

Week 4

Penetration Testing Report

Introduction

This report document hereby describes the proceedings and results of a Black Box security assessment conducted against the **Week 4 Labs**. The report hereby lists the findings and corresponding best practice mitigation actions and recommendations.

1. Objective

The objective of the assessment was to uncover vulnerabilities in the **Week 4 Labs** and provide a final security assessment report comprising vulnerabilities, remediation strategy and recommendation guidelines to help mitigate the identified vulnerabilities and risks during the activity.

2. Scope

This section defines the scope and boundaries of the project.

Application Name	{Open Redirect Labs}, {Exchangeable Image File Format Labs}
------------------	---

3. Summary

Outlined is a Black Box Application Security assessment for the **Week 4 Labs**.

Total number of Sub-labs: 9 Sub-labs

High	Medium	Low
3	2	4

High - Number of Sub-labs with hard difficulty level

Medium - Number of Sub-labs with Medium difficulty level

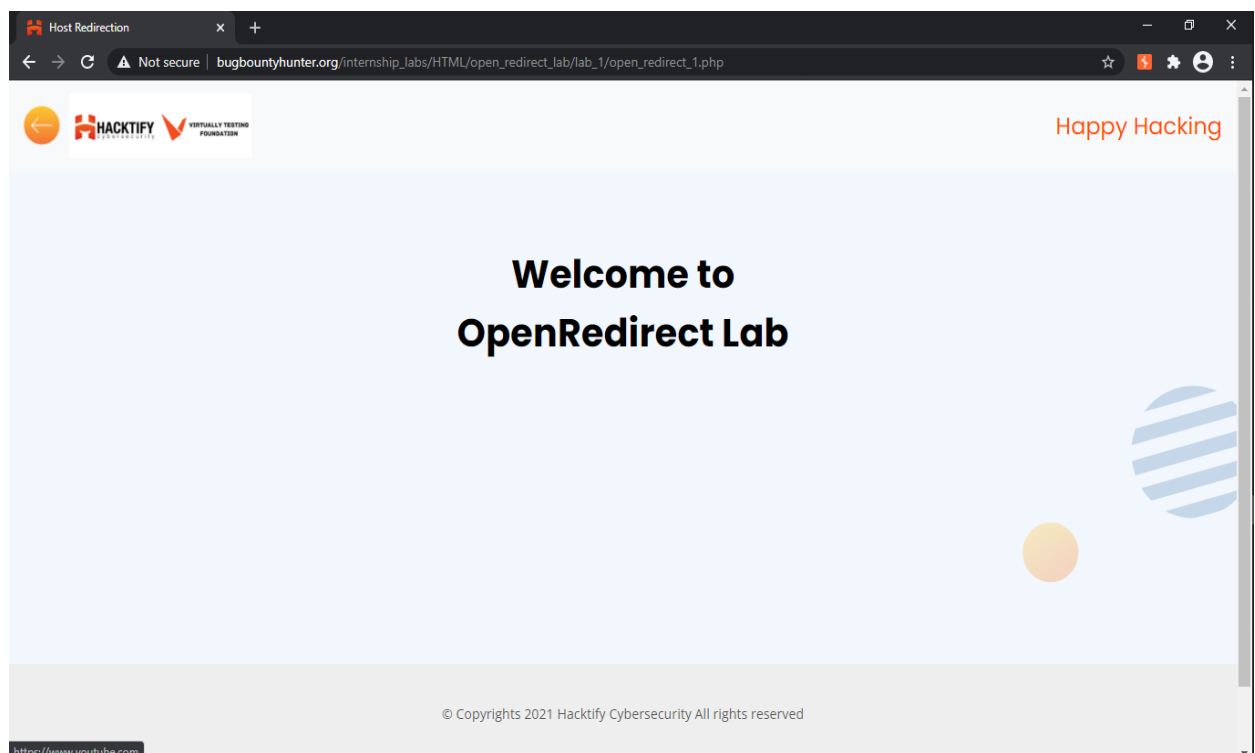
Low - Number of Sub-labs with Easy difficulty level

1. {Open Redirect Labs}

1.1. {A Simple Host!}

Reference	Risk Rating
A Simple Host!	Low
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability using the burp suite tool by intercepting it.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_1/open_redirect_1.php	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
Maintain a server-side list of all URLs that are permitted for redirection. Instead of passing the target URL as a parameter to a redirector, pass an index into the list.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

Proof of Concept



1.2. { Story of a beautiful header!}

Reference	Risk Rating
Story of a beautiful header	Low
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a redirect request into burp suite and inserting payload into it and then by forwarding through repeater by verifying the location of the payload result.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_2/open_redirect_2.php	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
Remove the redirection function from the application, and replace links to it with direct links to the relevant target URLs.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

Proof of Concept

The screenshot displays the Burp Suite Professional interface, specifically the Repeater tab. The target URL is <https://www.bugbountyhunter.org>. The request is a GET to `https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_2/open_redirect_2.php?username=hacktify&password=hacktify&url=https://www.youtube.com&login=Login HTTP/1.1`. The response is a 302 Found status with a Location header pointing to `https://www.youtube.com/channel/UCv37sLc4p2yXAm0...`. The response body shows the HTML structure of the YouTube channel page.

```
1 GET /internship_labs/HTML/open_redirect_lab/lab_2/open_redirect_2.php?username=hacktify&password=hacktify&url=https://www.youtube.com&login=Login HTTP/1.1
2 Host: www.bugbountyhunter.org
3 Connection: close
4 Cache-Control: max-age=0
5 sec-ch-ua: "Not A Brand";v="55", "Chromium";v="88"
6 sec-ch-ua-mobile: ?0
7 Upgrade-Insecure-Requests: 1
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
10 Sec-Fetch-Site: none
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-User: ?1
13 Sec-Fetch-Dest: document
14 Accept-Encoding: gzip, deflate
15 Accept-Language: en-US,en;q=0.9,en;q=0.8
16 Cookie: PHPSESSID=opin7h0btvniilms572lughsa
17
18
```

```
1 HTTP/1.1 302 Found
2 Date: Tue, 02 Nov 2021 11:10:09 GMT
3 Content-Type: text/html; charset=UTF-8
4 Connection: close
5 expires: Thu, 15 Nov 1981 08:52:00 GMT
6 cache-control: no-store, no-cache, must-revalidate
7 pragma: no-cache
8 set-cookie: PHPSESSID=usiskln0ba15su29sg6dnn5dgm; path=/
9 location: youtube.com
10 CP-Cache-Status: DYNAMIC
11 Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/";
12 Report-To: {"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v3?s=Lc4p2yXAm0..."}],"success_fraction":0,"report_to":"cf-nel","max_age":604800)}
13 NEL: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
14 Server: cloudflare
15 CF-RAY: 6a7ceacbbcf3187-BOM
16 alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443
17 Content-Length: 3682
18
19
20 <html>
21 <head>
22 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
23 <meta name="viewport" content="width=device-width, initial-scale=1.0" />
24 <meta name="keywords" content="" />
25 <link rel="icon" href="https://www.youtube.com/assets/img/favicon.png" />
26 <link rel="stylesheet" type="text/css" href="https://www.youtube.com/assets/css/animate.css" />
27 <link rel="stylesheet" type="text/css" href="https://www.youtube.com/assets/css/bootstrap.min.css" />
28 <link rel="stylesheet" type="text/css" href="https://www.youtube.com/assets/css/font-awesome.min.css" />
29 <link rel="stylesheet" type="text/css" href="https://www.youtube.com/assets/css/main.css" />
30 <link rel="stylesheet" type="text/css" href="https://www.youtube.com/assets/css/responsive.css" />
31 <title>
32 Login
33 </title>
34 <style>
35 .containers{
36 margin:0;
37
```

1.3. {Sanitize Params!!}

Reference	Risk Rating
Sanitize params!!	Medium
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a redirect request into burp suite and inserting payload into it and then by forwarding through repeater by verifying the location of the payload result.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_3/open_redirect_3_dashboard.php	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
The application should use relative URLs in all of its redirects, and the redirection function should strictly validate that the URL received is a valid URL.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

Proof of Concept

Burp Suite Professional v2021.2.1 - Temporary Project - licensed to Uncia

Burp Project Intruder Repeater Window Help

Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options

8 x ...

Send Cancel < > Follow redirection

Target: https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_3/open_redirect_3_dashboard.php

Request

Pretty Raw \n Actions

```
1 GET /internship_labs/HTML/open_redirect_lab/lab_3/open_redirect_3.php?username=hacktify&password=hacktify&url=youtube.com&login=login HTTP/1.1
2 Host: www.bugbountyhunter.org
3 Connection: close
4 Cache-Control: max-age=0
5 sec-ch-ua: ",Not A Brand",v="99", "Chromium",v="88"
6 sec-ch-ua-mobile: ?0
7 Upgrade-Insecure-Requests: 1
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36
9 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.9,application/signed-exchange;v=b3;q=0.9
10 Sec-Fetch-Site: none
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-User: ?1
13 Sec-Fetch-Dest: document
14 Accept-Encoding: gzip, deflate
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Cookie: PHPSESSID=opin7k0btvnlk1ms72lugksa
17
18
```

Response

Pretty Raw Render \n Actions

```
1 HTTP/1.1 302 Found
2 Date: Tue, 02 Nov 2021 11:10:09 GMT
3 Content-Type: text/html; charset=UTF-8
4 Connection: close
5 expires: Thu, 19 Nov 1981 08:52:00 GMT
6 cache-control: no-store, no-cache, must-revalidate
7 pragma: no-cache
8 set-cookie: PHPSESSID=usiskln0bal5su29sg8dm5dgm; path=/
9 location: youtube.com
10 CF-Cache-Status: DYNAMIC
11 Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/rep
12 Report-To: {\"endpoints\": [{\"url\": \"https://a.nel.cloudflare.com/rep
13 NEL: {\"success_fraction\": 0, \"report_to\": \"cf-nel\", \"max_age\": 604800}
14 Server: cloudflare
15 CF-RAY: 6a7cea2bcb2f3187-BOM
16 alt-svc: h3=\":443\"; ma=86400, h3-29=\":443\"; ma=86400, h3-28=\":443\"; m
17 Content-Length: 3682
18
19
20 <html>
21 <head>
22 <meta http-equiv=\"Content-Type\" content=\"text/html; charset=UTF-8
23 <meta name=\"viewport\" content=\"width=device-width, initial-scale=
24 <meta name=\"keywords\" content=\"\" />
25 <link rel=\"icon\" href=\"../assets/img/favicon.png\" />
26 <link rel=\"stylesheet\" type=\"text/css\" href=\"../assets/css/ani
27 <link rel=\"stylesheet\" type=\"text/css\" href=\"../assets/css/boo
28 <link rel=\"stylesheet\" type=\"text/css\" href=\"../assets/css/fon
29 <link rel=\"stylesheet\" type=\"text/css\" href=\"../assets/css/mai
30 <link rel=\"stylesheet\" type=\"text/css\" href=\"../assets/css/res
31 <title>
32 Login
33 </title>
34 <style>
35 .containers{
```

1.4. {Patterns are important!!}

Reference	Risk Rating
Patterns are important!!	Medium
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability by inserting payload at the end of link by adding "?" and then followed by a payload.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_4/open_redirect_4.php///youtube.com	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
Maintain a server-side list of all URLs that are permitted for redirection. Instead of passing the target URL as a parameter to a redirector, pass an index into the list.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

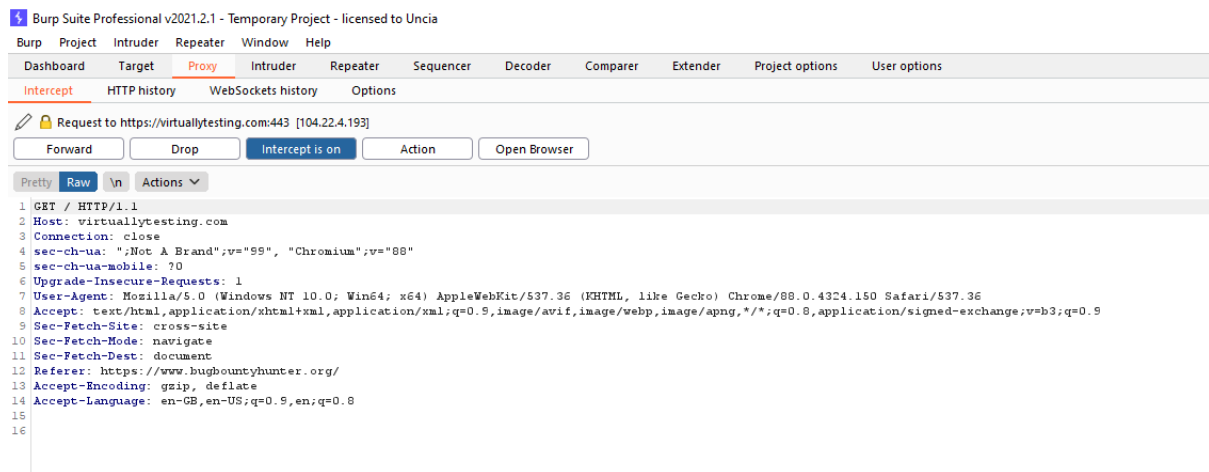
Proof of Concept



1.5. {File Upload? Redirect IT!}

Reference	Risk Rating
File upload? Redirect IT!	Low
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a file upload request into burp suite and changing file name to make it vulnerable.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://virtuallytesting.com/	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
Maintain a server-side list of all URLs that are permitted for redirection. Instead of passing the target URL as a parameter to a redirector, pass an index into the list.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

Proof of Concept



1.6. {Same Param Twice!}

Reference	Risk Rating
Same Param Twice!	High
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a redirect request into burp suite and inserting payload into it and then by forwarding through repeater by verifying the location of the payload result.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_6/open_redirect_6_dashboard.php	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
Remove the redirection function from the application, and replace links to it with direct links to the relevant target URLs.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

Proof of Concept

The screenshot displays the Burp Suite Professional interface. The 'Request' tab is active, showing a GET request to `https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_6/open_redirect_6_dashboard.php` with a payload `?username=hacktify&password=hacktify&url=open_redirect_6_dashboard.php&youtube.com&login=Login HTTP/1.1`. The 'Response' tab shows the server's reply, which is an HTML document with a 302 status code and a 'Location' header pointing to `https://youtube.com`. The response body contains HTML markup for a login page.

Request:

```
1 GET /internship_labs/HTML/open_redirect_lab/lab_6/open_redirect_6.php?username=hacktify&password=hacktify&url=open_redirect_6_dashboard.php&youtube.com&login=Login HTTP/1.1
2 Host: www.bugbountyhunter.org
3 Connection: close
4 sec-ch-ua: "Not A Brand";v="99", "Chromium";v="88"
5 sec-ch-ua-mobile: ?0
6 Upgrade-Insecure-Requests: 1
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
9 Sec-Fetch-Site: same-origin
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Referer: https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_6/open_redirect_6.php
14 Accept-Encoding: gzip, deflate
15 Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
16 Cookie: PHPSESSID=eivklnk3jdvrsjh04ufokh4pn
17
18
```

Response:

```
1 HTTP/1.1 302 Found
2 Date: Tue, 02 Nov 2021 11:31:55 GMT
3 Content-Type: text/html; charset=UTF-8
4 Connection: close
5 expires: Thu, 19 Nov 1981 08:52:00 GMT
6 cache-control: no-store, no-cache, must-revalidate
7 pragma: no-cache
8 set-cookie: PHPSESSID=315t0grjql3svlv3b8hohskA0; path=/
9 Location: https://youtube.com
10 CF-Cache-Status: DYNAMIC
11 Expect-CT: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct";
12 Report-To: {"endpoints":[{"url":"https://a.nel.cloudflare.com/report/v3?s=rak0urU7bQm"}]}
13 NEL: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
14 Server: cloudflare
15 CF-RAY: 6a7d0A0cdbe60f6c-BOM
16 alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"
17 Content-Length: 3692
18
19
20 <html>
21 <head>
22 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
23 <meta name="viewport" content="width=device-width, initial-scale=1.0" />
24 <meta name="keywords" content="" />
25 <link rel="icon" href="../../assets/img/favicon.png" />
26 <link rel="stylesheet" type="text/css" href="../../assets/css/animate.css" />
27 <link rel="stylesheet" type="text/css" href="../../assets/css/bootstrap.min.css" />
28 <link rel="stylesheet" type="text/css" href="../../assets/css/font-awesome.min.css" />
29 <link rel="stylesheet" type="text/css" href="../../assets/css/main.css" />
30 <link rel="stylesheet" type="text/css" href="../../assets/css/responsive.css" />
31 <title>
32 Login
33 </title>
34 <style>
35 .containers{
36 margin:0;
37
```

1.7. {Domains? Not Always!}

Reference	Risk Rating
Domains? Not Always!	High
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a redirect request into burp suite and inserting payload into it and then by forwarding through repeater by verifying the location of the payload result.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_7/open_redirect_7_dashboard.php?url=google.co,m&login=Login	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
Maintain a server-side list of all URLs that are permitted for redirection. Instead of passing the target URL as a parameter to a redirector, pass an index into the list.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

Proof of Concept

The screenshot displays the Burp Suite Professional interface. The top menu bar includes options like Dashboard, Project, Intruder, Repeater, Window, and Help. Below the menu is a toolbar with buttons for Send, Cancel, and navigation arrows. The main workspace is divided into two panels: Request and Response.

Request Panel: Shows a raw HTTP request. The first line is a GET request to `/internship_labs/HTML/open_redirect_lab/lab_7/open_redirect_7.php?username=hacktify&password=hacktify&url=open_redirect_7_dashboard.php?url=www.facebook.com&login=Login`. The request includes standard headers like Host, Connection, User-Agent, and Referer.

Response Panel: Shows the raw HTTP response. The first line is a 200 OK status. The response includes headers like Date, Content-Type, Expires, Cache-Control, Pragma, Set-Cookie, Location, Expect-CT, Report-To, Vary, Server, CF-RAY, and Content-Length. The body of the response is HTML, starting with `<html>` and `<head>` tags.

The bottom status bar indicates "0 matches" for the search and "4,651 bytes | 292 millis" for the response.

1.8. {Style digit symbols < 3}

Reference	Risk Rating
Style digit symbols < 3	High
Tools Used	
Google Chrome and Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a redirect request into burp suite and inserting IP Address as payload into it and then by forwarding through repeater by verifying the location of the payload result.	
How It Was Discovered	
Manual Analysis and Automated Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_8/open_redirect_8.php https://www.google.com/?gws_rd=ssl	
Consequences of not Fixing the Issue	
An attacker can use this vulnerability to redirect users to other malicious websites, which can lead to phishing or other similar attacks.	
Suggested Countermeasures	
Remove the redirection function from the application, and replace links to it with direct links to the relevant target URLs.	
References	
https://cheatsheetseries.owasp.org/cheatsheets/Unvalidated_Redirects_and_Forwards_Cheat_Sheet.html	

Proof of Concept

The screenshot displays the Burp Suite Professional interface. The top toolbar includes buttons for Send, Cancel, and Follow redirection. The main window is divided into two panes: Request and Response.

Request Pane: Shows a raw HTTP request to `https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_8/open_redirect_8.php?username=hacktify&password=hacktify&url=142.250.188.142&login=Login`. The request includes headers like `Host: www.bugbountyhunter.org`, `Connection: close`, `sec-ch-ua: "Not A Brand",v="59", "Chromium",v="88"`, `sec-ch-ua-mobile: ?0`, `Upgrade-Insecure-Requests: 1`, `User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36`, `Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9`, `Sec-Fetch-Site: same-origin`, `Sec-Fetch-Mode: navigate`, `Sec-Fetch-User: ?1`, `Sec-Fetch-Dest: document`, and `Referer: https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_8/open_redirect_8.php`. The body is empty.

Response Pane: Shows a raw HTTP response from `https://www.bugbountyhunter.org/internship_labs/HTML/open_redirect_lab/lab_8/open_redirect_8.php`. The response status is `200 OK`. The headers include `Date: Tue, 02 Nov 2021 12:06:27 GMT`, `Content-Type: text/html; charset=UTF-8`, `Connection: close`, `expires: Thu, 19 Nov 1981 08:52:00 GMT`, `cache-control: no-store, no-cache, must-revalidate`, `pragma: no-cache`, `set-cookie: PHPSESSID=qdnohiv73a9nj2a3kh01262pb; path=/`, and `location: http://142.250.188.142`. The body contains HTML code for a login page, including `<html>`, `<head>`, `<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">`, `<meta name="viewport" content="width=device-width, initial-scale=1.0" />`, `<meta name="keywords" content="" />`, `<link rel="icon" href=".../assets/img/favicon.png" />`, `<link rel="stylesheet" type="text/css" href=".../assets/css/animate.css" />`, `<link rel="stylesheet" type="text/css" href=".../assets/css/bootstrap.min.css" />`, `<link rel="stylesheet" type="text/css" href=".../assets/css/font-awesome.min.css" />`, `<link rel="stylesheet" type="text/css" href=".../assets/css/main.css" />`, `<link rel="stylesheet" type="text/css" href=".../assets/css/responsive.css" />`, `<title>`, `Login`, `</title>`, `<style>`, `.container{`, `margin:0;`, and `</style>`.

2. {Exchangeable Image File Format Labs}

2.1. Lets PII!

Reference	Risk Rating
Lets PII!	Low
Tools Used	
Google Chrome	
Vulnerability Description	
I found this vulnerability by uploading an image into the profile photo section. I found the entry point for EXIF in the profile photo upload section. This allowed me to upload images containing sensitive data.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/exif_lab/lab_1/exif.php	
Consequences of not Fixing the Issue	
This vulnerability violates the privacy of a user and shares sensitive information of the user who uploaded an image on the vulnerable websites.	
Suggested Countermeasures	
Disable geotagging on the digital device you use to take photographs.	
References	
https://photographylife.com/what-is-exif-data	

Proof of Concept

