: Admin interface

Pre-Engagement: No permission to test ← preparation

1) Run passive plugins ← **legit + no traffic to target**
Sitefinity CMS found

2) Identify best path of attack:

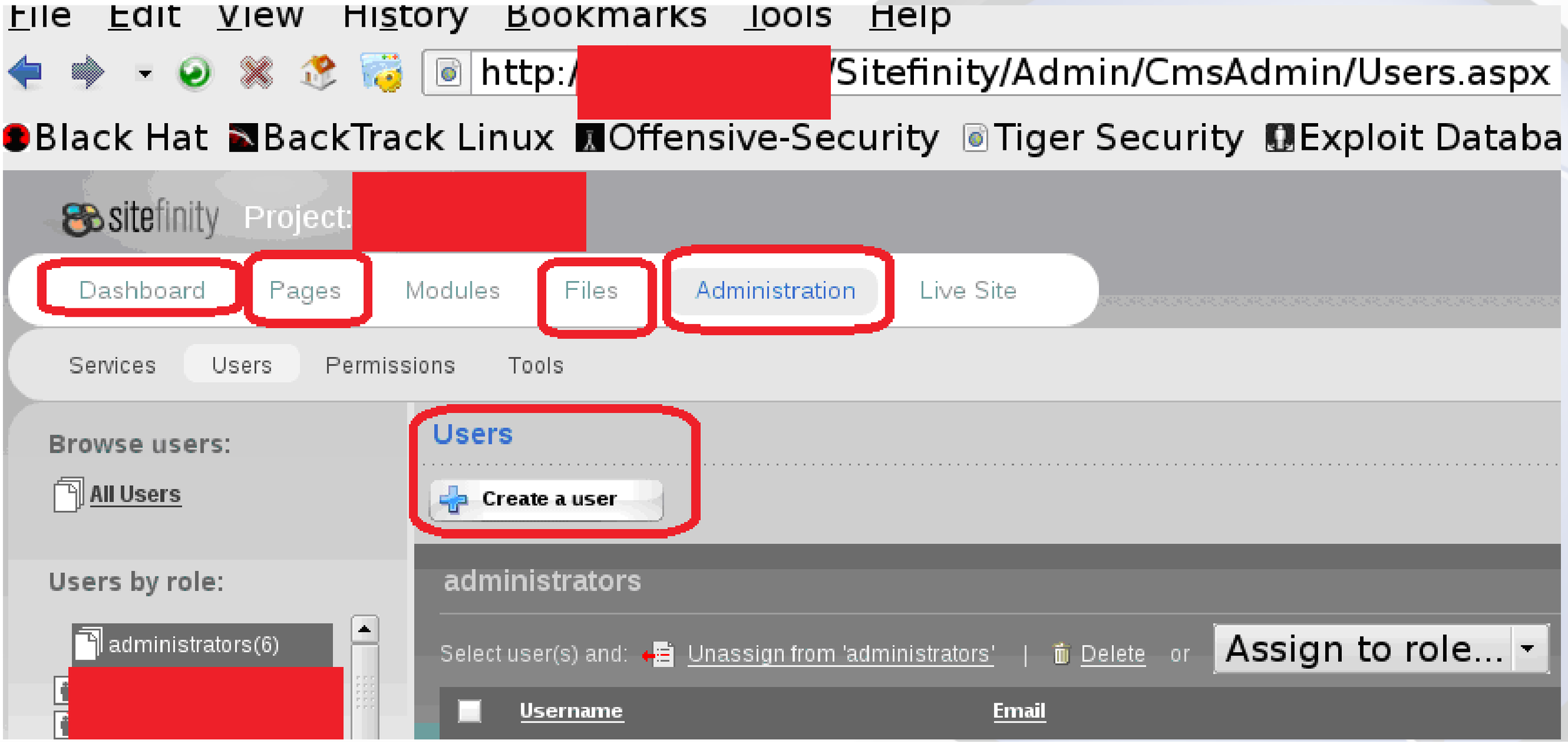
- *Sitefinity default admin password*
- *Public sitefinity shell upload exploits*

Engagement: Permission to test ← exploitation

Try best path of attack first

Demo 1: Outcome

1 minute after getting permission …



Demo 1: Outcome

5 minutes after getting permission …

BackTrack Linux Offensive Security Exploit-DB Aircrack-

```
El volumen de la unidad C no tiene etiqueta.
El n£mero de serie del volumen es: 5CFC-2842

 Directorio de c:\windows

14/02/2013  09:44

               .
      14/02/2013  09:44
                    ..
           03/03/2013  23:22                 0 0.log
           14/06/2002  18:46            19.274 000001_.tmp
           17/07/2004  11:40            19.528 002292_.tmp
           29/12/2006  00:31            19.569 002961_.tmp
           24/08/2001  16:00            17.336 A pescar.bmp
           04/09/2007  17:14           180.000 aaRemove.exe
           24/08/2001  16:00            26.680 Abanicos.bmp
           27/04/2008  18:13
```

Demo 2: Crossdomain

Attack preparation (pre-engagement safe) ← preparation

1) Run semi-passive plugins ← **legit**

Missconfigured crossdomain, fingerprint wordpress version

2) Identify best path of attack:

crossdomain + phishing + wordpress plugin upload + meterpreter

3) Replicate customer environment in lab

4) Prep attack: *Adapt public payloads to target*

5) Test in lab

Launching the attack ← exploitation

1) Tested attack works flawlessly on the first shot

2) Pivot

3) Show impact

OWTF Financials: Ideas plz ☺

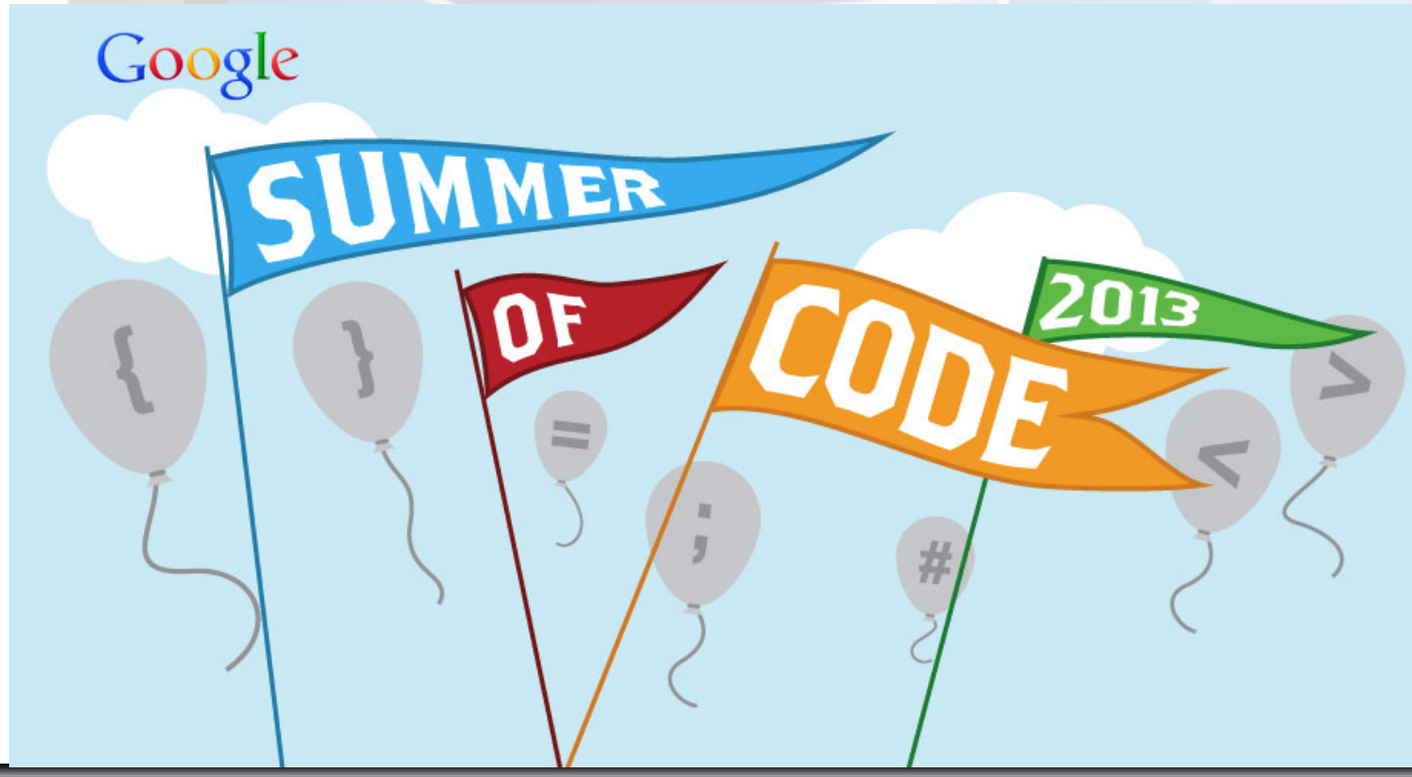
Funding granted so far (THANK YOU Brucon + Google!):

- €5,000 – Brucon 5x5

http://blog.brucon.org/2013/02/the-5by5-race-is-on.html

- $2,000 – GSoC ($500 x student)

What should we do with that money?



Status update on OWTF GSoC Projects

OWTF Reporting

by

Assem Chelli

Dedicated Mentor: Gareth Heyes (@garethheyes)
Co-mentors: Azeddine Islam Mennouchi, Hani Benhabiles, Johanna Curiel, Abraham Aranguren

Reporting Agenda

- Old report limitations
- Reporting goals
- Pre-implementation research
- Prototype voting/feedback
- Upcoming features

Old Report != Sexy ☹

Summary Report

SEED	REVIEW SIZE	TOTAL SIZE	VERSION	SITE
uHjQPnC13s	22 KB	39 KB	0.20 "Summer Storm I"	owtf.org

Filter | Review | History | Logs | Miscelaneous: | Exploitation | Methodology | Calculators | Test/Learn

172 172 172 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

178.32.28.116

- 443

HTTPS://WWW.ALFANOUS.ORG | 178.32.28.116 | 443

Filter | Review | History | Logs

86 86 86 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Spiders, Robots, and Crawlers (OWASP-IG-001) robots.txt Analysis

Results: external | passive | **semi_passive**

Spiders Robots And Crawlers - ***SEMI PASSIVE***

PLUGIN	START	END	RUNTIME	OUTPUT FILES
semi_passive/Spiders_Robots_and_Crawlers@OWASP-IG-001.py	18/06/2013-06:43	18/06/2013-06:43	0s, 775ms	Browse

NOTES

Edit

https://www.alfanous.org:443/robots.txt was NOT found

HTTP TRANSACTIONS

REQUEST	RESPONSE
See Transaction 1 (0s, 772ms) Site F R H B	[] ERROR: The connection was not refused, unknown error! (Requester Object): https://www.alfa (<class 'urllib2.URLError'>, URLError('unknown url type: https',), <traceback object at 0x

Old report limitations

- Complicated + hard to understand
- Poor loading time of “big” reports (i.e. 30+ websites)
- Not cross-browser compatible (Firefox only)
- Inability to suit various screen sizes
- Not visually appealing :(
- Direct HTML generation from python code

Reporting Goals

- UI simplification + intuitiveness
- Better load time + responsiveness
- Cross-browser compatibility
- Improved screen size support (i.e. mobile users, etc)
- Improve visual appeal ← **with community backing**
- Build a skin system ← **Users can choose/create skins**
- Move HTML into template files:

!python = designer-friendly = more people can help us

- Optimise click flow + mouse movement

Pre-implementation research

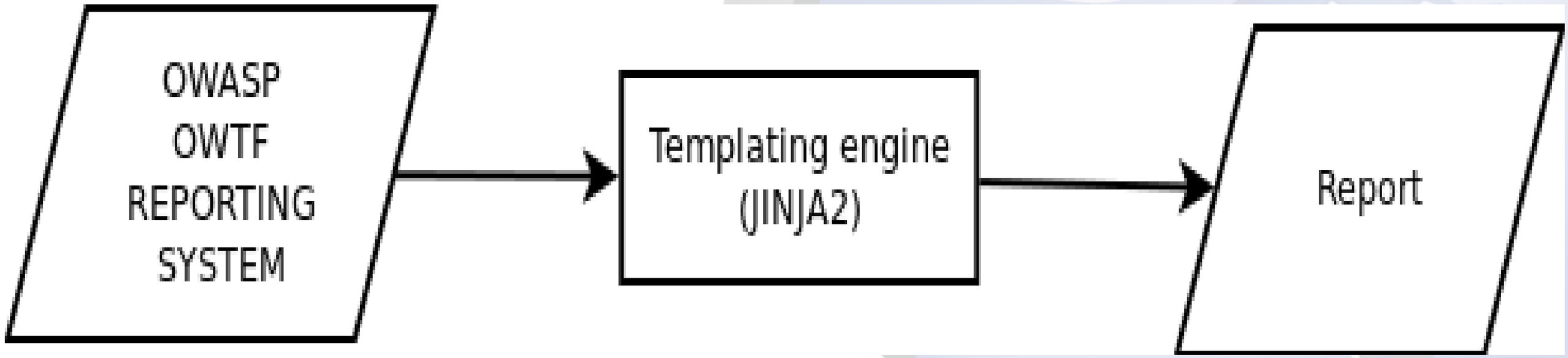
Twitter bootstrap gives us:

- Browser compatibility
- Pre-configured layouts
- Pre-defined styles
- Icon sets
- jQuery plugin integration
- Responsiveness + Simplicity

Pre-implementation research

Jinja2 gives us:

- A python templating engine
- Python-like expressions
- Templates evaluated in a sandbox



Prototype Voting/Feedback

Demo 3:

Online Survey Results

Want to vote? ☺ ← Shortcut: http://7-a.org + search “voting”
Survey:
https://docs.google.com/forms/d/1w613Y-rwPMw454k2oAd2M

Demo 4: Voted Prototype

OWTF Summary Report v0.20

Summary | Home | Browse | Review | History | Targets (4) | Plugins (23) | Logs | Links | Skins

145 | 23 | 21 | 0 | 2 | 24 | 16 | 21 | 21 | 5 | 5 | 2 | 23 | 2 | 4 | 1

Matches: 145

TARGETS

- 195.251.127.254
- 209.85.148.141
- 209.85.148.84
- 65.61.137.117

195.251.127.254:80 http://hackademic1.teilar.gr/ Review

General Info | OWASP Test Groups | Ports

...

Ports

content here

195.251.127.254:80 http://hackademic1.teilar.gr/ Review

Upcoming Features (WIP)

- Implement skin system
- Implement chosen prototype
- Extraction of CSS/HTML into templates
- Sub-report loading via AJAX
- **Default plugin vulnerability rankings**

OWTF Multiprocessing

by

Ankush Jindal

Dedicated Mentor: Andrés Riancho (@w3af)
Co-mentor: Abraham Aranguren

Multiprocessing Agenda

- Multiprocessing goals
- Pre-implementation research
- Development challenges
- Net plugins demo
- Upcoming features

Multiprocessing Goals

- **Reduce scanning time**
- Port of OSCP scripts into OWTF ← **net plugins**
- Scan multiple targets in parallel
- Rational usage of disk/RAM/CPU
- Stability + Reliability = !crash
- Identify + parallelise bottleneck components:

Plugin execution, Reporting

Pre-Implementation Research

- **Tested** candidate libraries:

Library	***Multiprocessing***	***Threading***	***gevent (distributed)***
Shared Memory	No	Yes	Yes

- **Results**:

1.Shared memory led to incorrect results in legacy code

2.Multiprocessing performed better or approx. the same

3.Threading = GIL FUD on multiple-core machines ☺

- **Conclusion**:

Multiprocessing for plugins, Threading for smaller tasks

Challenges during development

- OWTF resets config on the fly via “SwitchToTarget”

Solved via memory separation in multiprocessing

Process 1	**Process 2**
Config = Target 1	Config = Target 2

- Concurrent DB queries + no shared memory + File DB:

Solved via dedicated DB process + messaging system + file locks for integrity

(Processes perform DB reads+writes via messages)

- **Implemented ncurses interface to stop OWTF**
- Debugging unusual behaviour on concurrent processes ☺

Demo 5: Net Plugins

Port of the OSCP scripts into OWTF:

- Ping sweep + DNS zone transfers + port scanning
- Port scanning via nmap using “waves” (--portwaves)

owtf.py --portwaves=10,100,1000 target.com

First scan “top 10” ports, then “remaining until top 100”, ..

- Firing relevant **net** plugins depending on ports open

Net plugins implement:

- Vulnerability probing of network services (i.e. ftp, smtp,..)

Upcoming Features

- **Plugin profiling** for better resource usage:

Monitor resources to determine “launchable” plugins depending on [load + expected resource consumption]

- **Reporter process**:

To run in parallel + reduce report re-assembly iterations (i.e. instead of re-assemble once x plugin execution)

- Identify + parallelise other bottleneck components

OWTF MiTM Proxy

by

Bharadwaj Machiraju

Dedicated Mentor: Krzysztof Kotowicz (@kkotowicz)
Co-mentors: Javier Marcos de Prado, Martin Johns, Abraham Aranguren

MiTM Proxy Agenda

- MiTM Proxy Goals
- Pre-implementation research
- Development challenges
- Examples of working functionality ☺
- Performance benchmarks
- Upcoming features

MiTM Proxy Goals

- **Extended grep plugin coverage**:

  1) Data from manual browsing

  2) Data from proxified tools

- **Tool proxification** (if launched from OWTF)
- **SSL MiTM**
- **Proxy cache**: Avoid redundant requests
- **Request Throttling** based on target responsiveness

(i.e. avoid unintended DoS)

- **Intelligent request retries**

(i.e. ensure HTTP response retrieval where possible)

Pre-Implementation Research

- **Goal**:

Select best python proxy framework ← **best starting point**

- **Test Cases**:

Speed, HTTP Verb support, HTTP/1.1, HTTPS support, etc.

- **Frameworks**:

Twisted, Mitmproxy, Tornado, Honeyproxy

- **Verdict**: ***Tornado***

Best [performance + feature-set + reusability]

Pre-Implementation Research

MiTM Proxy

Pre-Implementation

Research Doc

Development Challenges

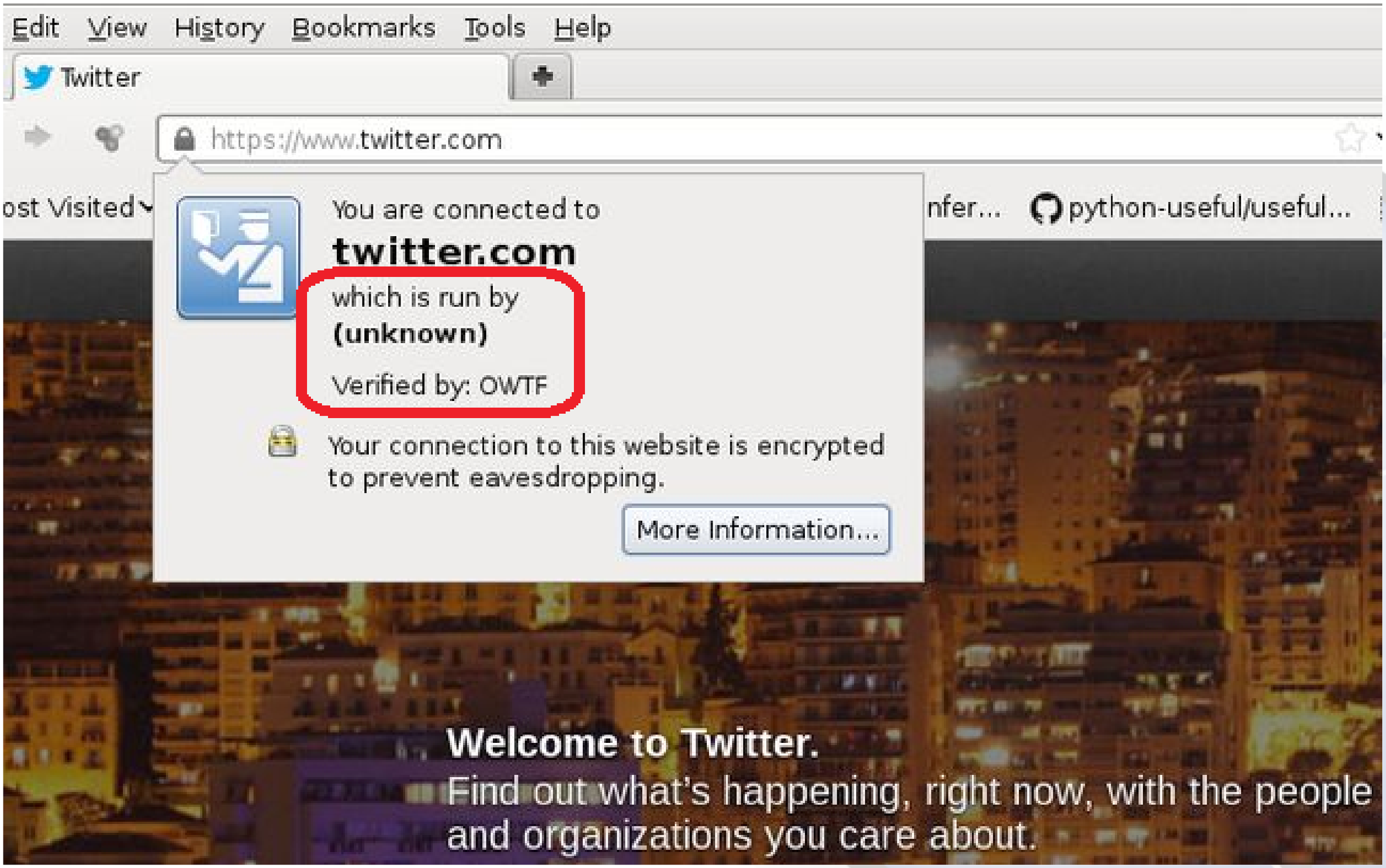
- **Tornado**: Is a python **web framework** (!proxy)

Pros	**Cons**
Scalability: Tens of thousands of connections	Not built to make proxy servers
Server + Client = Proxy	Client is more limited than server. **Solution**: Use tornado's async curl client

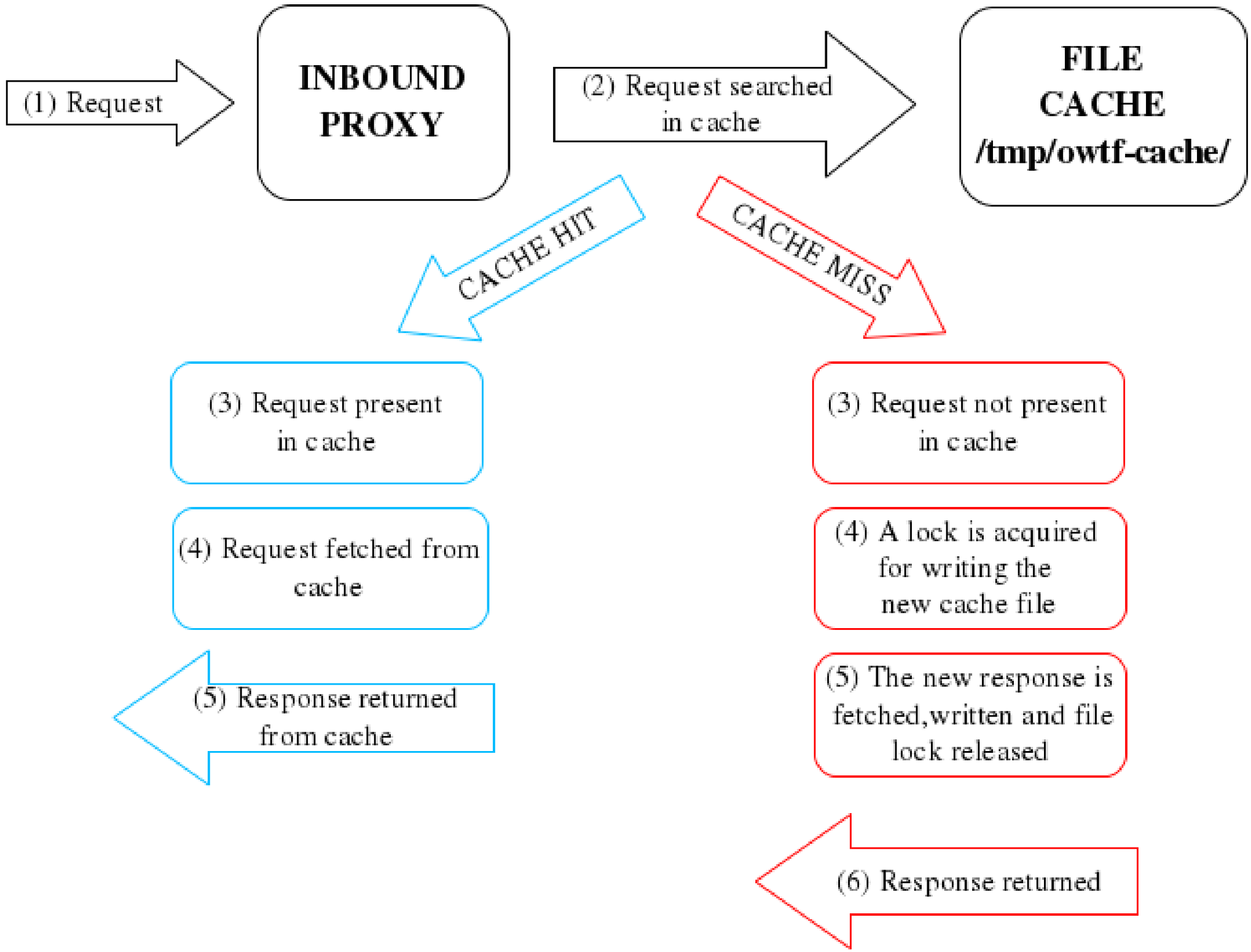
- **SSL MiTM**: on-the-fly certificate generation, etc.
- **Proxy cache**: Race condition handling
- **Tool Proxification**: Not all tools could be proxified

BUT Tool Proxification for tools with proxy CLI options IS working ☺

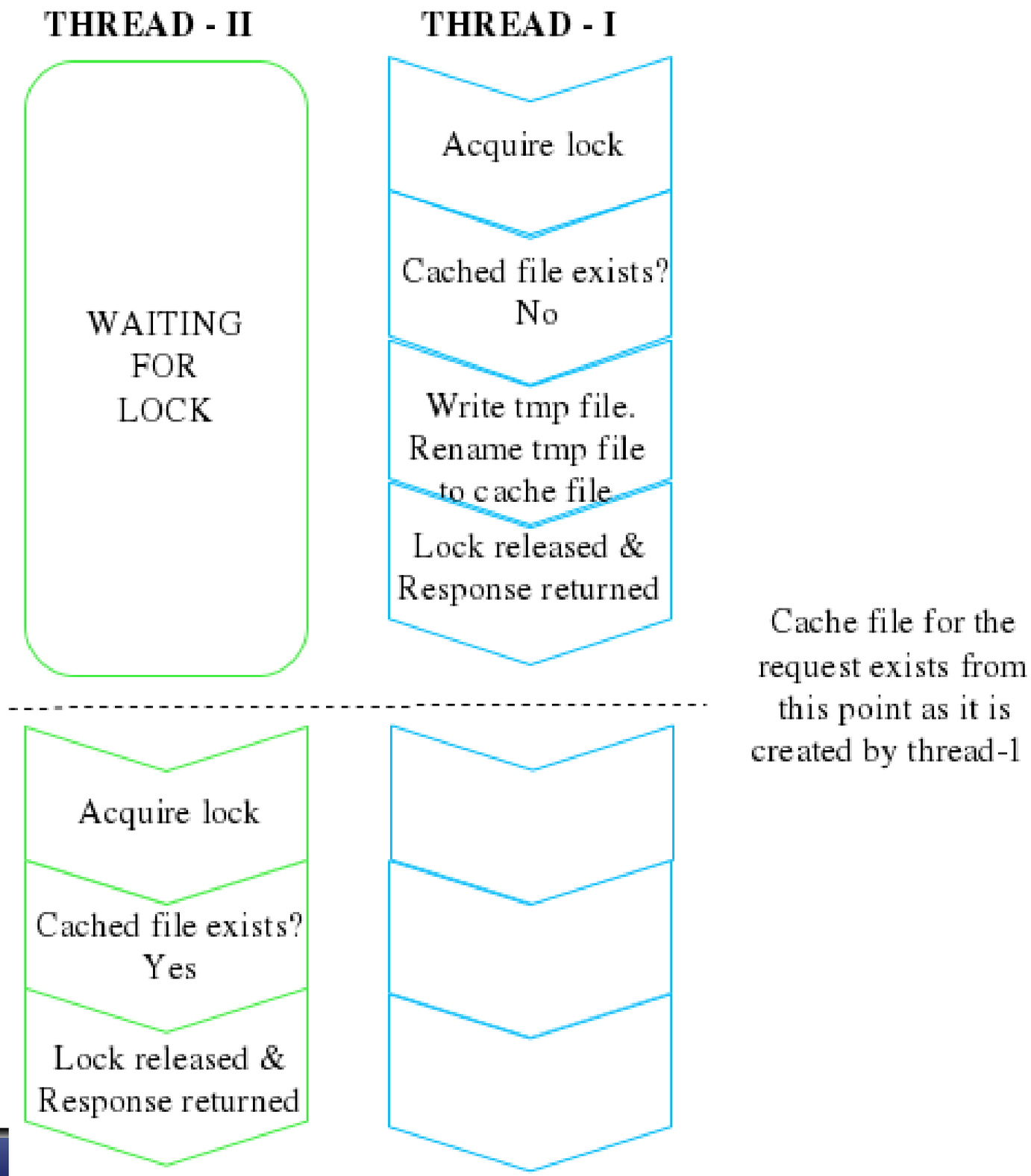
Proxy SSL MiTM is working ☺



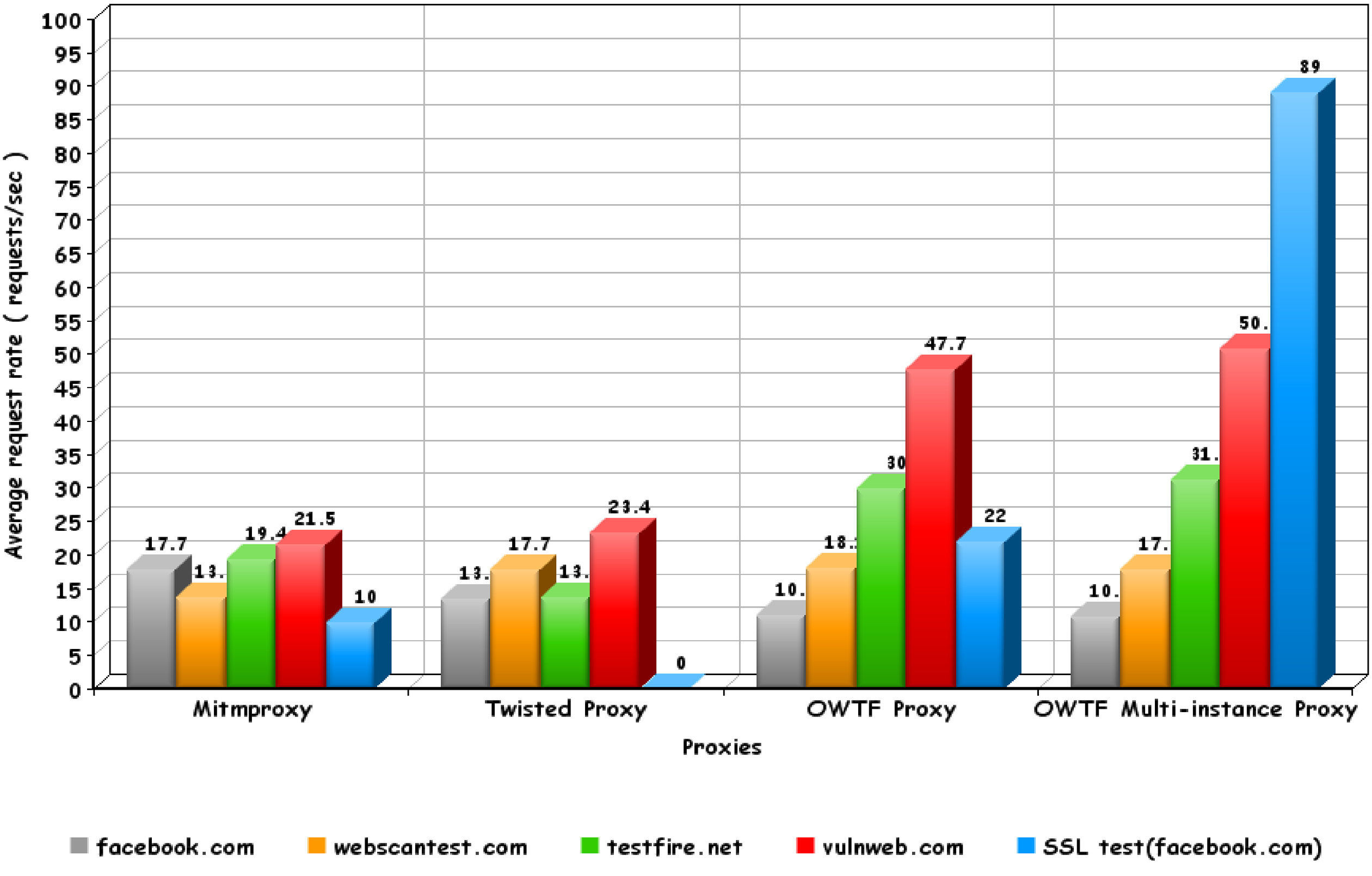
Proxy Cache is working ☺



Race-condition handling is working



Performace Benchmarks



Upcoming features

- **Improved grep plugins**: Run on all transactions
- **Request Throttling** based on target responsiveness

(i.e. avoid unintended DoS)

- **Intelligent request retries**

(i.e. ensure HTTP response retrieval where possible)

- **Cookie based authentication**

At proxy level = Ability to scan authenticated portions of a website.

- **Plug-n-Hack support**: Upcoming Mozilla standard

OWTF Testing Framework

by

Alessandro Fanio González

Dedicated Mentor: Andrés Morales Zamudio (@andresmz)
Co-mentor: Abraham Aranguren

Testing Framework Agenda

- Importance of testing
- Testing framework goals
- Pre-implementation research
- Development challenges
- Initial focus: Unit testing
- New focus: Functional testing
- Upcoming features

Importance of testing

- Improve code quality
- Ensure everything works as expected
- Prevent unintentional bugs:

While developing new features or fixing other bugs

- Provide stability to the project

Testing Framework Goals

- **Writing OWTF tests = As easy as possible**
- **Ensure OWTF integrity after code changes**:

  1. *Automated tests to verify OWTF modules behave as expected (**unit tests**)*

  2. *Automated tests to verify OWTF security test output is as expected (**functional tests**)*

Pre-implementation research

Goals: ← Determine best starting point

1. Select best testing/mocking library ← for **unit tests**
2. Select best mock web server ← for **functional tests**

Tests:

1. Feature-set comparison among many mocking libraries
2. Reuse of Bharadwaj's research (for mock web server)

Results:

1. Best mock library for OWTF = **Flexmock**
2. Best mock web server for OWTF = **Tornado**

Development Challenges

- Understand internal OWTF components
- Extend the testing library to complete features
- Make the testing framework easy to use:

Generate classes and methods dynamically, using metaclasses and introspection

- Fix broken tests due to fast-moving codebase

*Due to initial **unit testing** focus*

Initial focus: Unit testing

Important metric for unit testing = **code coverage**

Test coverage:

Number of executed lines of code after running all tests

When we run the entire test suite:

1. An HTML code coverage report is generated

2. Lines executed x file can be viewed in the report

Current OWTF code coverage = **58%**

New focus: Functional testing

Unit test approach	**Functional test approach**
Pro: Fast	**Con**: Slower
Pro: Isolated	**Con**: Not isolated
Pro: Code coverage metrics (i.e. are we at 100% or not?)	**Con**: No code coverage metrics
Con: Harder to write (i.e. you kinda have to love/know TDD ☺)	**Pro**: Easier to write (i.e. closer to command-line usage)
Con: Code dependent (i.e. refactoring = broken test)	**Pro**: Code independent (i.e. refactoring != broken test)
Con: Difficult to create tests for security edge cases (i.e. unusual web server behaviour)	**Pro**: Easier to create tests for security edge cases (i.e. unusual web server behaviour)
Con: **Can't** find bugs due to third-party tools/incompatibilities	**Pro**: **Will** find bugs due to third-party tools/incompatibilities

Demo 6: A testing example

Functional testing:

- Set the web server to return a custom robots.txt file, and start the server
- Write tests (almost) as if you were using OWTF from the command line: run the Spiders_Robots_and_Crawlers plugin
- Assert that the URLs contained in robots.txt are in the OWTF output

Unit testing:

- Show code coverage report from initial project focus

Upcoming features

Functional tests for:

1. **web** plugins: *OWASP Testing Guide coverage*
2. **net** and **aux** plugins: *PTES coverage*

- **Automated Continuous Integration**:

Run tests automatically after each commit

Questions?

OWASP Testing Guide

with

OWASP OWTF

Context consideration:

Case 1 → robots.txt Not Found

...should Google index a site like this?

Or should robots.txt exist and be like this?

User-agent: *

Disallow: /

Spiders, Robots, and Crawlers (OWASP-IG-001)

Case 1 → robots.txt Not Found - **Semi passive**

- **Direct** request for robots.txt
- Without visiting entries

Spiders, Robots, and Crawlers (OWASP-IG-001) robots.txt Analysis

Results: passive **semi_passive**

Spiders Robots And Crawlers - ***SEMI PASSIVE***

PLUGIN	START	END	RUNTIME	OUTPUT FIL
semi_passive/Spiders_Robots_and_Crawlers@OWASP-IG-001.py	08/02/2012-13:44	08/02/2012-13:44	0s, 869ms	Browse

NOTES

Edit

http://demo.testfire.net/robots.txt was NOT found

HTTP TRANSACTIONS

REQUEST

See Transaction 3 (0s, 863ms) Site F R H B

```
GET /robots.txt HTTP/1.1
Accept-Encoding: identity
Host: demo.testfire.net
Connection: close
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:6.0) Gecko/20100101 Firefo
```

RESPONSE

```
404 Not Found
Content-Length: 1635
Content-Type: text/html
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Date: Wed, 08 Feb 2012 14:26:06 GMT
Connection: close


<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR
<HTML><HEAD><TITLE>The page cannot be found</TITLE>
<META HTTP-EQUIV="Content-Type" Content="text/html; charset=Windows-125
<STYLE type="text/css">
  BODY { font: 8pt/12pt verdana }
```

Spiders, Robots, and Crawlers (OWASP-IG-001)

Case 2 → robots.txt Found – **Passive**

- **Indirect** Stats, Downloaded txt file for review, “Open All in Tabs”

Spiders Robots And Crawlers - ***PASSIVE***

PLUGIN	START	END	RUNTIME	OUTPUT FILES
passive/Spiders_Robots_and_Crawlers@OWASP-IG-001.py	08/02/2012-13:37	08/02/2012-13:37	2s, 384ms	Browse

NOTES

Edit

Passive Analysis Results:

Analysis via tool.motoricerca.info

Online Resources:

Analysis via tool.motoricerca.info

robots.txt via anonymouse.org

Raw regexp processing:

robots.txt was found. 16 lines: 0 Allowed, 14 Disallowed, 0 Sitemap.

Saved to: owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Spiders_Robots_And_Crawlers/passive/robots1.txt

Disallowed Entries: Open All In Tabs

/administrator/

/cache/

/components/

Spiders, Robots, and Crawlers (OWASP-IG-001)

OWTF HTML Filter challenge: Embedding of untrusted third party HTML
Defence layers:
1) HTML Filter: Open source challenge
Filter 6 unchallenged since 04/02/2012, Can you hack it? ☺
http://blog.7-a.org/2012/01/embedding-untrusted-html-xss-challenge.html
2) HTML 5 sanboxed iframe
3) Storage in another directory = cannot access OWTF Review in localStorage

New Robots.txt Syntax Checker: a validator for robots.txt files

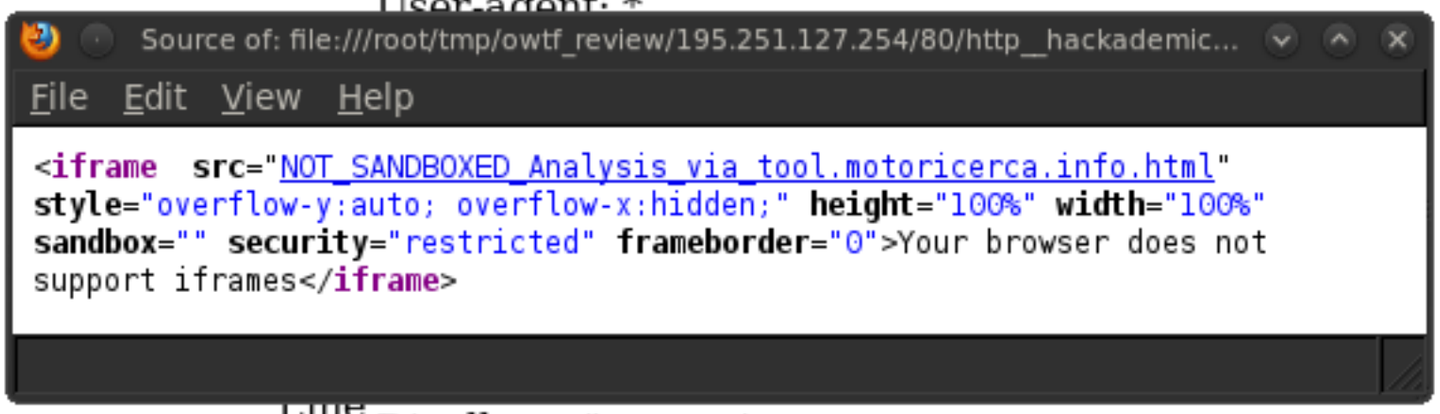
Analyzing file http://hackademic1.teilar.gr/robots.txt

No errors found in this robots.txt file

Hide empty and comments lines:

The following block of code DISALLOWS the crawling of the following files and directories: /administrator/ /cache/ /components/ /images/ /includes/ /installation/ /language/ /libraries/ /media/ /modules/ /plugins/ /templates/ /tmp/ /xmlrpc/ to all spiders/robots.

Line User-agent: *

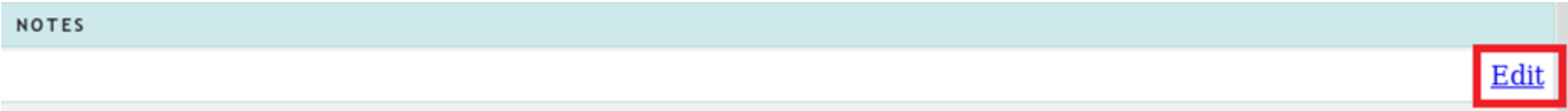


Line 5 Disallow: /images/

Spiders, Robots, and Crawlers (OWASP-IG-001)

Start reporting!: Take your notes with fancy formatting

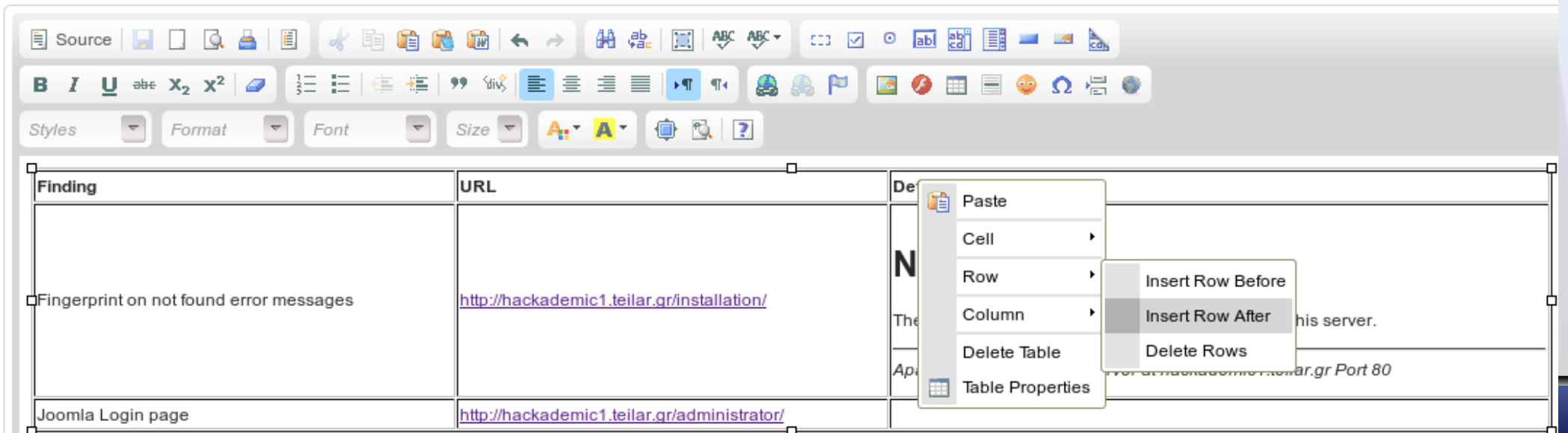
Step 1 – Click the “Edit” link



Step 2 – Start documenting findings + Ensure preview is ok

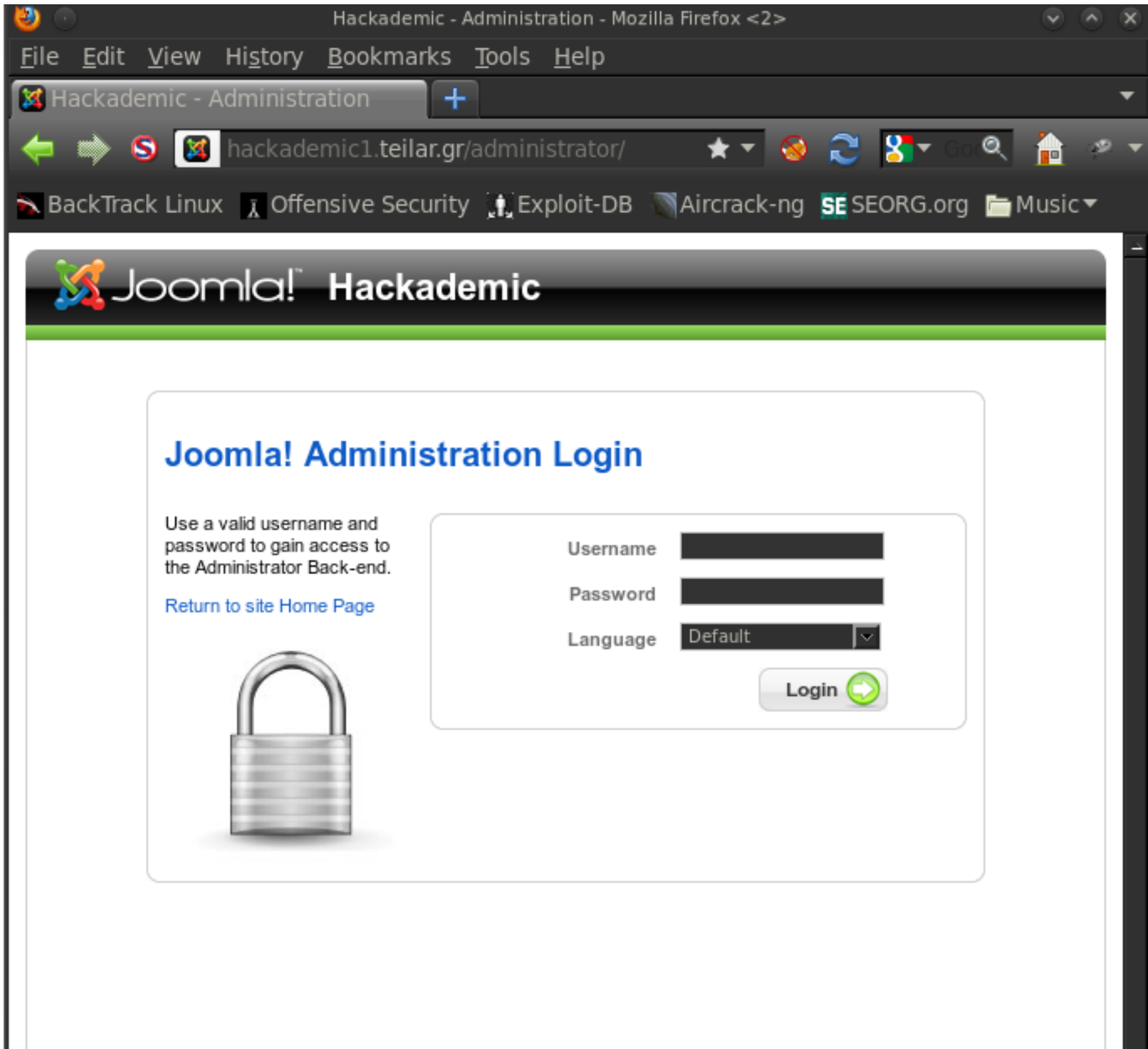
NOTES

Finding	URL	Detail
Fingerprint on not found error messages	http://hackademic1.teilar.gr/installation/	**Not Found** The requested URL /installation/ was not found on this server. --- *Apache/2.2.17 (Fedora) Server at hackademic1.teilar.gr Port 80*
Joomla Login page	http://hackademic1.teilar.gr/administrator/	



Start reporting!: Paste PoC screenshots

Joomla Login page	http://hackademic1.teilar.gr/administrator/



Spiders, Robots, and Crawlers (OWASP-IG-001)

The magic bar ;) – Useful to generate the **human** report later

Report for target: http://hackademic1.teilar.gr

Statistics | Passed Tests | **Findings** | Not Rated

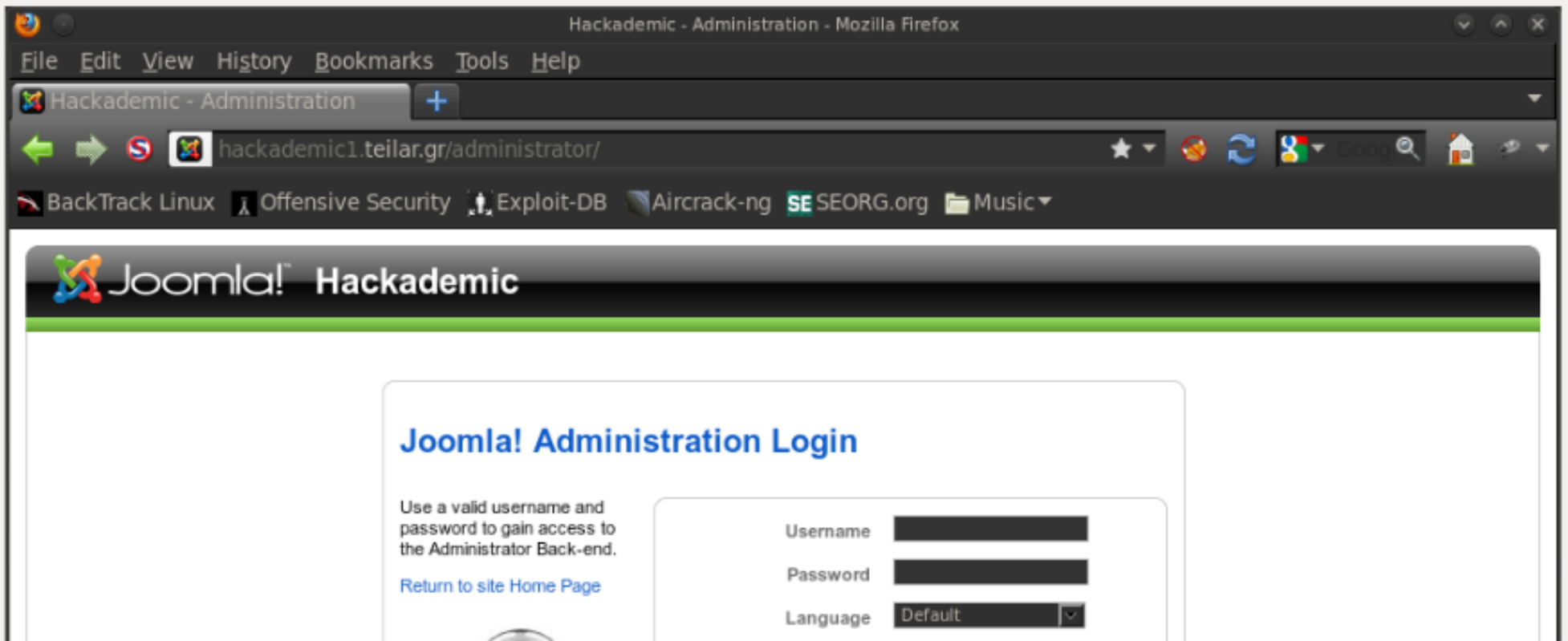
Findings

1. **High Severity**
   - ***Cross Site Flashing (OWASP-DV-004) - High Severity***

     No notes found for any plugin under this category

2. **Medium Severity**
   - ***Spiders, Robots, and Crawlers (OWASP-IG-001) - Medium Severity***

     A Joomla administrator login URL was found at: http://hackademic1.teilar.gr/administrator/



Search engine discovery/reconnaissance (OWASP-IG-002)

Passive Plugin

Step 1- Browse output files to review the full raw tool output:

START	END	RUNTIME	OUTPUT FILES
08/02/2012-13:37	08/02/2012-13:37	2s, 384ms	Browse

Step 2 – Review tools run by the passive Search engine discovery plugin:

MetaSploit_search_email_collector.txt	4 KB	08/02/2012	13:40:02
TheHarvester.txt	6 KB	08/02/2012	13:39:04
goohost_Google_search_Email.txt		08/02/2012	13:40:07
goohost_Google_search_Host.txt		08/02/2012	13:40:05
goohost_Google_search_IP.txt	1 KB	08/02/2012	13:40:06
goohost_email_check.txt		08/02/2012	13:40:06
goohost_host_check.txt		08/02/2012	13:40:03
metasploit_emails.txt	1 KB	08/02/2012	13:40:02

Was your favourite tool not run?

Tell OWTF to run your tools on: **owtf_dir/profiles/resources/default.cfg** (backup first

Search engine discovery/reconnaissance (OWASP-IG-002)

Tool output can also be reviewed via clicking through the OWTF report directly:

TEST COMMAND

```
cd owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Search_Engine_Discovery_Reconnaissance/passive/; cd /pentest/enumeration/theharvester ; python theHarvester.py -d teilar.gr -b all -v -f -h -l 1500
```

THEHARVESTER **OUTPUT (EXECUTION TIME: 1M, 20S, 906MS)**

```
*************************************
*TheHarvester Ver. 2.0 (reborn)     *
*Coded by Christian Martorella      *
*Edge-Security Research             *
*cmartorella@edge-security.com      *
*************************************


Full harvest..
[-] Searching in Google..
        Searching 0 results...
        Searching 100 results...
        Searching 200 results...
        Searching 300 results...
        Searching 400 results...
        Searching 500 results...
        Searching 600 results...
        Searching 700 results...
        Searching 800 results...
        Searching 900 results...
        Searching 1000 results...
        Searching 1100 results...
        Searching 1200 results...
        Searching 1300 results...
```

NOTE: Output longer than 25 lines, **Click here to see all output!**

Search engine discovery/reconnaissance (OWASP-IG-002)

```
****************************************
*TheHarvester Ver. 2.0 (reborn)     *
*Coded by Christian Martorella      *
*Edge-Security Research             *
*cmartorella@edge-security.com      *
****************************************

 [+] Emails found:
 ------------------
 jfrost@webappsecurity.com
 [+] Hosts found in search engines:
 ------------------------------------
 15.216.12.12:zero.webappsecurity.com
[+] Proposed SET
---------------
[]
[+] Virtual hosts:
==================
15.216.12.12:zero.webappsecurity.com
```

The Harvester:

- Emails
- Employee Names
- Subdomains
- Hostnames

http://www.edge-security.com/theHarvester.php

Search engine discovery/reconnaissance (OWASP-IG-002)

Metadata analysis:

- TODO: Integration with FOCA when CLI callable via wine (/cc @chemaalonso ☺)
- Implemented: Integration with Metagoofil

Search Engine Discovery Reconnaissance - ***SEMI PASSIVE***

PLUGIN	START	END	R
semi_passive/Search_engine_discovery_reconnaissance@OWASP-IG-002.py	08/02/2012-13:44	08/02/2012-13:47	2

NOTES

Edit

TEST COMMAND

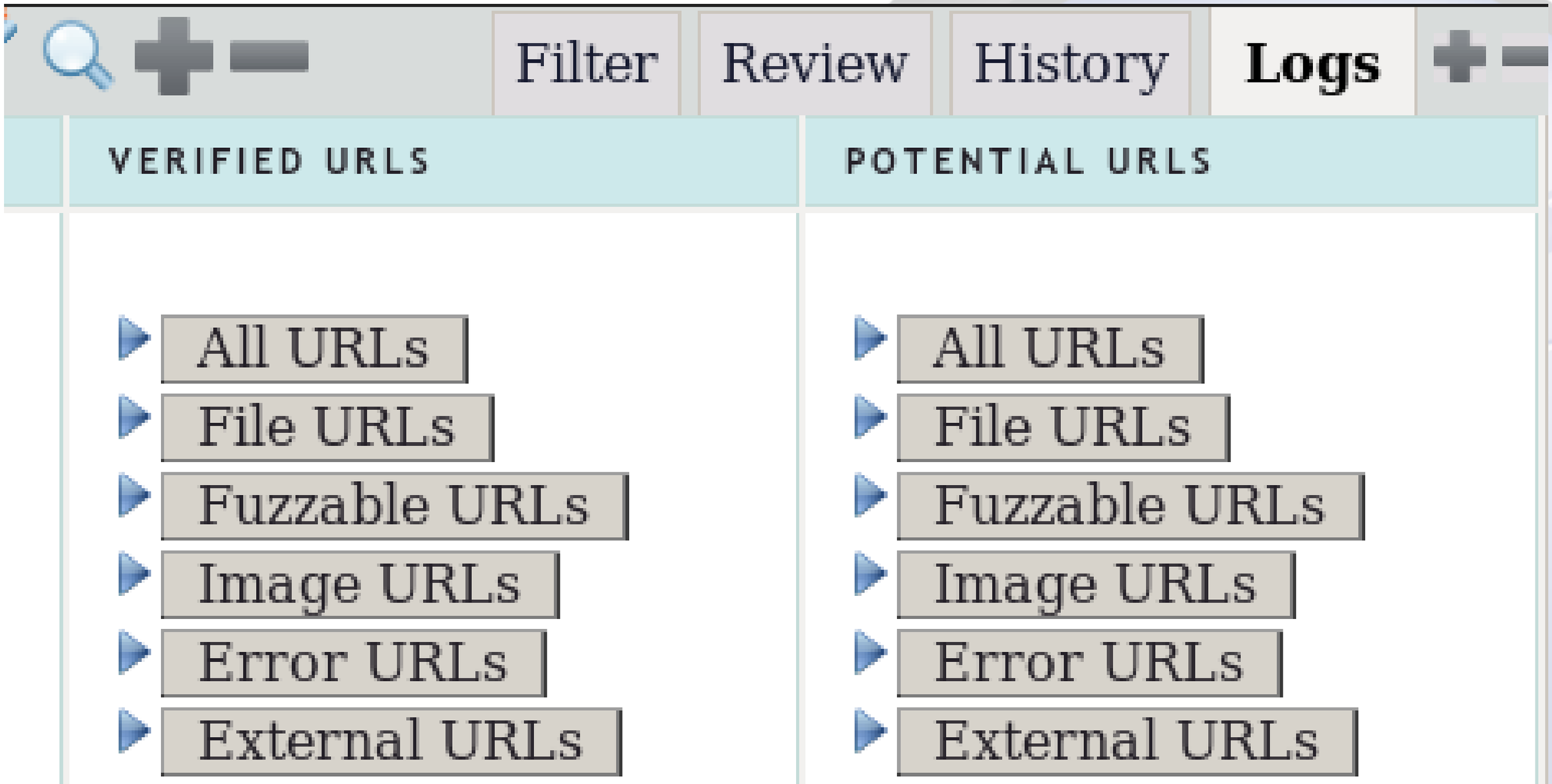
```
cd owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Search_Engine_Discovery_Reconnaissance
/semi_passive/; cd /pentest/enumeration/google/metagoofil ; python ./metagoofil.py -d hackademic1.teilar.gr -t
pdf,doc,xls,ppt,odp,ods,docx,xlsx,pptx -l 1500 -n 1500 -o /root/tmp/owtf_review/195.251.127.254
/80/http__hackademic1.teilar.gr/partial/Search_Engine_Discovery_Reconnaissance/semi_passive/ -f
metagoofil_report.html
```

METAGOOFIL OUTPUT (EXECUTION TIME: 2M, 49S, 581MS)

```
****************************************
* Metagoofil Ver 2.1 -                 *
* Christian Martorella                 *
* Edge-Security.com                    *
* cmartorella_at_edge-security.com     *
* Blackhat Arsenal Edition             *
****************************************

[-] Starting online search...

[-] Searching for pdf files, with a limit of 1500
```

Identify application entry points (OWASP-IG-003)

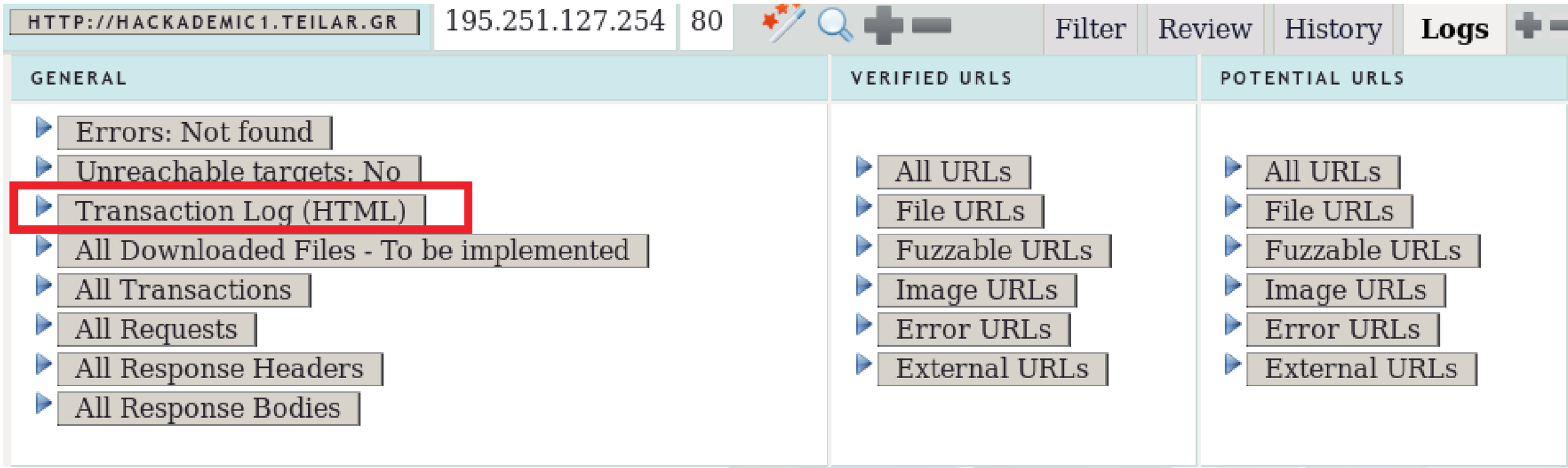
Inbound proxy not stable yet but all this happens automatically:
robots.txt entries added to “Potential URLs”
URLs found by tools are scraped + added to “Potential URLs”
During Active testing (later):
“Potential URLs” visited + added to “Verified URLs” + Transaction log



Identify application entry points (OWASP-IG-003)

All HTTP transactions logged by target in transaction log

Step 1 – Click on “Transaction Log”



Step 2 – Review transaction entries

SCOPE	LINKS	ID	SECONDS	TIME	STATUS	METHOD	URL
T	Site F R H B	3	0.4128510952	0s, 412ms	200 OK	GET	http://hackademic1.teilar.gr/robots.txt
T	Site F R H B	4	0.542858839035	0s, 542ms	200 OK	OPTIONS	http://hackademic1.teilar.gr

Identify application entry points (OWASP-IG-003)

Step 3 – Review raw transaction information (if desired)

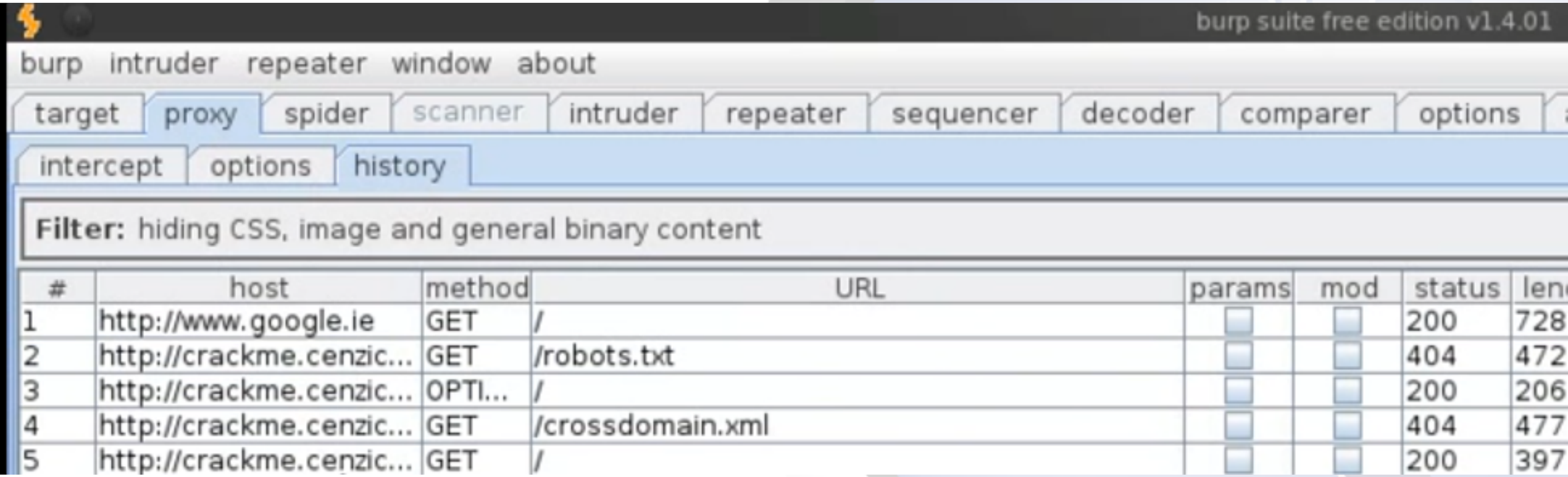
```
============================================== HTTP URL ==============================
http://hackademic1.teilar.gr/robots.txt
============================================== HTTP Request ==========================
GET /robots.txt HTTP/1.1
Accept-Encoding: identity
Host: hackademic1.teilar.gr
Connection: close
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:6.0) Gecko/20100101 Firefox/6.0

============================================== HTTP Response Headers =================
200 OK
Date: Wed, 08 Feb 2012 12:45:07 GMT
Server: Apache/2.2.17 (Fedora)
Last-Modified: Fri, 11 Mar 2011 22:29:48 GMT
ETag: "2610a3-130-49e3c7fe84f00"
Accept-Ranges: bytes
Content-Length: 304
Connection: close
Content-Type: text/plain; charset=UTF-8
============================================== HTTP Response Body ====================
User-agent: *
Disallow: /administrator/
Disallow: /cache/
Disallow: /components/
Disallow: /images/
Disallow: /includes/
```

Identify application entry points (OWASP-IG-003)

Step 1 - Make all direct OWTF requests go through Outbound Proxy:
Passes all entry points to the tactical fuzzer for analysis later

```
root@bt:/tmp# /root/owtf/owtf.py -f -x 127.0.0.1:8080 -t semi_passive http
://crackme.cenzic.com
```

Step 2 - Entry points can then also be analysed via tactical fuzzer:



Web Application Fingerprint (OWASP-IG-004)

Goal: What is that server running?

Manually verify request for fingerprint:

HTTP TRANSACTIONS

REQUEST

See Transaction 7 (0s, 446ms) Site F R H B

```
GET / HTTP/1.1
Accept-Encoding: identity
Host: hackademic1.teilar.gr
Connection: close
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:6.0) Gecko/20100101 F
```

RESPONSE

```
200 OK
Date: Wed, 08 Feb 2012 12:45:15 GMT
Server: Apache/2.2.17 (Fedora)
X-Powered-By: PHP/5.3.8
Set-Cookie: 26238b056396bb02ea2977b17de46c4c=pcar7hv2fejn92v14nfo
P3P: CP="NOI ADM DEV PSAi COM NAV OUR OTRo STP IND DEM"
Expires: Mon, 1 Jan 2001 00:00:00 GMT
Last-Modified: Wed, 08 Feb 2012 12:45:15 GMT
Cache-Control: no-store, no-cache, must-revalidate, post-check=0,
Pragma: no-cache
Content-Length: 7490
Connection: close
Content-Type: text/html; charset=utf-8
```

Web Application Fingerprint (OWASP-IG-004)

Whatweb integration with non-aggresive parameter (**semi passive detection**):

TEST COMMAND

```
cd owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Web_Application_Fingerprint
/semi_passive/; . /root/owtf_dev/scripts/setrubyenv.sh 1.8; /root/owtf_dev/tools/whatweb/whatweb-0.4.7/whatweb
--user-agent 'Mozilla/5.0 (X11; Linux i686; rv:6.0) Gecko/20100101 Firefox/6.0' --color=never --aggression 1
http://hackademic1.teilar.gr | sed "s/],/]\n/g"
```

WHATWEB SEMIPASSIVE CHECK (1 REQUEST) | OUTPUT (EXECUTION TIME: 6S, 749MS)

```
1.8
There are 2 choices for the alternative ruby (providing /usr/bin/ruby).

  Selection    Path                Priority   Status
  ------------------------------------------------------------
    0          /usr/bin/ruby1.8    500        auto mode
  * 1          /usr/bin/ruby1.8    500        manual mode
    2          /usr/bin/ruby1.9.2  400        manual mode

Press enter to keep the current choice[*], or type selection number: http://hackademic1.teilar.gr [200] PasswordField[passwd]
 MetaGenerator[Joomla! 1.5 - Open Source Content Management]
 HTTPServer[Fedora Linux][Apache/2.2.17 (Fedora)]
 Apache[2.2.17]
 IP[195.251.127.254]
 PHP[5.3.8]
 X-Powered-By[PHP/5.3.8]
 Joomla[1.5][com_content,com_user]
 Cookies[26238b056396bb02ea2977b17de46c4c]
 Title[Hackademic]
 probably Mambo[com_content,com_user]
 Country[GREECE][GR]
```

Web Application Fingerprint (OWASP-IG-004)

Fingerprint header analysis: Match stats

Web Application Fingerprint - ***SEMI PASSIVE***

PLUGIN	START	END	RUNTIME	OU
semi_passive/Web_Application_Fingerprint@OWASP-IG-004.py	08/02/2012-13:44	08/02/2012-13:44	7s, 679ms	

NOTES

Edit

Header Analysis Summary

LOG	See log
HTTP TRANSACTION STATS	5 out of 5 (100.0%) matched
ANALYSIS COMMAND	grep -lHiE "(Server|X-Powered-By|X-AspNet-Version|X-Runtime|X-Version|MicrosoftSharePointTeamServices): " owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_headers/scope_* | sed -e 's|owtf_review/195.251.127.254||g' -e 's|/response_headers/|/|g'

Web Application Fingerprint (OWASP-IG-004)

Convenient vulnerability search box (1 box per header found ☺):
Search All → Open all site searches in tabs

Header Value Analysis

NOTE: Only unique values per header are shown with a link to an example transaction

HEADER	VALUES
Server	Apache/2.2.17 (Fedora)
X-Powered-By	PHP/5.3.8
X-AspNet-Version	Not Found
X-Runtime	Not Found
X-Version	Not Found
MicrosoftSharePointTeamServices	Not Found

SEARCH FOR VULNERABILITIES: Apache/2.2.17 (Fedora) SEARCH ALL

NVD (High) | OSVDB (High) | BugTraq | ExploitDB | ExploitSearch (Exploits Only) | ExploitSearch (All) | NVD (All) | OSVDB (All)

SEARCH FOR VULNERABILITIES: PHP/5.3.8 SEARCH ALL

NVD (High) | OSVDB (High) | BugTraq | ExploitDB | ExploitSearch (Exploits Only) | ExploitSearch (All) | NVD (All) | OSVDB (All)

Web Application Fingerprint (OWASP-IG-004)

EXPLOIT DATABASE

blog exploit

Currently Archiving 15287 Exploits

Updated (CVE And Archive): Sun Nov 13 2011

HOME | BLOG | GHDB | FORUMS | ABOUT | REMOTE | LOCAL | WEB | DOS | SHELLCODE | PAPERS | SEARCH | SUBMIT

Search

Date	D	A	V	Description		Plat.	Author
2010-10-01		-		Microsoft IIS 6.0 ASP Stack Overflow (Stack Exhaustion) Denial of Service (MS10-065)	8951	windows	Kingcope
2010-08-14		-		Sports Accelerator Suite v2.0 (news_id) Remote SQL Injection Vulnerability	1789	php	LiquidWorm
2009-01-29		-		Full MSSQL Injection PWNage	1851		CWH Underground
2009-05-26		-		Microsoft IIS 6.0 WebDAV Remote Authentication Bypass Exploit (pl)	5209	windows	ka0x
2009-05-15		-		Microsoft IIS 6.0 WebDAV Remote Authentication Bypass Vulnerability	3153	windows	Kingcope
2008-11-28		-		Web Calendar System <= 3.40 (XSS/SQL) Multiple Remote Vulnerabilities	339	php	Bl@ckbe@rD
2008-11-17		-		Q-Shop 3.0 Remote XSS/SQL Injection Vulnerabilities	455	asp	Bl@ckbe@rD

Exploit DB - http://www.exploit-db.com

Web Application Fingerprint (OWASP-IG-004)

National Vulnerability Database
automating vulnerability management, security measurement, and compliance checking

Vulnerabilities | Checklists | 800-53 Controls | Product Dictionary | Impact Metrics | Data Feeds | Statistics

Home | SCAP | SCAP Validated Tools | SCAP Events | About | Contact | Vendor Comments

Mission and Overview

NVD is the U.S. government repository of standards based vulnerability management data. This data enables automation of vulnerability management, security measurement, and compliance (e.g. FISMA).

Resource Status

NVD contains:

48602 CVE Vulnerabilities
207 Checklists
221 US-CERT Alerts
2547 US-CERT Vuln Notes
6908 OVAL Queries
36734 CPE Names

Last updated: Thu Nov 17 23:23:21 EST 2011

CVE Publication rate:

Search Results (Refine Search)

There are **8** matching records. Displaying matches **1** through **8**.

CVE-2010-1256

TA10-159B

Summary: Unspecified vulnerability in Microsoft IIS 6.0, 7.0, and 7.5, when Extended Protection for Authentication is enabled, allows remote authenticated users to execute arbitrary code via unknown vectors related to "token checking" that trigger memory corruption, aka "IIS Authentication Memory Corruption Vulnerability."

Published: 06/08/2010

CVSS Severity: 8.5 (HIGH)

CVE-2009-3023

TA09-286A VU#276653

Summary: Buffer overflow in the FTP Service in Microsoft Internet Information Services (IIS) 5.0 through 6.0 allows remote authenticated users to execute arbitrary code via a crafted NLST (NAME LIST) command that uses wildcards, leading to memory corruption, aka "IIS FTP Service RCE and DoS Vulnerability."

Published: 08/31/2009

CVSS Severity: 9.3 (HIGH)

CVE-2009-1535

NVD - http://web.nvd.nist.gov - CVSS Score = High

Web Application Fingerprint (OWASP-IG-004)

OSVDB | Search OSVDB | Browse | Vendors | Project Info | Help OSVDB! | Sponsors | Ac

Quick Searches

General Search Go
Title Search Go
OSVDB ID Lookup Go
Vendor Search Go

Export Search Results

Alter Search Results: 2 : Show Descriptions Sort by

Search Query: cvss_score_to: 10 text_type: alltext vuln_title: IIS 6.0 cvss_

ID	Disc Date	Title
65216	2010-06-08	Microsoft IIS Extended Protection for Authentication Memory Corruption
568	2001-06-18	Microsoft IIS idq.dll IDA/IDQ ISAPI Remote Overflow

Show All Database IDS for this query

OSVDB - http://osvdb.org - CVSS Score = High

Web Application Fingerprint (OWASP-IG-004)

Microsoft-IIS/6.0 inurl:bid site:securityfocus.com

About 34 results (0.14 seconds)

Microsoft IIS Unicode Requests to WebDAV Multiple Authentication ...
www.securityfocus.com/**bid**/34993
15 May 2009 – Vulnerable: **Microsoft IIS 6.0** + Microsoft Windows Server 2003 Datacenter Edition + Microsoft Windows Server 2003 Datacenter Edition ...

Microsoft IIS ASP Remote Code Execution Vulnerability
www.securityfocus.com/**bid**/18858
11 Jul 2006 – Microsoft Windows 2000 Advanced Server SP1 Microsoft Windows 2000 Advanced Server **Microsoft IIS 6.0** + Microsoft Windows Server 2003 ...

http://www.securityfocus.com - Better on Google

Web Application Fingerprint (OWASP-IG-004)

EXPLOITSEARCH.NET

SEARCH TARGETED SEARCH STATS NAUGHTY LIST DONATIONS

BLOG ABOUT F.A.Q.

Search

Search Exploits Only

+1 152 Tweet 0 Share 2

View JSON results.

(**6.638** seconds)

ENTRY [METASPLOIT modules/exploits/windows/iis/ms01_026_dbldecode.rb] match rank: 100%

http://www.metasploit.com/modules/exploit/windows/iis/ms01_026_dbldecode

Microsoft IIS/PWS CGI Filename Double Decode Command Execution

This module will execute an arbitrary payload on a Microsoft IIS installation that is vulnerable to the CGI double-decode vulnerability of 2001. NOTE: This module will leave a metasploit payload in the IIS scripts directory.

Exploits

- METASPLOIT modules/exploits/windows/iis/ms01_026_dbldecode.rb - [Search]

References

- BID 2708 - [Search]
- CVE-2001-0333 - [Search]
- MS01-026 - [Search]
- OSVDB 556 - [Search]

Initial Date Seen [2011-07-15 15:33:35]
Last Date Updated [2011-07-15 15:33:35]

Web Application Fingerprint (OWASP-IG-004)

Passive Fingerprint analysis

Web Application Fingerprint - ***PASSIVE***

PLUGIN	START	END	RUNTIME	OUTPUT FI
passive/Web_Application_Fingerprint@OWASP-IG-004.py	08/02/2012-13:37	08/02/2012-13:37	0s, 19ms	Browse

NOTES

Edit

SEARCH FOR VULNERABILITIES: SEARCH ALL

NVD (High) | OSVDB (High) | BugTraq | ExploitDB | ExploitSearch (Exploits Only) | ExploitSearch (All) | NVD (All) | OSVDB (All)

Online Resources: Open All In Tabs

- centralops.net TCP Query
- netcraft.com General
- netcraft.com Uptime
- whois.webhosting.info Banner
- www.shodanhq.com
- builtwith.com

Web Application Fingerprint (OWASP-IG-004)

Site report for zero.webappsecurity.com

Site	http://zero.webappsecurity.com	**Last reboot**	unknown Uptime graph
Domain	webappsecurity.com	**Netblock owner**	Hewlett-Packard Company
IP address	15.216.12.12	**Site rank**	143078
Country	US	**Nameserver**	ns1.inflow.net
Date first seen	April 2004	**DNS admin**	dnsadmin@inflow.net
Domain Registrar	markmonitor.com	**Reverse DNS**	zero-g1w2555g.austin.hp.com
Organisation	Hewlett-Packard Company, 3000 Hanover St., United States	**Nameserver Organisation**	SunGard Data Systems Inc., PO Box 459ATTN INFLOW.NET, care of Network Solutions, Drums, Panama
Check another site:	[redacted]	**Netcraft Site Report Gadget**	Google™ [More Netcraft Gadgets]

Hosting History

Netblock Owner	IP address	OS	Web Server	Last changed
3000 Hanover Street Palo Alto CA US 94304	15.216.12.12	Windows Server 2003	Microsoft-IIS/6.0	23-Jun-2011
3000 Hanover Street Palo Alto CA US 94304	15.216.12.12	Windows Server 2003	Microsoft-IIS/6.0	21-May-2011
3000 Hanover Street Palo Alto CA US 94304	15.216.12.12	Windows Server 2003	Microsoft-IIS/6.0	14-Feb-2011

NETCRAFT http://toolbar.netcraft.com - Passive banner grab,etc.

Web Application Fingerprint (OWASP-IG-004)

http://builtwith.com/7-a.org

Content Management Systems

Blogger
Blogger Usage Statistics - Websites using Blogger
Google Blogger Software.

JavaScript Libraries

Google JS Api
Google JS Api Usage Statistics - Websites using Google JS Api
Google Mashup Editor (GME) includes a JavaScript API that gives you direct access to the document object model (DOM) via JavaScript. This API lets you use JavaScript to perform operations that duplicate and go beyond the features available in the GME tags. The API is useful when you want to access an object in the application from a JavaScript expression. You can also use the API to perform CRUD operations (create, read, update, delete) on entries in a data feed.

Widgets

Google Plus One
Google Plus One Usage Statistics - Websites using Google Plus One
Google's answer to Facebook Like.

Lightbox
Lightbox Usage Statistics - Websites using Lightbox
Lightbox JS is a simple, unobtrusive script used to overlay images on the current page. It's a snap to setup and works on all modern browsers.

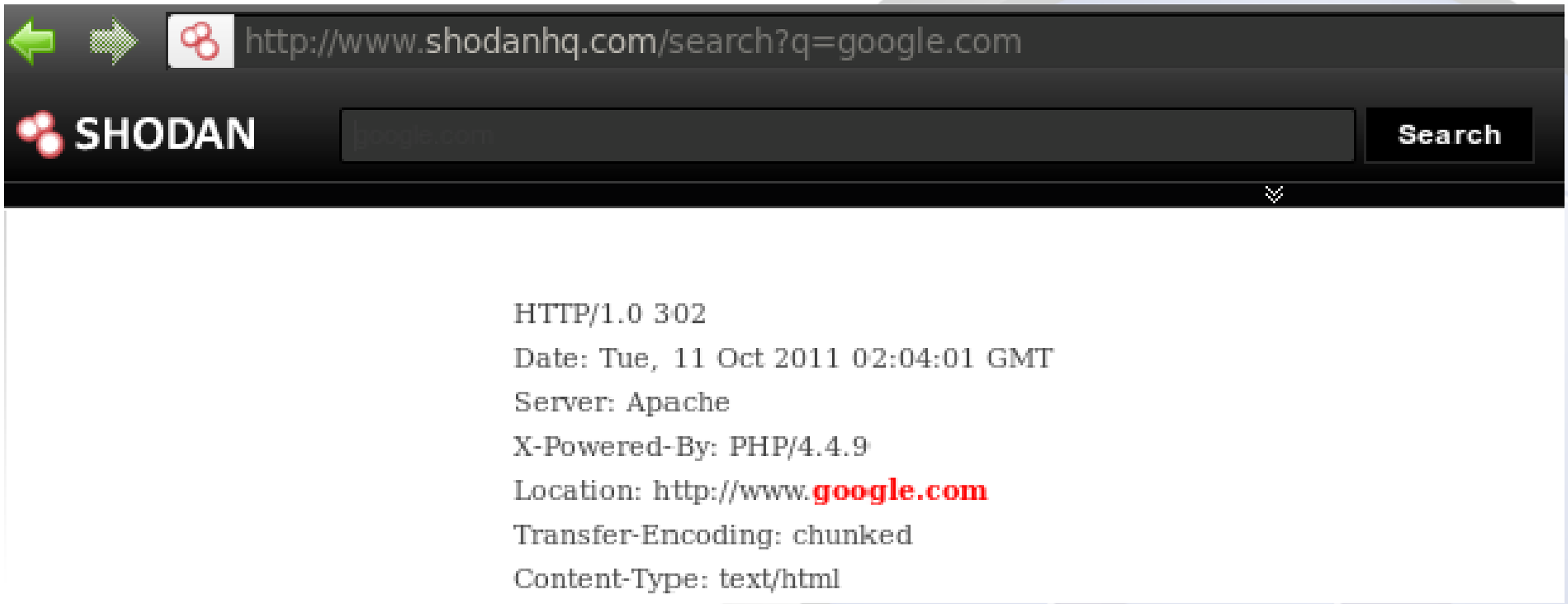
builtwith

- CMS
- Widgets
- Libraries
- etc

http://builtwith.com

Web Application Fingerprint (OWASP-IG-004)

Search in the headers without touching the site:



http://www.shodanhq.com/

Web Application Fingerprint (OWASP-IG-004)

Passive suggestions

- Prepare your test in a terminal window to hit “Enter” on “permission minute 1”

CMS Fingerprint - Potentially useful commands

All | **WordPress** | Joomla | Drupal | Mambo

WPSCAN PLUGIN ENUMERATION (WORDPRESS)

cd owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Web_Application_Fingerprint/passive/; ruby /root/owtf_dev/tools/wpscan/wpscan-1.1/wpscan.rb --url http://hackademic1.teilar.gr --enumerate p --threads 20

CMS EXPLORER PLUGIN ENUMERATION (WORDPRESS)

cd owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Web_Application_Fingerprint/passive/; cd /pentest/enumeration/web/cms-explorer; perl cms-explorer.pl -v 1 -url http://hackademic1.teilar.gr -type Wordpress

DIRBUSTER WORDPRESS ALL

cd owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Web_Application_Fingerprint/passive/; cd /pentest/web/dirbuster ; java -jar DirBuster-0.12.jar -u http://hackademic1.teilar.gr -t 20 -R -r '/root/tmp/owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Web_Application_Fingerprint/passive/dirbuster_report.txt' -l /root/owtf_dev/dictionaries/wp/dir_buster.all.wp.txt | grep -v "^java." | tr "\t" " " | grep -v "^ at" # Remove java exception garbage at the end

DIRBUSTER WORDPRESS PLUGINS

cd owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Web_Application_Fingerprint/passive/; cd /pentest/web/dirbuster ; java -jar DirBuster-0.12.jar -u http://hackademic1.teilar.gr -t 20 -R -r '/root/tmp/owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/partial/Web_Application_Fingerprint/passive/dirbuster_report.txt' -l /root/owtf_dev/dictionaries/wp/dir_buster.wp_plugins.txt | grep -v "^java." | tr "\t" " " | grep -v "^ at" # Remove java exception garbage at the end

DIRBUSTER WORDPRESS THEMES

What else can be done with a fingerprint?

Environment replication

Download it .. Sometimes from project page ☺

drupal.org/node/3060/release?page=1

drupal 6.16

Posted by Gábor Hojtsy on *March 4, 2010 at 12:17am*

Download	Size	md5 hash
drupal-6.16.tar.gz	1.04 MB	bb27c1f90680b86df2c535b2d52e8021
drupal-6.16.zip	1.22 MB	0ce2cd42371625d69642f043525c1cb7

Official release from tag: 6.16
Last updated: December 24, 2010 - 22:08
View usage statistics for this release
The sixteenth maintenance and security release of the Drupal 6 series. Only fixes for security vulr committed. New features are only being added to the forthcoming Drupal 7.0 release.

This release fixes **security vulnerabilities**. Sites are **urged to upgrade immediately** after reac

- SA-CORE-2010-001 - Drupal Core - Multiple vulnerabilities

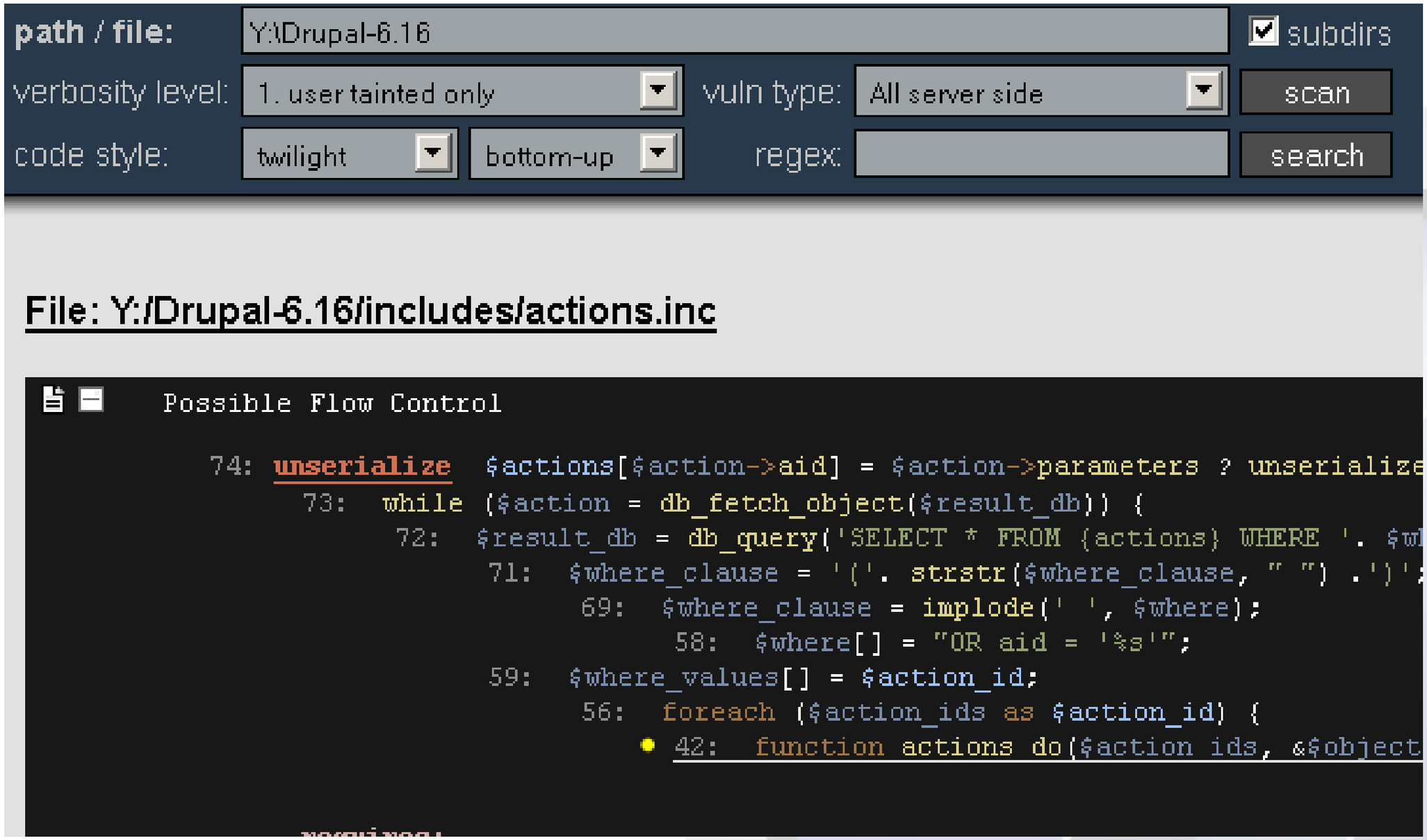
In addition to this security vulnerability, the following bugs have been fixed since the 6.15 release

- #673974 by sun: PHP notice when mass-unpublishing or deleting comments, and wrong form
- #424372 by mr.baileys, bombatower, Arancaytar: :: in .info files caused fatal error, use list of
- #370958 by Rob Loach, drewish, c960657, neilnz: some Adobe Flash MIME types were missing

Also check http://www.oldapps.com/, Google, etc.

Web Application Fingerprint (OWASP-IG-004)

Static Analyis, Fuzz, Try exploits, ..



RIPS for PHP: http://rips-scanner.sourceforge.net/
Yasca for most other (also PHP): http://www.scovetta.com/yasca.html

Questions?

Application Discovery (OWASP-IG-005)

Application Discovery - ***PASSIVE***

PLUGIN	START	END	RUNTIME	OUTPUT FILES
passive/Application_Discovery@OWASP-IG-005.py	08/02/2012-13:37	08/02/2012-13:37	0s, 15ms	Browse

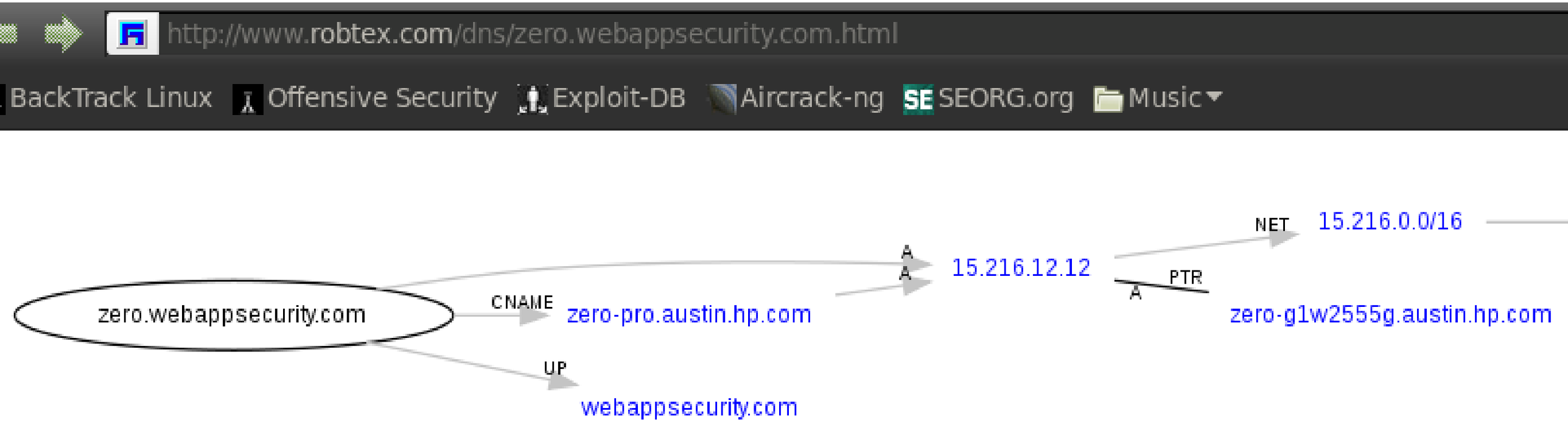
NOTES

Edit

Online Resources: Open All In Tabs

- Hurricane Electric TOP Domain DNS records
- Hurricane Electric Host Name DNS records
- whois.webhosting.info (Virtual Hosts)
- intodns.com
- www.robtext.com
- centralops.net TCP Query
- centralops.net Domain Dossier
- centralops.net AutoWhois
- centralops.net Ping
- centralops.net NsLookup
- dnsgoodies.com SMTP Open Relay
- dnsgoodies.com Spam DB Check
- dnsgoodies.com Abuse Lookup

Application Discovery (OWASP-IG-005)



IP numbers of host (1 item)

15.216.12.12

PTRs of IP numbers (1 item)

zero-g1w2555g.austin.hp.com

Host names sharing IP with A records (2 items)

zero-g1w2555g.austin.hp.com
zero-pro.austin.hp.com

CNAME (1 item)

zero-pro.austin.hp.com

http://www.robtex.com - Passive DNS Discovery

Application Discovery (OWASP-IG-005)

http://whois.domaintools.com/webapps

Whois Record | Site Profile | Registration | Server Stats | My Whois

Reverse Whois: **"Domain Administrator" owns about 416,674 other domains**

Email Search: hp.domains@hp.com is associated with about **3,108 domains**

hostmaster@hp.com is associated with about **1,414 domains**

Registrar History: **2 registrars**

NS History: **5 changes** on **2** unique name servers over **9** years.

IP History: **5 changes** on **4** unique IP addresses over **7** years.

Whois History: **45 records** have been archived **since 2004-04-01** .

Log In or Create a FREE account to start monitoring this domain name

```
Registrant:
        Domain Administrator
        Hewlett-Packard Company
        3000 Hanover St.
         Palo Alto CA 94304
        US
        hp.domains@hp.com  +1.8005247638 Fax: +1.6508522936
```

http://whois.domaintools.com

Application Discovery (OWASP-IG-005)

CentralOps.net Advanced online Internet utilities

Utilities

- Domain Dossier
- Domain Check
- Email Dossier
- Browser Mirror
- Ping
- Traceroute
- NsLookup
- AutoWhois
- TcpQuery
- AnalyzePath

Domain Dossier Investigate domains and IP addresses

domain or IP address: zero.webappsecurity.com

- [ ] domain whois record
- [ ] DNS records
- [ ] traceroute
- [ ] network whois record
- [x] service scan

go

user: anonymous
balance: 48 units
log in | account info

CentralOps.net

http://centralops.net

Application Discovery (OWASP-IG-005)

AutoWhois
TcpQuery
AnalyzePath

Service scan

FTP - 21

```
Error: TimedOut
```

SMTP - 25

```
Error: TimedOut
```

HTTP - 80

```
HTTP/1.1 302 Object moved
Connection: close
Date: Tue, 15 Nov 2011 08:57:10 GMT
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Location: banklogin.asp?serviceName=FreebankCaastA
AD_REFERRING_URL=http://www.Freebank.com
Content-Length: 263
Content-Type: text/html
Set-Cookie: ASPSESSIONIDAATCACCS=LMEKDKIAEPKAGOFAM
Cache-control: private
```

POP3 - 110

```
Error: TimedOut
```

IMAP - 143

```
Error: TimedOut
```

Testing for Error Code (OWASP-IG-006)

Has Google found error messages for you?

Testing For Error Code - *PASSIVE*

PLUGIN	START
passive/Testing_for_Error_Code@OWASP-IG-006.py	08/02/2012

NOTES

Online Resources: Open All In Tabs

- hexillion.com For Passive Verification Queries
- Google Search (Errors in title)
- Google Search (Errors in body)

Testing for Error Code (OWASP-IG-006)

"not found" OR denied OR error OR incorrect OR invalid OR unexpected C

Invalid Data Please try again.
zero.webappsecurity.com/rootlogin.asp
Invalid Data Please try again.

Invalid Data >'>"> Please try again.
zero.webappsecurity.com/rootlogin.asp?txtPassPhrase...
Invalid Data >'>"> Please try again.

The Test Page
zero.webappsecurity.com/test/test.html
LOGIC CHECKS WORKED. The welcome page · **Error** logs.

Check errors via Google Cache

Testing for SSL-TLS (OWASP-CM-001)

Testing for SSL-TLS (OWASP-CM-001) SSL Testing

Results: **passive** grep

Testing For Ssl-Tls - *PASSIVE*

PLUGIN	START	END
passive/Testing_for_SSL-TLS@OWASP-CM-001.py	08/02/2012-13:37	08/0

NOTES

Online Resources:

- www.ssllabs.com

Testing for SSL-TLS (OWASP-CM-001)

The link is generated with OWTF with that box ticked: Important!

QUALYS® SSL LABS

Home Qualys.com

You are here: Home > Projects > SSL Server Test

SSL Server Test

This free online service performs a deep analysis of the configuration of any SSL web server on the public In
**that the information you submit here is used only to provide you the service. We don't use the don
test results, and we never will.**

Domain name: Submit

Do not show the results on the boards

https://www.ssllabs.com/ssldb/analyze.html

Testing for SSL-TLS (OWASP-CM-001)

Pretty graphs to copy-paste to your OWTF report ☺

https://www.ssllabs.com/ssldb/analyze.html

Testing for SSL-TLS (OWASP-CM-001)

Do not forget about Strict-Transport-Security!
sslstrip chances decrease dramatically:
Only 1st time user visits the site!

Testing For Ssl-Tls - *GREP*

PLUGIN	START	END	RUNTIME	OUTPUT FILES
grep/Testing_for_SSL-TLS@OWASP-CM-001.py	09/02/2012-08:32	09/02/2012-08:32	0s, 35ms	Browse

NOTES

Edit

This plugin looks for server-side protection headers to enforce SSL

Header Analysis Summary

LOG	See log
HTTP TRANSACTION STATS	0 out of 197 (0.0%) matched
ANALYSIS COMMAND	grep -IHiE "(Strict-Transport-Security): " owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_headers/scope_* | sed -e 's|owtf_review/195.251.127.254||g' -e 's|/response_headers/|/|g'

Testing for SSL-TLS (OWASP-CM-001)

Not found example:

Header Value Analysis

NOTE: Only unique values per header are shown with a link to an example transaction

HEADER	VALUES
Strict-Transport-Security	Not Found

Found example:

Header Analysis Summary

LOG	See log
HTTP TRANSACTION STATS	2 out of 5 (40.0%) matched
ANALYSIS COMMAND	grep -IHiE "(Strict-Transport-Security): " owtf_review/173.194.65.84/443/https__accounts.google.com/transactions/response_headers/scope_* | sed -e 's|owtf_review/173.194.65.84||g' -e 's|/response_headers/|/|g'

Header Value Analysis

NOTE: Only unique values per header are shown with a link to an example transaction

HEADER	VALUES
Strict-Transport-Security	max-age=2592000; includeSubDomains

HTML content analysis: HTML Comments

PLUGIN	START	END	RUNTIME
grep/Application_Configuration_Management@OWASP-CM-004.py	02/03/2012-08:24	02/03/2012-08:24	0s, 874ms

NOTES
Edit

HTML Comments

STATS	• 17 Unique HTML Comments found • 52 out of 197 (26.0%) transactions matched
HTML COMMENTS	• Unique as TEXT • Unique as HTML • All as HTML
COMMAND	grep -IHiE "<!--" owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_bodies/scope_* | cut -f1 -d:|sort -u
LOG	See log

Efficient HTML content matches analysis

Step 1 - Click Unique as TEXT

Step 2 – Human Review of Unique matches

```
<!-- Start of StatCounter Code -->
<!-- End of StatCounter Code -->
<!--
 var prefix = 'm&#97;&#105;lt&#111;:';
 var suffix = '';
 var attribs = '';
 var path = 'hr' + 'ef' + '=';
 var addy55072 = '&#97;lp&#97;p&#97;n&#105;k' + '&#64;';
 addy55072 = addy55072 + '&#111;w&#97;sp' + '&#46;' + 'gr';
 document.write( '<a ' + path + '\'' + prefix + addy55072 + suffix + '\'' + attribs + '>' );
 document.write( addy55072 );
 document.write( '<\/a>' );
 //-->
<!--
 document.write( '<span style=\'display: none;\'>' );
 //-->
<!--
 document.write( '</' );
 document.write( 'span>' );
 //-->
```

Efficient HTML content matches analysis

Step 1 - Click Unique as HTML

Step 2 –Review Unique matches (click on links for sample match info)

Unique Matches

ID	Links	Match
10	Site F R H B	<!--[if lt IE 7.]> <link href="/templates/blackbearpro/css/ie6.css" rel="styleshe
10	Site F R H B	<!-- #content { padding-left:0px; width: 600px; } #container { background-ima /images/body.png); } -->
186	Site F R H B	<!--[if IE 7]> <link href="templates/khepri/css/ie7.css" rel="stylesheet" type="
186	Site F R H B	<!--[if lte IE 6]> <link href="templates/khepri/css/ie6.css" rel="stylesheet" type
192	Site F R H B	<!--[if lt IE 7.]> <link href="/gr/templates/blackbearpro/css/ie6.css" rel="styles <![endif]-->
192	Site F R H B	<!-- #content { padding-left:0px; width: 600px; } #container { background-ima /images/body.png); } -->

Want to see all? then click All as HTML

HTML content analysis: CSS and JavaScript Comments (/* */)

CSS/JS Comments

STATS	• 12 Unique CSS/JS Comments found • 3 out of 197 (1.0%) transactions matched
CSS/JS COMMENTS	• Unique as TEXT • Unique as HTML • All as HTML
COMMAND	grep -IHiE "/*" owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_bodies/scope_* | cut -f1 -d:|sort -u
LOG	See log

HTML content analysis: Single line JavaScript Comments (//)

Single Line JS Comments

STATS	• 0 Unique Single Line JS Comments found • 0 out of 197 (0.0%) transactions matched
SINGLE LINE JS COMMENTS	• Unique as TEXT • Unique as HTML • All as HTML
COMMAND	grep -IHiE "[^-:]//" owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_bodies/scope_* | cut -f1 -d:|sort -u
LOG	See log

HTML content analysis: PHP source code

Potential PHP source code

STATS	• 0 Unique Potential PHP source code found • 0 out of 197 (0.0%) transactions matched
POTENTIAL PHP SOURCE CODE	• Unique as TEXT • Unique as HTML • All as HTML
COMMAND	grep -IHiE "<?" owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_bodies/scope_* | cut -f1 -d:|sort -u
LOG	See log

HTML content analysis: ASP source code

Potential ASP source code

STATS	• 0 Unique Potential ASP source code found • 0 out of 197 (0.0%) transactions matched
POTENTIAL ASP SOURCE CODE	• Unique as TEXT • Unique as HTML • All as HTML
COMMAND	grep -IHiE "<%" owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_bodies/scope_* | cut -f1 -d:|sort -u
LOG	See log

Old, Backup and Unreferenced Files (OWASP-CM-006)

Old Backup And Unreferenced Files - *PASSIVE*

PLUGIN	START
passive/Old_Backup_and_Unreferenced_Files@OWASP-CM-006.py	08/02

NOTES

Online Resources: Open All In Tabs

- Google Search (Logs,Passwords,Juicy stuff)
- Google Search (Email files)
- Google Search (Source code, DB Dumps, Other)
- Google Search (Obscure extensions)
- Google Search (Directory Indexing)

Old, Backup and Unreferenced Files (OWASP-CM-006)

Old Backup And Unreferenced Files - *GREP*

PLUGIN	START	END
grep/Old_Backup_and_Unreferenced_Files@OWASP-CM-006.py	09/02/2012-08:32	09/0

NOTES

This plugin shows all URLs classified as 'Files' for review, there could be cool stuff here :)

All known File URLs in Scope: Open All In Tabs

- http://demo.testfire.net/admin/clients.xls
- http://demo.testfire.net/pr/communityannualreport.pdf

Questions?

Testing for Admin Interfaces (OWASP-CM-007)

Testing For Admin Interfaces - *PASSIVE*

PLUGIN	START
passive/Testing_for_Admin_Interfaces@OWASP-CM-007.py	08/02/2012-13:3

NOTES

Online Resources: Open All In Tabs

- Google Search (phpmyadmin,admin,backend,private,secret,login,logon)
- Google Search (username,login,password)

Testing for Admin Interfaces (OWASP-CM-007)

If you find an admin interface don't forget to ..
Google for default passwords:

sitefinity "by default" password

About 5,770 results (0.20 seconds)

Advanced sea

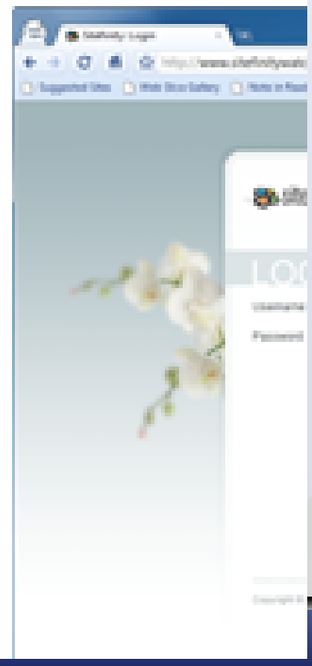
▸ **Sitefinity** Watch > How to secure **Sitefinity's** Administrative UI
www.**sitefinity**watch.com/.../How_to_secure_**Sitefinity**_rsquo_s_... - Cached
4 Mar 2010 – Users are then required to provide a valid username & **password** to gain entry to **Sitefinity**. **By default**, **Sitefinity's** administrative username ...

How to secure Sitefinity's Administrative UI

Thursday, March 04, 2010

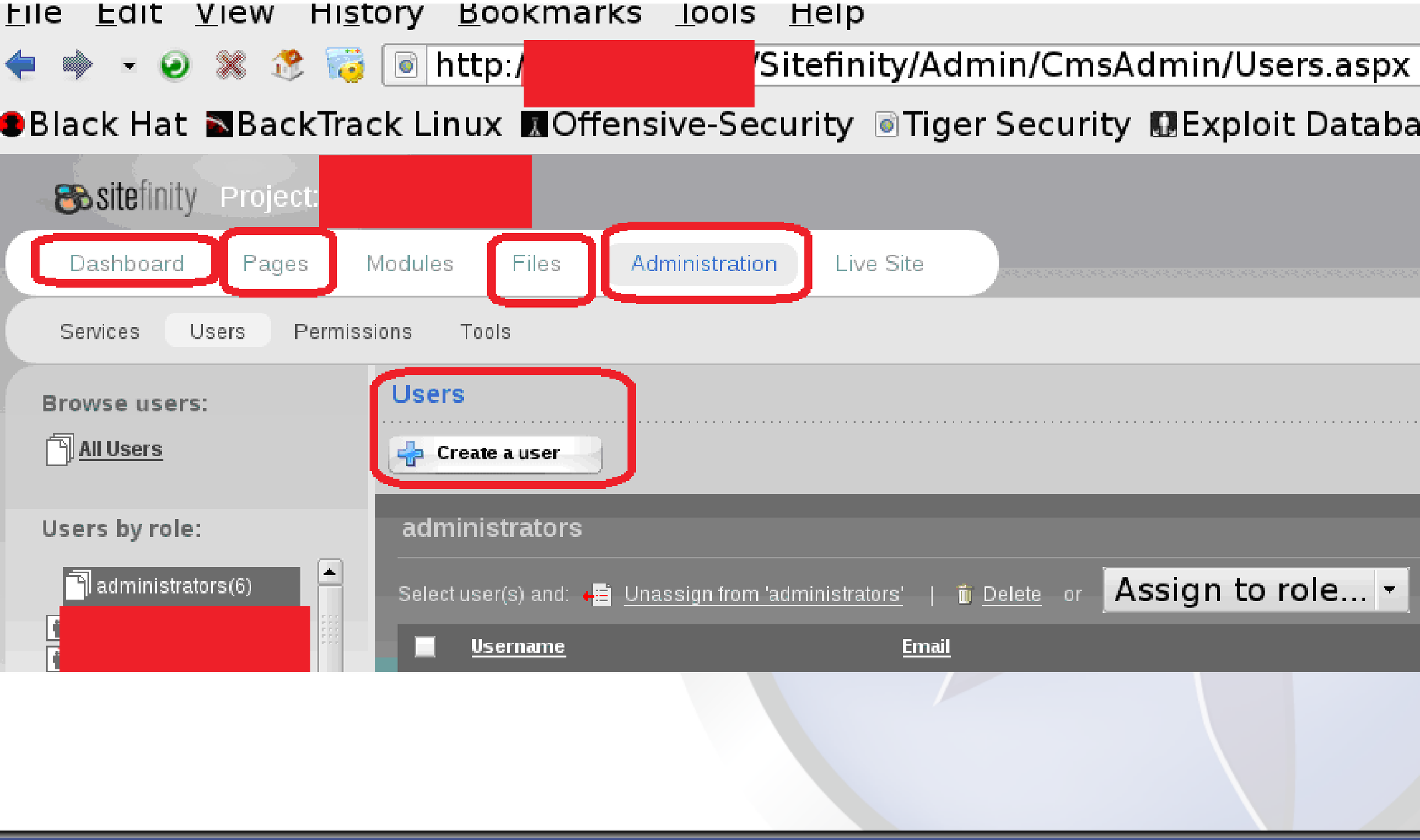
Sitefinity's Administrative Web Interface is accessed by adding **/Sitefinity** to the web site's URL. Users are then required to provide a valid username & password to gain entry to Sitefinity. By default, Sitefinity's administrative username is set to **admin**.

A few customers have expressed concern that this does not offer enough protection from malicious users or bots. If an attacker knows a web site is using Sitefinity then they also know the login URL and the **admin** username. The only thing that remains is the admin password.



Testing for Admin Interfaces (OWASP-CM-007)

Disclaimer: Permission is required for this



HTTP Methods and XST (OWASP-CM-008)

Http Methods And Xst - ***SEMI PASSIVE***

PLUGIN	START	END	RUNTIME	OUTPU
semi_passive/HTTP_Methods_and_XST@OWASP-CM-008.py	08/02/2012-13:44	08/02/2012-13:44	1s, 230ms	Bro

NOTES

Edit

HTTP TRANSACTIONS

REQUEST

See Transaction 4 (0s, 403ms) Site F R H B

```
OPTIONS / HTTP/1.1
Accept-Encoding: identity
Host: demo.testfire.net
Connection: close
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:6.0) Gecko/20100101 F
```

RESPONSE

```
200 OK
Allow: OPTIONS, TRACE, GET, HEAD
Content-Length: 0
Server: Microsoft-IIS/6.0
Public: OPTIONS, TRACE, GET, HEAD, POST
X-Powered-By: ASP.NET
Date: Wed, 08 Feb 2012 14:26:09 GMT
Connection: close
```

HTTP Methods and XST (OWASP-CM-008)

Http Methods And Xst - *PASSIVE*

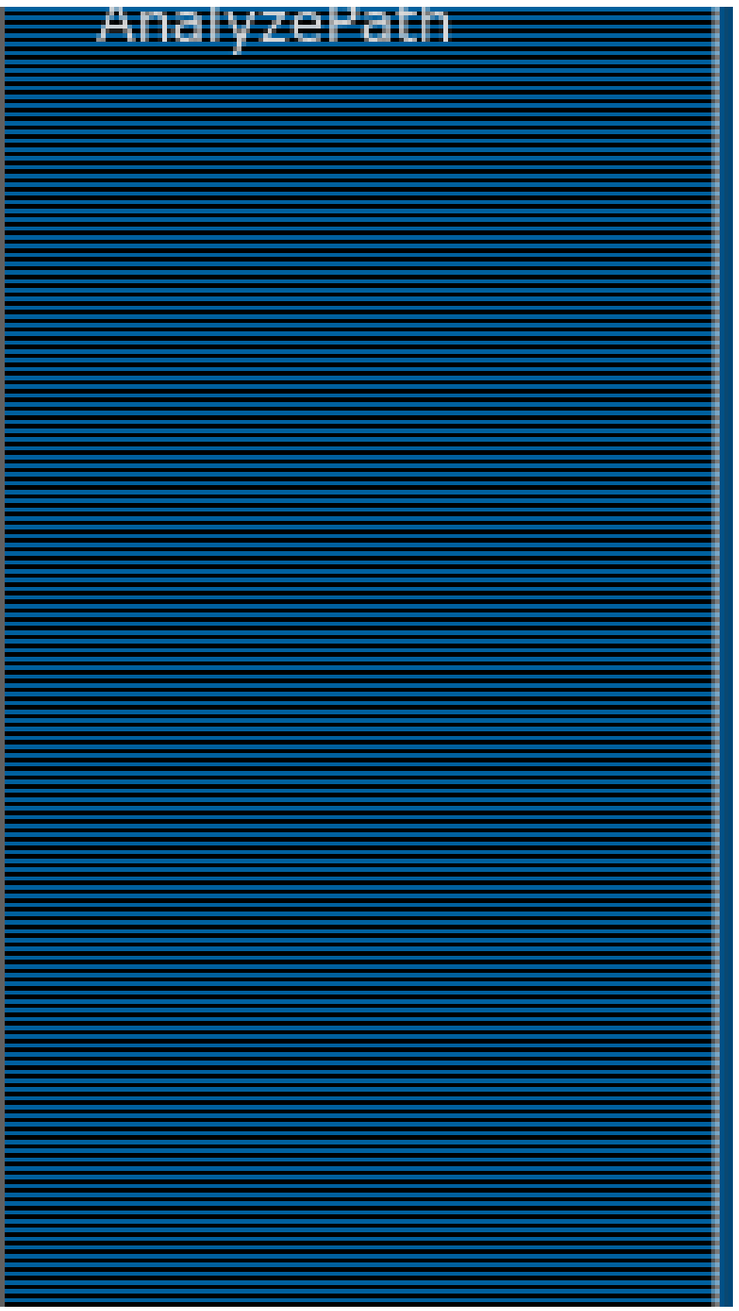
PLUGIN	START
passive/HTTP_Methods_and_XST@OWASP-CM-008.py	08/02/201

NOTES

Online Resources: Open All In Tabs

- hexillion.com OPTIONS check
- hexillion.com TRACE check

HTTP Methods and XST (OWASP-CM-008)



Querying **zero.webappsecurity.com [15.216.12.12]**...

[begin response]

```
HTTP/1.1 200 OK
Content-Length: 111
Content-Type: message/http
Server: Microsoft-IIS/6.0
X-Powered-By: ASP.NET
Date: Tue, 15 Nov 2011 08:36:26 GMT
Connection: close

TRACE / HTTP/1.0
Host: zero.webappsecurity.com
User-Agent: AspTcpQuery sample (http://www.hexillion.com/)
```

[end response]

Testing for Credentials Transport (OWASP-AT-001)

Is the login page on “http” instead of “https”?

Credentials Transport Over An Encrypted Channel - ***GREP***

PLUGIN	START
grep/Credentials_transport_over_an_encrypted_channel@OWASP-AT-001.py	02/03/2

NOTES

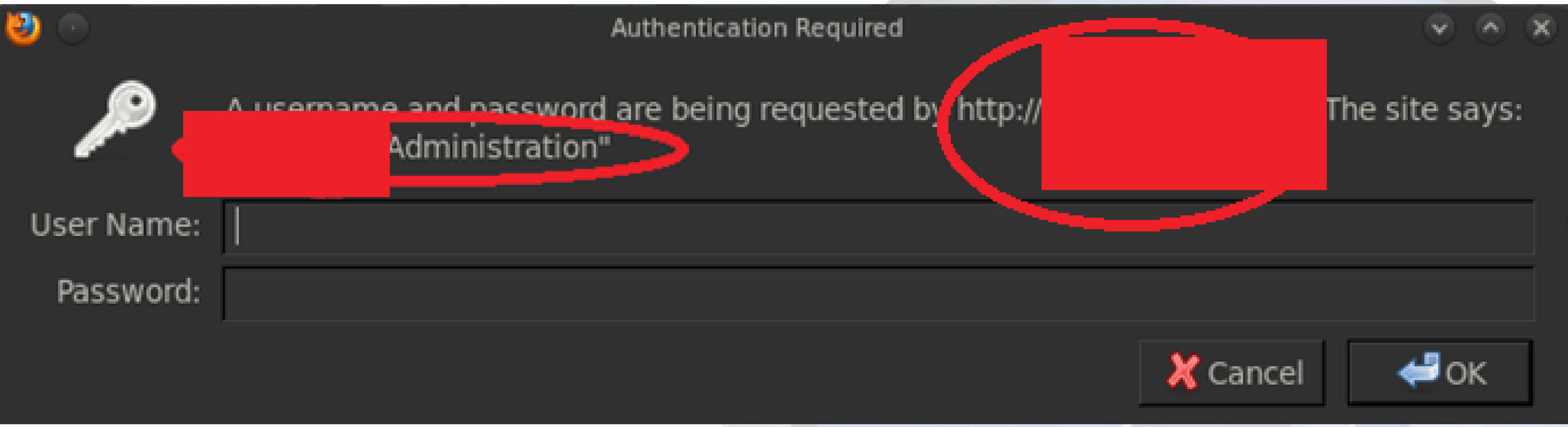
This plugin looks for password fields and then checks the URL (i.e. http vs. https)
Uniqueness in this case is performed via URL + password field
Total insecure matches: 53

Password fields

STATS	• 47 Unique Password fields found • 52 out of 197 (26.0%) transactions matched
PASSWORD FIELDS	• Unique as TEXT • Unique as HTML • All as HTML

Testing for Credentials Transport (OWASP-AT-001)

Pro Tip: When browsing the site manually ..
... **look** carefully at pop-ups like this:

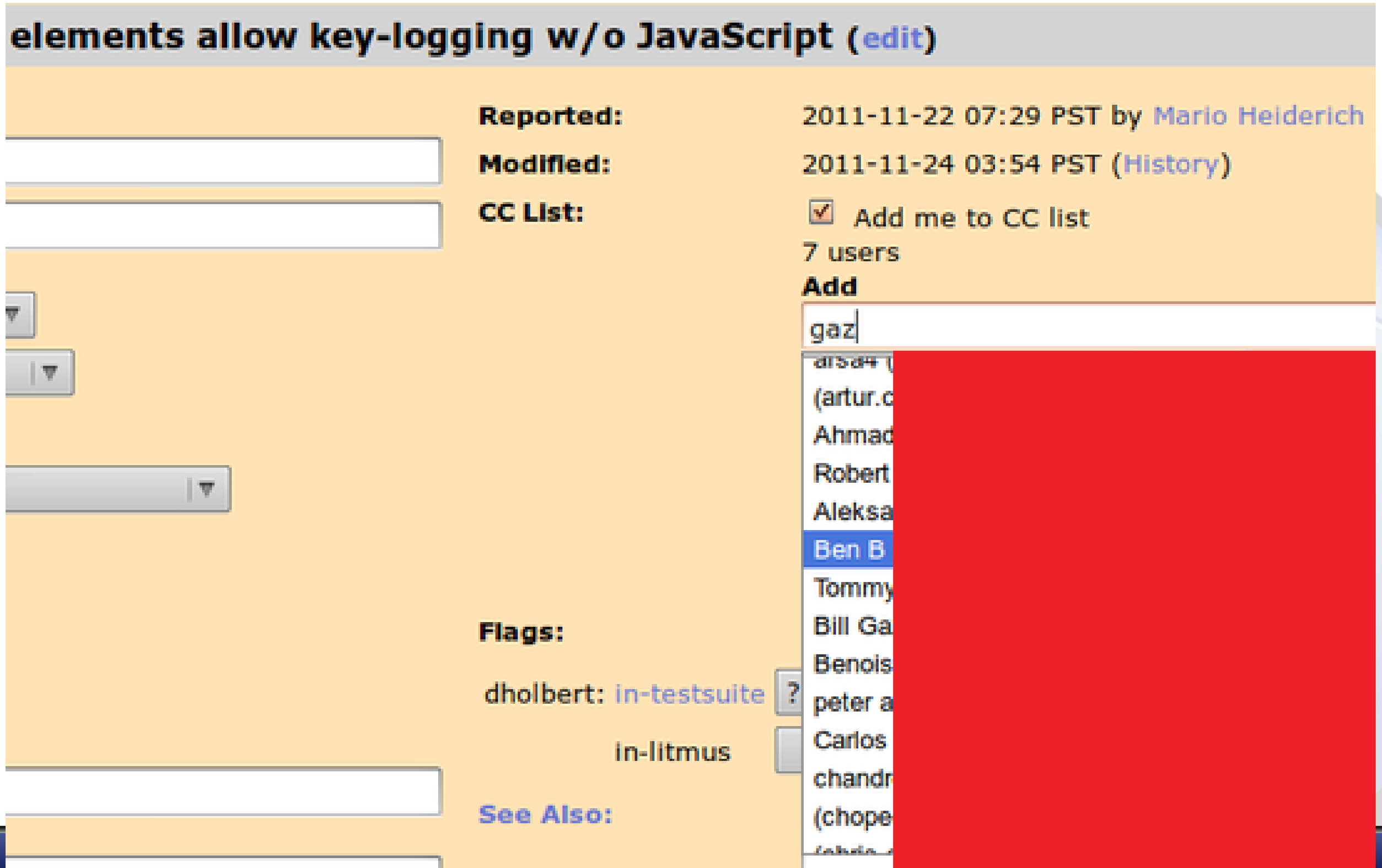


Consider (i.e. **prep the attack**):

Firesheep: http://codebutler.github.com/firesheep/
SSLStrip: https://github.com/moxie0/sslstrip

Testing for User Enumeration (OWASP-AT-002)

Mario was going to report a bug to Mozilla and found another!



Testing for User Enumeration (OWASP-AT-002)

Abuse user/member public search functions:

- Search for “” (nothing) or “a”, then “b”, ..
- Download all the data using 1) + pagination (if any)
- Merge the results into a CSV-like format
- Import + save as a spreadsheet
- Show the spreadsheet to your customer

2	TCGA-A6-2670	45	Sigmoid Colon	NO
3	TCGA-A6-2671	85	Sigmoid Colon	NO
4	TCGA-A6-2672	82	Transverse Colon	NO
5	TCGA-A6-2674	71	Sigmoid Colon	NO
6	TCGA-A6-2676	75	Cecum	NO
7	TCGA-A6-2677	68	Cecum	NO
8	TCGA-A6-2678	43	Transverse Colon	NO
9	TCGA-A6-2679	73	Ascending Colon	NO
10	TCGA-A6-2680	72	Hepatic Flexure	NO
11	TCGA-A6-2681	73	Cecum	NO
12	TCGA-A6-2682	70	Cecum	NO
13	TCGA-A6-2683	57	Ascending Colon	NO
14	TCGA-A6-2684	75	Cecum	NO
15	TCGA-A6-2685	48	Sigmoid Colon	NO
16	TCGA-A6-2686	81	Cecum	NO
17	TCGA-A6-3807	null	null	null
18	TCGA-A6-3808	73	Cecum	NO
19	TCGA-A6-3809	71	Transverse Colon	NO
20	TCGA-A6-3810	62	Sigmoid Colon	NO
21	TCGA-A6-4107	57	Ascending Colon	NO
22	TCGA-AA-3488	59	Sigmoid Colon	NO
23	TCGA-AA-3492	90	Ascending Colon	NO
24	TCGA-AA-3494	55	Sigmoid Colon	NO
25	TCGA-AA-3495	79	Hepatic Flexure	NO
26	TCGA-AA-3502	74	Transverse Colon	NO

Default or Guessable User Account (OWASP-AT-003)

Analyse the username(s) they gave you to test:

- Username based on numbers?

USER12345

- Username based on public info? (i.e. names, surnames, ..)

name.surname

- Default CMS user/pass?

Vulnerable Remember Password and Pwd Reset (OWASP-AT-006)

Part 1 – Remember Password: Autocomplete

Good	Bad
Via 1) <form … autocomplete="off"> Or Via 2) <input … autocomplete="off">	<form action="/user/login" method="post"> <input type="password" name="pass" />

Vulnerable Remember Password And Pwd Reset - *GREP*

PLUGIN	START
grep/Vulnerable_Remember_Password_and_Pwd_Reset@OWASP-AT-006.py	02/03/2012-10:46

NOTES

This plugin looks for password and form tags to review the autocomplete attribute

Autocomplete fields

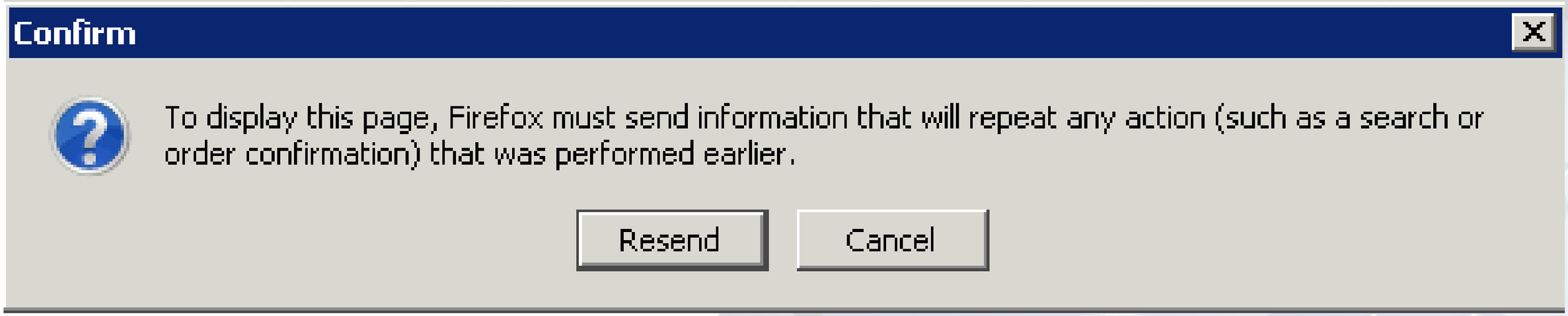
STATS	• 12 Unique Autocomplete fields found • 52 out of 197 (26.0%) transactions matched
AUTOCOMPLETE FIELDS	• Unique as TEXT • Unique as HTML • All as HTML
	grep -IHiE "type=.password" owtf_review/195.251.127.254

Vulnerable Remember Password and Pwd Reset (OWASP-AT-006)

Manual verification for password autocomplete (i.e. for the customer)

Easy “your grandma can do it” test:

1. Login
2. Logout
3. Click the browser Back button twice*
4. Can you login again –without typing the login or password- by re-



Can the user re-submit the login form via the back button?

* Until the login form submission

Other sensitive fields: Pentester manual verification

- Credit card fields
- Password hint fields
- Other

Part 2 - Password Reset forms

Manually look at the questions / fields in the password reset form

- Does it let you specify your email address?
- Is it based on public info? (name, surname, etc)
- Does it send an email to a potentially dead email address you can register? (i.e. hotmail.com)

Logout and Browser Cache Management (OWASP-AT-007)

Goal: Is Caching of sensitive info allowed?

Manual verification steps: “your grandma can do it” ☺ (need login):

1. Login
2. Logout
3. Click the browser Back button
4. Do you see logged in content or a this page has expired error / the login page?

Manual analysis tools:

- Commands: curl –i http://target.com
- Proxy: Burp, ZAP, WebScarab, etc
- Browser Plugins:

LiveHTTPHeaders

Firebug

https://addons.mozilla.org/en-US/firefox/addon/live-http-headers/

https://addons.mozilla.org/en-US/firefox/addon/firebug/

Logout and Browser Cache Management (OWASP-AT-007)

HTTP/1.1 headers

Good	Bad
Cache-Control: no-cache	Cache-control: private

HTTP/1.0 headers

Good	Bad
Pragma: no-cache Expires: <past date or illegal (e.g. 0)>	Pragma: private Expires: <way too far in the future>

The world

Good	Bad
https://accounts.google.com	No caching headers = caching allowed
Cache-control: no-cache, no-store Pragma: no-cache Expires: Mon, 01-Jan-1990 00:00:00 GMT	HTTP/1.1 200 OK Date: Tue, 09 Aug 2011 13:38:43 GMT Server: .... X-Powered-By: .... Connection: close Content-Type: text/html; charset=UTF-8

Logout and Browser Cache Management (OWASP-AT-007)

Logout And Browser Cache Management - ***GREP***

PLUGIN	START	END	RUNTIME
grep/Logout_and_Browser_Cache_Management@OWASP-AT-007.py	02/03/2012-10:46	02/03/2012-10:46	0s, 323m

NOTES
Edit

This plugin looks for server-side protection headers and tags against cache snooping

Header Analysis Summary

LOG	See log
HTTP TRANSACTION STATS	53 out of 197 (26.0%) matched
ANALYSIS COMMAND	grep -IHiE "(Cache-Control|Pragma|Expires): " owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_headers/scope_* | sed -e 's|owtf_review/195.251.127.254||g' -e 's|/response_headers/|/|g'

Header Value Analysis

NOTE: Only unique values per header are shown with a link to an example transaction

HEADER	VALUES
Cache-Control	no-store, no-cache, must-revalidate, post-check=0, pre-check=0
Pragma	no-cache
Expires	Mon, 1 Jan 2001 00:00:00 GMT

Logout and Browser Cache Management (OWASP-AT-007)

Repeat for Meta tags

Good	Bad
<META HTTP-EQUIV="Cache-Control" CONTENT="no-cache">	<META HTTP-EQUIV="Cache-Control" CONTENT=“private">

Cache Control Meta Tags

STATS	• 0 Unique Cache Control Meta Tags found • 0 out of 197 (0.0%) transactions matched
CACHE CONTROL META TAGS	• Unique as TEXT • Unique as HTML • All as HTML
COMMAND	grep -IHiE "<META.*?HTTP-EQUIV" owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_bodies/scope_* | cut -f1 -d:|sort -u
LOG	See log

Testing for Captcha (OWASP-AT-008)

Step 1 – Find CAPTCHAs: Passive search

Testing For Captcha - *PASSIVE*

PLUGIN	START	END
passive/Testing_for_Captcha@OWASP-AT-008.py	08/02/2012-13:37	08/02/

NOTES

Online Resources:

- Google Search (captcha, security code)

Testing for Captcha (OWASP-AT-008)

Offline Manual analysis:

- Download image and try to break it
- Are CAPTCHAs reused?
- Is a hash or token passed? (Good algorithm? Predictable?)
- Look for vulns on CAPTCHA version

CAPTCHA breaking tools

PWNtcha - captcha decoder - http://caca.zoy.org/wiki/PWNtcha

Captcha Breaker - http://churchturing.org/captcha-dist/

PLUGIN	START	END	RUNTIME	OUTPUT FILES
external/Testing_for_Captcha@OWASP-AT-008.py	03/03/2012-04:54	03/03/2012-04:54	0s, 79ms	Browse

NOTES

Edit

SEARCH FOR VULNERABILITIES: SEARCH ALL

NVD (High) | OSVDB (High) | BugTraq | ExploitDB | ExploitSearch (Exploits Only) | ExploitSearch (All) | NVD (All) | OSVDB (All)

Tools: Open All In Tabs

- PWNtcha - captcha decoder
- Captcha Breaker

Session Management Schema (OWASP-SM-001)

Manually Examine cookies for weaknesses offline

Base64 Encoding (!= Encryption ☺)	**Decoded value**
MTkyLjE2OC4xMDAuMTpvd2FzcHVzZXI 6cGFzc3dvcmQ6MTU6NTg=	owaspuser:192.168.100.1: a7656fafe94dae72b1e148767014841 2

Session Management Schema - ***EXTERNAL***

PLUGIN	START	END	RU
external/Session_Management_Schema@OWASP-SM-001.py	03/03/2012-07:15	03/03/2012-07:15	0s

NOTES

Edit

Online Resources: Open All In Tabs

- Gareth Hayes' HackVertor
- Raul Siles' (Taddong) F5 BIG IP Cookie Decoder

Questions?

Session Management Schema (OWASP-SM-001)

Charsets | Decode | Encode | Encrypt | Exec | Hacker | Hash | Math | SQLi | Str

Natural language conversion

Convert this to hex then octal

Convert

You are not logged in. You can still view everyone's public tags but you need to re

Input 100 100

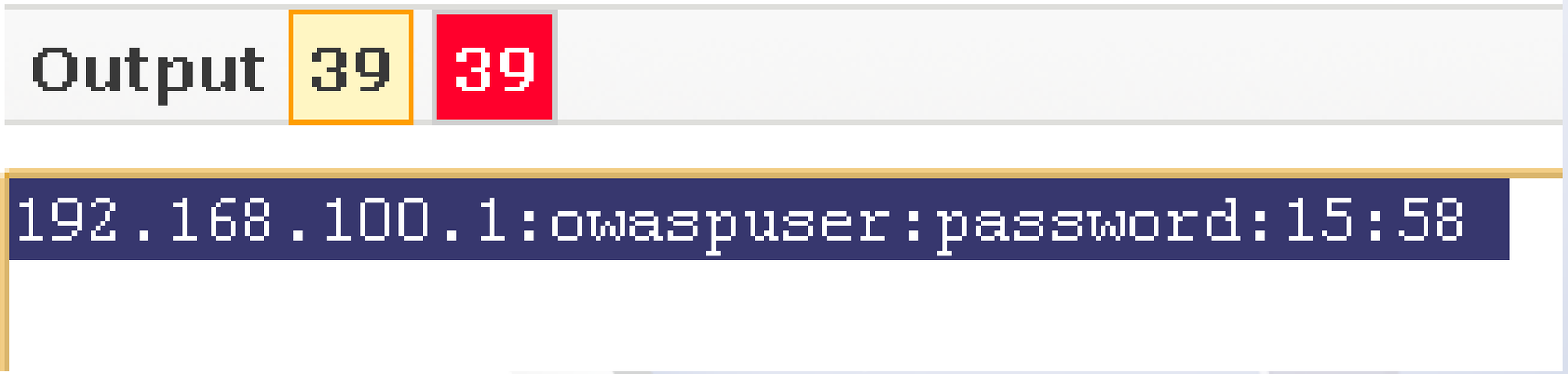
<@auto_decode_repeat_0>MTkyLjE2OC4xMDAuMTpvd2FzcHVzZXI6cGFzc3dvcmQ6MTU6NTg=<@/auto_decode_repeat_0>

http://hackvertor.co.uk/public

Session Management Schema (OWASP-SM-001)

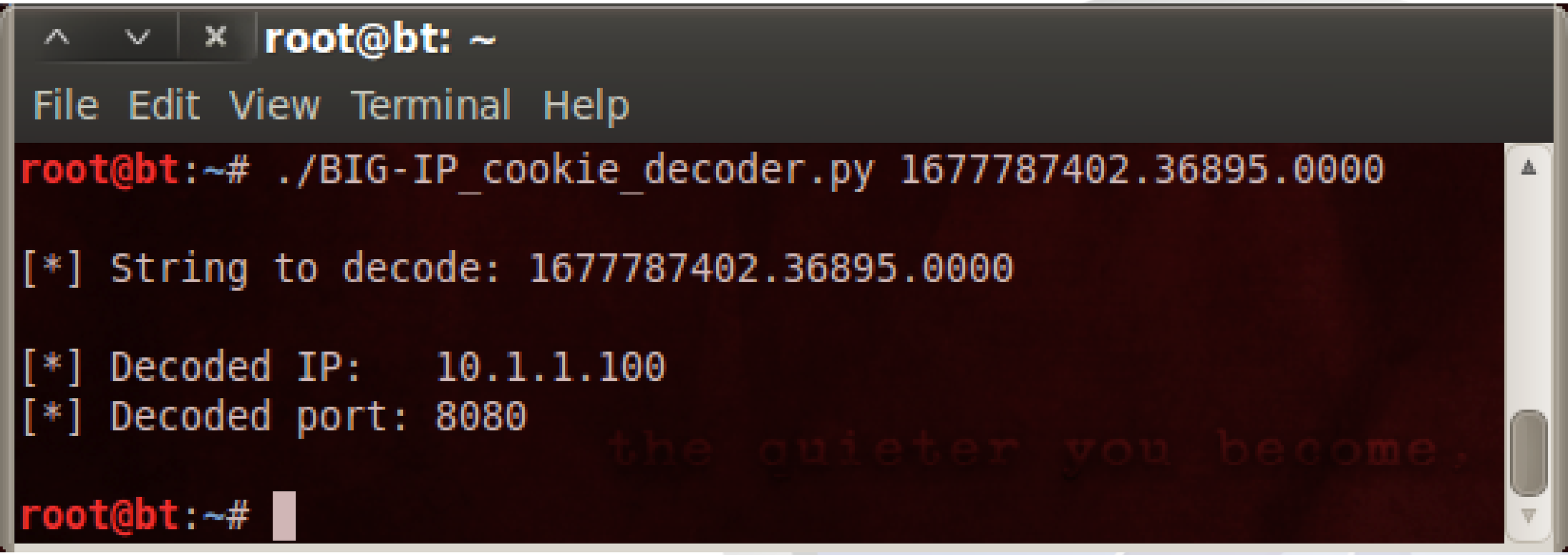
Lots of decode options, including:

- auto_decode
- auto_decode_repeat
- d_base64
- etc.



Session Management Schema (OWASP-SM-001)

F5 BIG-IP Cookie decoder:



http://blog.taddong.com/2011/12/cookie-decoder-f5-big-ip.html

Cookies Attributes (OWASP-SM-002)

- **Secure**: not set= session cookie leaked= pwned
- **HttpOnly**: not set = cookies stealable via JS
- **Domain**: set properly
- **Expires**: set reasonably
- **Path**: set to the right /sub-application
- 1 session cookie that works is enough ..

Name	Expires	HttpOnly	Security
⊞ SPRING_SECURITY_REMEMBER_ME_COOKIE	Thu 15 S		
⊞ JSESSIONID	Session	HttpOnly	Secure

Cookies Attributes - ***GREP***

PLUGIN	START	END	RUNTIME	OUTPUT
grep/Cookies_attributes@OWASP-SM-002.py	02/03/2012-10:46	02/03/2012-10:46	0s, 52ms	Brows

NOTES
Edit

This plugin looks for cookie setting headers (TODO: Check vuln scanners' output!)

Header Analysis Summary

LOG	See log
HTTP TRANSACTION STATS	58 out of 197 (29.0%) matched

Cookies Attributes (OWASP-SM-002)

Header Value Analysis

NOTE: Only unique values per header are shown with a link to an example transaction

HEADER	VALUES
	7bf9911fab0c9735a81838a8466b569d=nao2mmgho6p9jisslen9v3t6o5; path=/
	26238b056396bb02ea2977b17de46c4c=3h20bvblbinnmrfti751kgmf94; path=/
	26238b056396bb02ea2977b17de46c4c=e5to3mpc56qdgfj61o9rlghfg3; path=/
	26238b056396bb02ea2977b17de46c4c=i4t79up0lp1kl4oihpa0n3uf20; path=/
	74d4eed8cbb936df5ee62291facacd8c=4k03b9r77mdrvhp7ukr23s0td5; path=/
	26238b056396bb02ea2977b17de46c4c=p9hf1fu9069pq9j56dcj465ra2; path=/

Cookie Attribute Analysis

COOKIE: 7BF9911FAB0C9735A81838A8466B569D

ATTRIBUTE	VALUE
Value	nao2mmgho6p9jisslen9v3t6o5
secure	**Not Found**
HttpOnly	**Not Found**
domain	**Not Found**
path	path=/
expires	**Not Found**

Session Fixation (OWASP-SM-003)

Manually check when verifying credentials during pre-engagement:
Login and analyse the Session ID cookie (i.e. PHPSESSID)

Good	Bad (normal + by default)
Before: 10a966616e8ed63f7a9b741f80e65e3c After: Nao2mxgho6p9jisslen9v3t6o5f943h	Before: 10a966616e8ed63f7a9b741f80e65e3c After: 10a966616e8ed63f7a9b741f80e65e3c

IMPORTANT: **You can also set the session ID via JavaScript (i.e. XSS)**

Exposed Session Variables (OWASP-SM-004)

Session ID:

- In URL
- In POST
- In HTML

Example from the field:
http://target.com/xxx/xyz.function?session_num=7785

Bypassing Authorization Schema (OWASP-AZ-002)

Look at unauthenticated cross-site requests:

http://other-site.com/user=3&report=4
Referer: site.com

Change ids in application: (ids you have permission for!)
http://site.com/view_doc=4

Reflected Cross Site Scripting (OWASP-DV-001)

Headers Enabling/Disabling Client-Side XSS filters:

- **X-XSS-Protection** (IE-Only)
- **X-Content-Security-Policy** (FF >= 4.0 + Chrome >= 13)

Header Analysis Summary

LOG	See log
HTTP TRANSACTION STATS	0 out of 197 (0.0%) matched
ANALYSIS COMMAND	grep -IHiE "(X-Content-Security-Policy|X-XSS-Protection): " owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_headers/scope_* | sed -e 's|owtf_review/195.251.127.254||g' -e 's|/response_headers/|/|g'

Header Value Analysis

NOTE: Only unique values per header are shown with a link to an example transaction

HEADER	VALUES
X-Content-Security-Policy	Not Found
X-XSS-Protection	Not Found

DOM-based Cross Site Scripting (OWASP-DV-003)

Review JavaScript code on the page:

```
<script>
document.write("Site is at: " + document.location.href + ".");
</script>
```

Sometimes active testing possible in your browser (no trip to server = not an attack = not logged):
http://target.com/...#vulnerable_param=xss

http://blog.mindedsecurity.com/2010/09/twitter-domxss-wrong-fix-and-something.html

SQL Injection (OWASP-DV-005)

Testing For Sql Injection - ***PASSIVE***

PLUGIN	START	END	RUNTIME
passive/Testing_for_SQL_Injection@OWASP-DV-005.py	08/02/2012-13:37	08/02/2012-13:37	0s, 5ms

NOTES

Edit

Online Resources:

Google Search (sql, error, syntax)

Did Google find SQLi for you?

sql OR syntax OR error site:zero.webappsecurity.com

7 results (0.11 seconds)

LSWEB General Access **Error** Log
zero.webappsecurity.com/**error**s/**error**s.log
File Format: Unrecognized - View as HTML
... Feb 21 11:10:58 2001] [**error**] [client 192.107.108.150] Premature end of script headers: /www/htdocs/depts/anth/discus/scripts/show.cgi [Wed Feb 21 11:10:58 ...

My **ERROR** - zero.webappsecurity.com (HP)
zero.webappsecurity.com/**error**.html
Error Diagnostic Information The welcome page.

SSI Injection (OWASP-DV-009)

<!--#exec cmd="/bin/ls /" -->
<!--#INCLUDE VIRTUAL="/web.config"-->

Testing For Ssi Injection - ***GREP***

PLUGIN	START	END	RUNTIME
grep/Testing_for_SSI_Injection@OWASP-DV-009.py	02/03/2012-10:46	02/03/2012-10:46	0s, 81ms

NOTES

Edit

Server Side Includes

STATS	• 0 Unique Server Side Includes found • 0 out of 197 (0.0%) transactions matched
SERVER SIDE INCLUDES	• Unique as TEXT • Unique as HTML • All as HTML
COMMAND	grep -IHiE "<!--#" owtf_review/195.251.127.254/80/http__hackademic1.teilar.gr/transactions/response_bodies/scope_* | cut -f1 -d:|sort -u
LOG	See log

DoS Failure to Release Resources (OWASP-DS-007)

1. Browse Site
2. Time requests
3. Get top X slowest requests
4. Slowest = Best DoS target

Dos Failure To Release Resources - ***GREP***

PLUGIN	START	END	R
grep/DoS_Failure_to_Release_Resources@OWASP-DS-007.py	02/03/2012-10:46	02/03/2012-10:46	0

NOTES

Edit

Top 10 slowest transactions

Hint: You can also sort by time in descending order on the Transaction log

HTTP TRANSACTIONS

REQUEST	RESPONSE
See Transaction 9 (0s, 435ms) Site F R H B DEBUG / HTTP/1.1 Accept-Encoding: identity Host: hackademic1.teilar.gr Command: start-debug Connection: close User-Agent: Mozilla/5.0 (X11; Linux i686; rv:6.0) Gecko/20	200 OK Date: Wed, 08 Feb 2012 13:08:51 GMT Server: Apache/2.2.17 (Fedora) X-Powered-By: PHP/5.3.8 Set-Cookie: 26238b056396bb02ea2977b17de46c4c=t2neuqkhoihdl P3P: CP="NOI ADM DEV PSAi COM NAV OUR OTRo STP IND DEM" Expires: Mon, 1 Jan 2001 00:00:00 GMT Last-Modified: Wed, 08 Feb 2012 13:08:52 GMT Cache-Control: no-store, no-cache, must-revalidate, post- Pragma: no-cache Content-Length: 7490 Connection: close Content-Type: text/html; charset=utf-8

WS Information Gathering (OWASP-WS-001)

Google searches: inurl:wsdl site:example.com

Public services search:
http://seekda.com/
http://www.wsindex.org/
http://www.soapclient.com/

Ws Information Gathering - ***PASSIVE***

PLUGIN	START	END
passive/WS_Information_Gathering@OWASP-WS-001.py	08/02/2012-13:37	08/02/201

NOTES

Edit

Online Resources: Open All In Tabs

- Google Search (Web Services)
- wsindex.org
- www.soapclient.com
- www.xmethods.net

Testing WSDL (OWASP-WS-002)

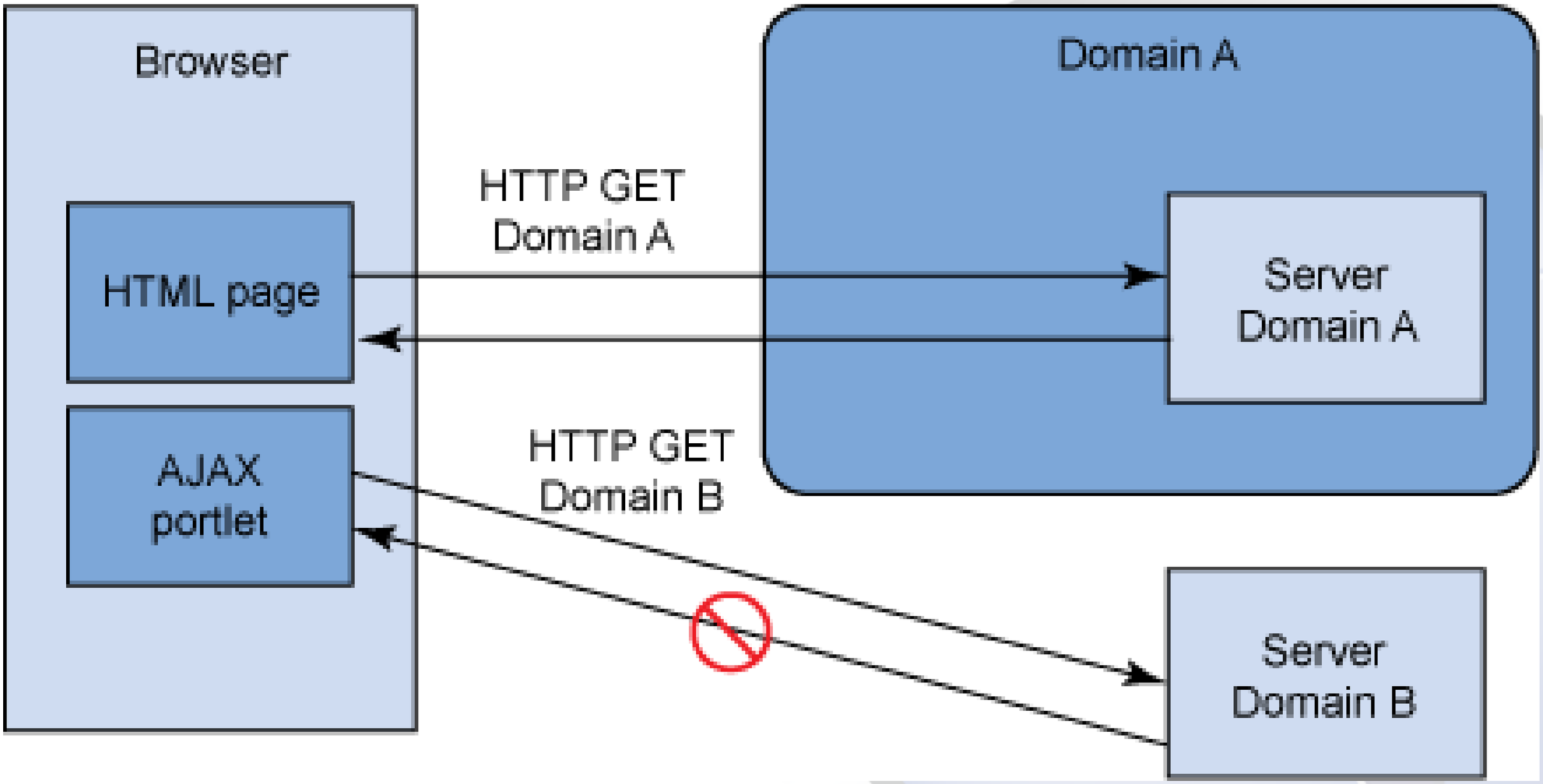
WSDL analysis

Sensitive methods in WSDL?
i.e. Download DB, Test DB, Get CC, etc.
http://www.example.com/ws/FindIP.asmx?WSDL

```
<wsdl:operation name="getCreditCard" parameterOrder="id">
    <wsdl:input message="impl:getCreditCardRequest"
name="getCreditCardRequest"/>
    <wsdl:output message="impl:getCreditCardResponse"
name="getCreditCardResponse"/>
</wsdl:operation>
```

Same Origin Policy (SOP) 101

1. Domain A's page **can send a request to** Domain B's page from Browser
2. BUT Domain A's page **cannot read** Domain B's page from Browser



http://www.ibm.com/developerworks/rational/library/09/rationalapplicationdeveloperportaltoolkit3/

Testing for CSRF (OWASP-SM-005)

- Request == Predictable → Pwned → "..**can send a request to** Domain B" (SOP)

CSRF Protection 101:

- Require long random token (99% hidden anti-CSRF token) → Not predictable
- Attacker cannot **read** the token from Domain B (SOP) → Domain B ignores **request**

Potentially Good	Bad
Anti-CSRF token present: Verify **with permission**	No anti-CSRF token

Testing For Csrf - *GREP*

PLUGIN	START	END	RUNTIME	OUTPUT
grep/Testing_for_CSRF@OWASP-SM-005.py	02/03/2012-10:46	02/03/2012-10:46	0s, 397ms	Brow

NOTES

Edit

Hidden fields

STATS	• 99 Unique Hidden fields found • 52 out of 197 (26.0%) transactions matched
HIDDEN FIELDS	• Unique as TEXT • Unique as HTML • All as HTML

Testing for WS Replay (OWASP-WS-007)

Similar to CSRF:
Is there an anti-replay token in the request?

Potentially Good	Bad
Anti-CSRF token present: Verify **with permission**	No anti-CSRF token

Cross Site Flashing (OWASP-DV-004)

1) Passive search for Flash/Silverlight files + policies:

Testing For Cross Site Flashing - ***PASSIVE***

PLUGIN	START	END	RU
passive/Testing_for_Cross_site_flashing@OWASP-DV-004.py	08/02/2012-13:37	08/02/2012-13:37	0s

NOTES

Edit

Online Resources: Open All In Tabs

- Google Search (SWF Files)
- Google Search (Silverlight Files)
- Google Search (crossdomain.xml,clientaccesspolicy.xml Files)

Flash file search:	Silverlight file search:
filetype:swf site:adobe.com	filetype:xap OR filetype:scr site:microsoft.com
About 12,300 results (0.13 seconds)	2 results (0.19 seconds)
[FLASH] Visual Components Print Controls Validators and Effects ... examples.adobe.com/flex3/componentexplorer/explorer.swf File Format: Shockwave Flash Visual Components. Print Controls. Validators and Formatters. Eff	Communications: Standby Continuous Replication in Exchange ... lab.technet.microsoft.com/en-us/magazine/2007.12.scr One of the most exciting features offered by Service Pack 1 is Standby Continuous Replication. Find out how this can help you improve uptime, limit data loss, ...

Cross Site Flashing (OWASP-DV-004)

Static analysis: Download + decompile Flash files

$ flare hello.swf

```
onClipEvent (enterFrame) {
  if (this._y > -254) {
    this._y += -3;
  }
  if (this._y > -254 and this._y < -175.3) {
    this._yscale -= 0;
    this._xscale -= 0;
  } else {
    if (this._y <= -157.7) {
      this._yscale -= 2;
```

Flare: http://www.nowrap.de/flare.html

Flasm (timelines, etc): http://www.nowrap.de/flasm.html

Cross Site Flashing (OWASP-DV-004)

Static analysis tools

Adobe SWF Investigator

http://labs.adobe.com/technologies/swfinvestigator

SWFScan

SWFScan: http://www.brothersoft.com/hp-swfscan-download-253747.html

Cross Site Flashing (OWASP-DV-004)

Active testing ☺

1) Trip to server = need permission

http://target.com/test.swf?xss=foo&xss2=bar

2) But … <u>your browser is yours</u>:

No trip to server = no permission needed

http://target.com/test.swf#?xss=foo&xss2=bar

Good news: Unlike DOM XSS, the # trick will always work for Flash Files

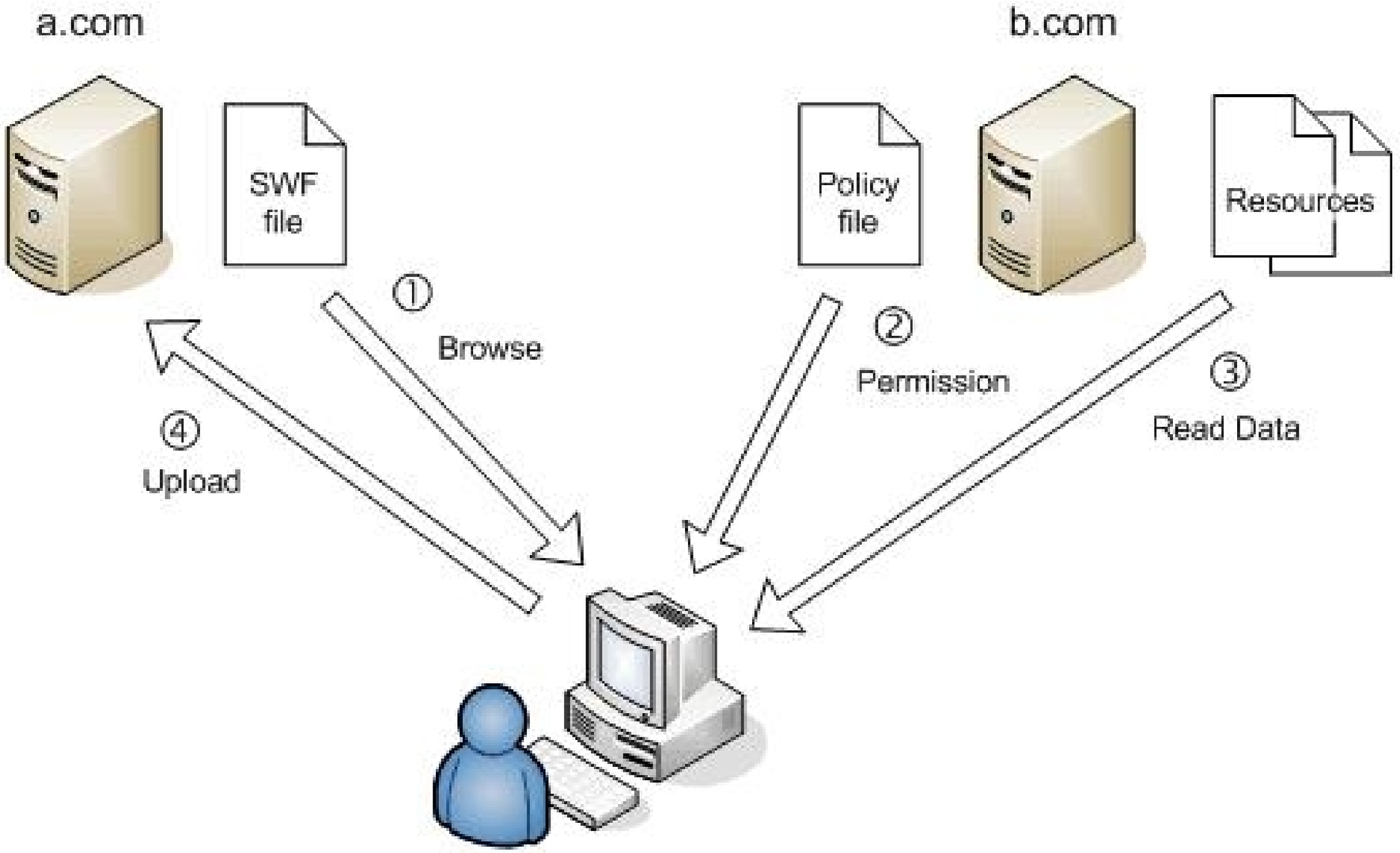
Cross Site Flashing (OWASP-DV-004)

Cross Origin Resource Sharing (CORS) (OWTF-WGP-002)

Some technologies allow settings that relax SOP:

- Adobe Flash (via policy file)
- Microsoft Silverlight (via policy file)
- HTML 5 Cross Origin Resource Sharing (via HTTP headers)

Cheating: Reading the policy file or HTTP headers != attack



http://www.adobe.com/devnet/flashplayer/articles/fplayer9_security.html

Cross Site Flashing (OWASP-DV-004)

Policy file retrieval for analysis

Testing For Cross Site Flashing - ***SEMI PASSIVE***

PLUGIN	START	END
semi_passive/Testing_for_Cross_site_flashing@OWASP-DV-004.py	08/02/2012-13:44	08/02/2012-13:4

NOTES

Edit

HTTP://HACKADEMIC1.TEILAR.GR/CROSSDOMAIN.XML

Not Found

HTTP://HACKADEMIC1.TEILAR.GR/CLIENTACCESSPOLICY.XML

Not Found

HTTP TRANSACTIONS

REQUEST

See Transaction 5 (0s, 245ms) Site F R H B

```
GET /crossdomain.xml HTTP/1.1
Accept-Encoding: identity
Host: hackademic1.teilar.gr
Connection: close
User-Agent: Mozilla/5.0 (X11; Linux i686; rv:6.0) Gecko/20
```

RESPONSE

```
404 Not Found
Date: Wed, 08 Feb 2012 12:45:13 GMT
Server: Apache/2.2.17 (Fedora)
Content-Length: 300
Connection: close
Content-Type: text/html; charset=iso-8859-1


<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
```

Cross Site Flashing (OWASP-DV-004)

CSRF by design → read tokens = attacker WIN

Flash / Silverlight - crossdomain.xml

```
<cross-domain-policy>
<allow-access-from domain="*"/>
</cross-domain-policy>
```

Bad defence example: restrict pushing headers accepted by Flash:
All headers from any domain accepted

```
<allow-http-request-headers-from domain="*" headers="*" />
```

Flash: http://kb2.adobe.com/cps/403/kb403185.html

Cross Site Flashing (OWASP-DV-004)

CSRF by design → read tokens = attacker WIN

Silverlight - clientaccesspolicy.xml

```
<?xml version="1.0" encoding="utf-8"?><access-policy><cross-domain-
access><policy>
    <allow-from http-request-headers="SOAPAction">
        <domain uri="*"/>
    </allow-from>
    <grant-to><resource path="/" include-subpaths="true"/></grant-to>
</policy></cross-domain-access></access-policy>
```

Silverlight: http://msdn.microsoft.com/en-us/library/cc197955%28v=vs.95%29.aspx

Cross Site Flashing (OWASP-DV-004)

Need help?

Testing For Cross Site Flashing - *EXTERNAL*

PLUGIN	START	E
external/Testing_for_Cross_site_flashing@OWASP-DV-004.py	08/02/2012-13:37	C

NOTES

Online Resources: Open All In Tabs

- Krzysztof Kotowicz's CORS proxy browser
- Erlend Oftedal's MalaRIA proxy for crossdomain.xml + clientaccesspolicy.xml
- Julien Couvreur's PoC via URL
- Craft Flash file for Free via Haxe
- Mario Heiderich's sample Haxe file
- Silverlight's clientaccesspolicy.xml info
- crossdomain.xml explained
- fscommand to call JavaScript from Flash

Workshop exercise

1) Install swtftools:

wget http://www.swftools.org/swftools-0.9.2.tar.gz
tar xvfz swftools-0.9.2.tar.gz
cd swftools-0.9.2
sh ./configure
make
make install
whereis swfdump ← Check that we have swfdump installed now
swfdump: /usr/local/bin/swfdump

Workshop exercise (continued)

2) Analyse vulnerable file:

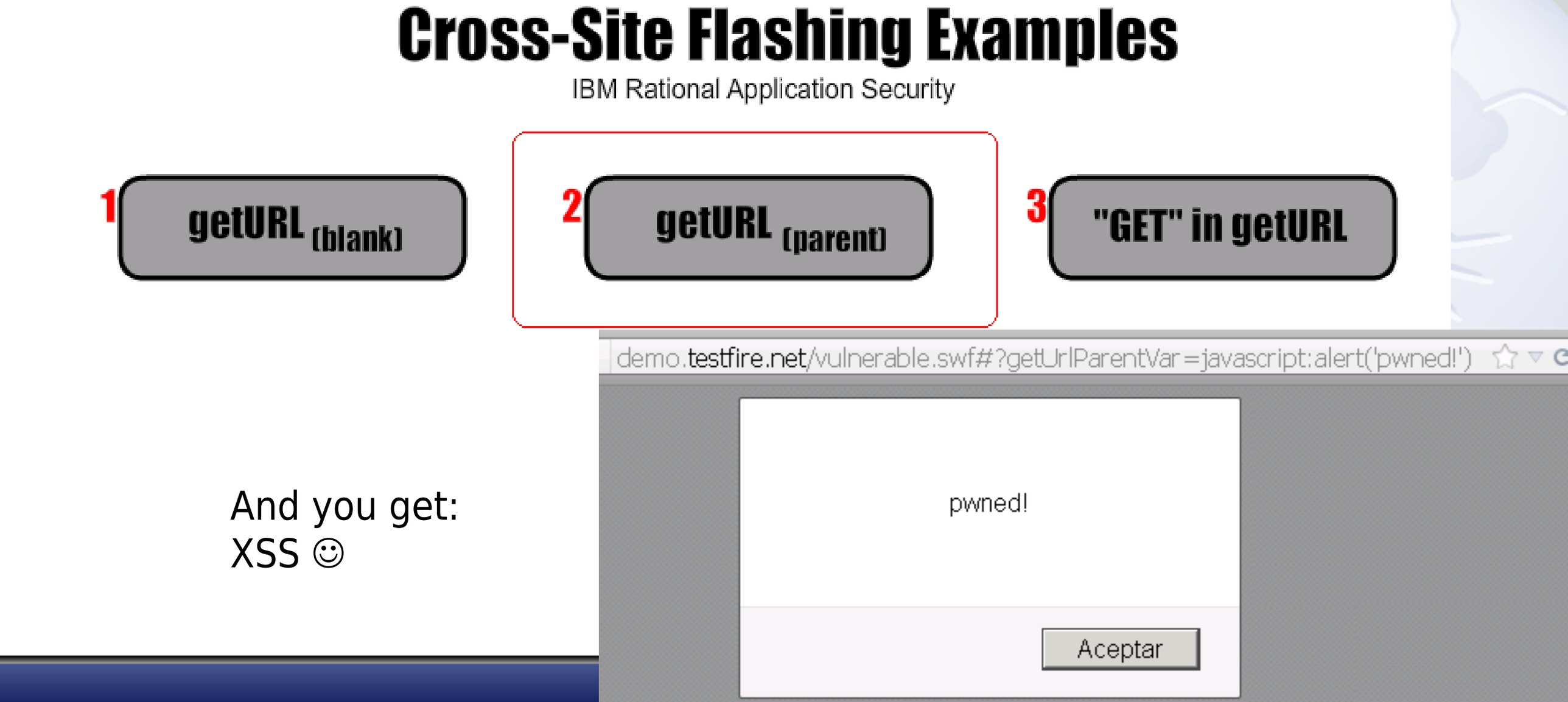
wget http://demo.testfire.net/vulnerable.swf ← Download vulnerable file
swfdump -a vulnerable.swf > vulnerable.txt ← Disassemble flash file
grep -B1 GetVariable vulnerable.txt|tr " " "\n"|grep '("'|sort –u ← Get FlashVars
("empty_mc")
("externalInterfaceVar")
("flash")
("font")
("fontTxtFieldExists")
("fontVar")
("getUrlBlankVar")
("getUrlJSParam")
("**getUrlParentVar**") ← Used in this example
...

Cross Site Flashing (OWASP-DV-004)

Workshop exercise (continued)

3) Verify using the "#" trick (payload not sent to target):

http://demo.testfire.net/vulnerable.swf**#**?
getUrlParentVar=javascript:alert('pwned!')



And you get:
XSS ☺

Cross Origin Resource Sharing (CORS) (OWTF-WGP-002)

Cors - ***GREP***

PLUGIN	START	END	RUNTIME	OUTPUT FILES
grep/CORS@OWTF-WGP-002.py	09/02/2012-08:32	09/02/2012-08:32	0s, 47ms	Browse

NOTES

This plugin looks for HTML 5 Cross Origin Resource Sharing (CORS) headers

Header Analysis Summary

LOG	See log
HTTP TRANSACTION STATS	0 out of 74 (0.0%) matched
ANALYSIS COMMAND	grep -IHiE "(Access-Control-Allow-Origin|Access-Control-Allow-Credentials): " owtf_review/65.61.137.117/80/http__demo.testfire.net/transactions/response_headers/scope_* | sed -e 's|owtf_review/65.61.137.117||g' -e 's|/response_headers/|/|g'

Header Value Analysis

NOTE: Only unique values per header are shown with a link to an example transaction

HEADER	VALUES
Access-Control-Allow-Origin	Not Found
Access-Control-Allow-Credentials	Not Found

ClickJacking (OWTF-WGP-001)

UI Redressing protections:

- **X-Frame-Options** (best)
- **X-Content-Security-Policy** (FF >= 4.0 + Chrome >= 13
- **JavaScript Frame busting** (bypassable sometimes)

Good	Bad
X-Frame-Options: Deny	

Clickjacking - ***GREP***

PLUGIN	START	END	RUNTIME	OUTPUT FILES
grep/Clickjacking@OWTF-WGP-001.py	04/03/2012-07:36	04/03/2012-07:36	0s, 24ms	Browse

NOTES

Edit

This plugin looks for server-side protection headers against Clickjacking (TODO: Add rudimentary searcl for frame busting)

Header Analysis Summary

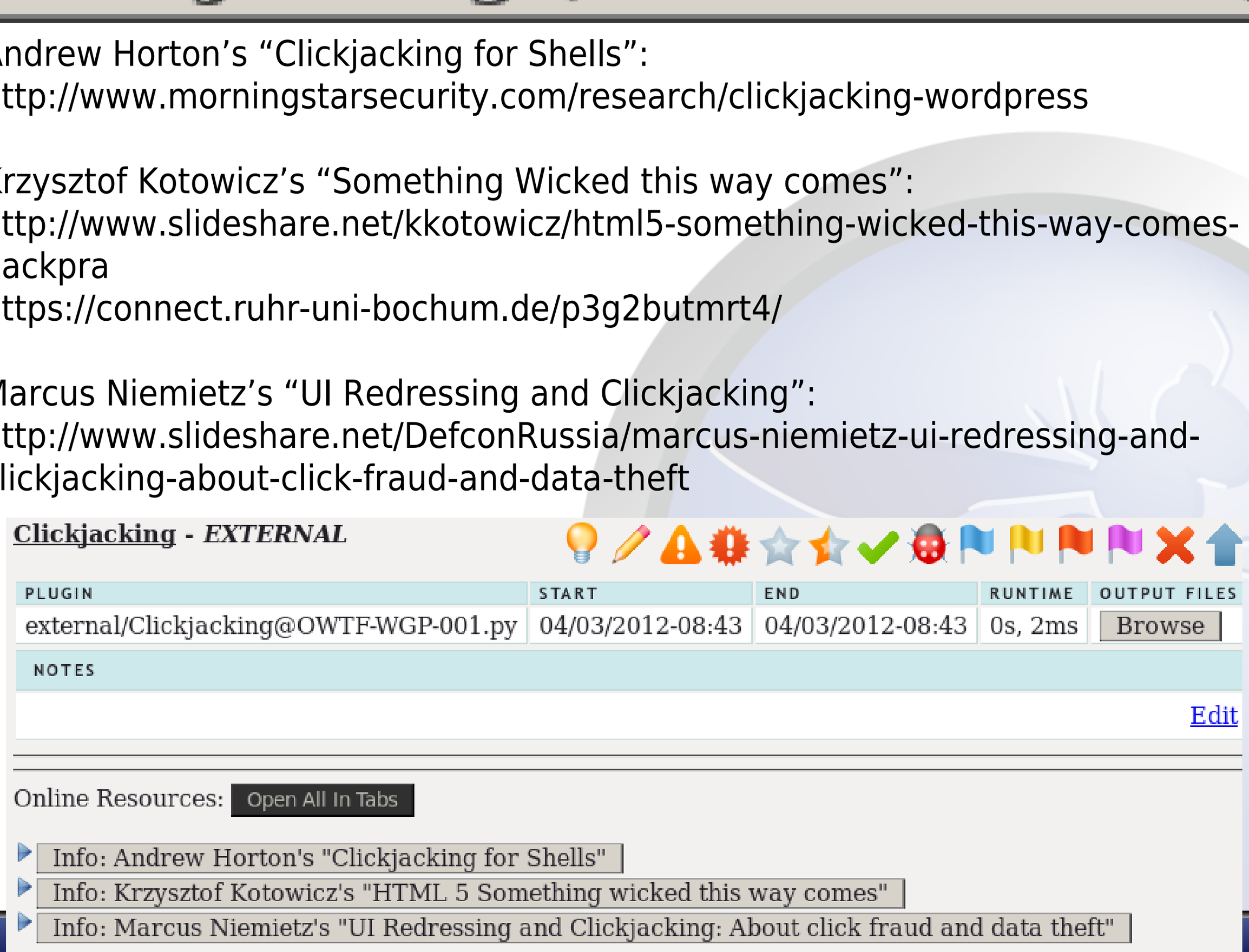
LOG	See log
HTTP TRANSACTION STATS	0 out of 74 (0.0%) matched

ClickJacking (OWTF-WGP-001)

Andrew Horton's "Clickjacking for Shells":
http://www.morningstarsecurity.com/research/clickjacking-wordpress

Krzysztof Kotowicz's "Something Wicked this way comes":
http://www.slideshare.net/kkotowicz/html5-something-wicked-this-way-comes-hackpra
https://connect.ruhr-uni-bochum.de/p3g2butmrt4/

Marcus Niemietz's "UI Redressing and Clickjacking":
http://www.slideshare.net/DefconRussia/marcus-niemietz-ui-redressing-and-clickjacking-about-click-fraud-and-data-theft



Special thanks to

Adi Mutu (@an_animal), Alessandro Fanio González, Anant Shrivastava, Andrés Morales, Andrés Riancho (@w3af), Ankush Jindal, Assem Chelli, Azeddine Islam Mennouchi, Bharadwaj Machiraju, Chris John Riley, Gareth Heyes (@garethheyes), Hani Benhabiles, Javier Marcos de Prado, Johanna Curiel, Krzysztof Kotowicz (@kkotowicz), Marc Wickenden (@marcwickenden), Marcus Niemietz (@mniemietz), Mario Heiderich (@0x6D6172696F), Martin Johns, Michael Kohl (@citizen428), Nicolas Grégoire (@Agarri_FR), Sandro Gauci (@sandrogauci), OWASP Testing Guide contributors

All those OWTF students that **tried** to participate in the GSoC even if they couldn't make it this time ☹

Finux Tech Weekly – Episode 17 – mins 31-49
http://www.finux.co.uk/episodes/mp3/FTW-EP17.mp3
Finux Tech Weekly – Episode 12 – mins 33-38
http://www.finux.co.uk/episodes/mp3/FTW-EP12.mp3
http://www.finux.co.uk/episodes/ogg/FTW-EP12.ogg
Exotic Liability – Episode 83 – mins 49-53
http://exoticliability.libsyn.com/exotic-liability-83-oh-yeah

Q & A

Contact/Links:

http://owtf.org

@7a_ @owtfp

abraham.aranguren@owasp.org