Week 5 Penetration Testing Report

Introduction

This report document hereby describes the proceedings and results of a Black Box security assessment conducted against the **Week 5 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 5 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	**{Cross-Origin Resource Sharing Labs}**

3. Summary

Outlined is a Black Box Application Security assessment for the **Week 5 Labs**.

Total number of Sub-labs: 7 Sub-labs

High	Medium	Low
3	2	2

High - **Number of Sub-labs with hard difficulty level**

Medium - **Number of Sub-labs with Medium difficulty level**

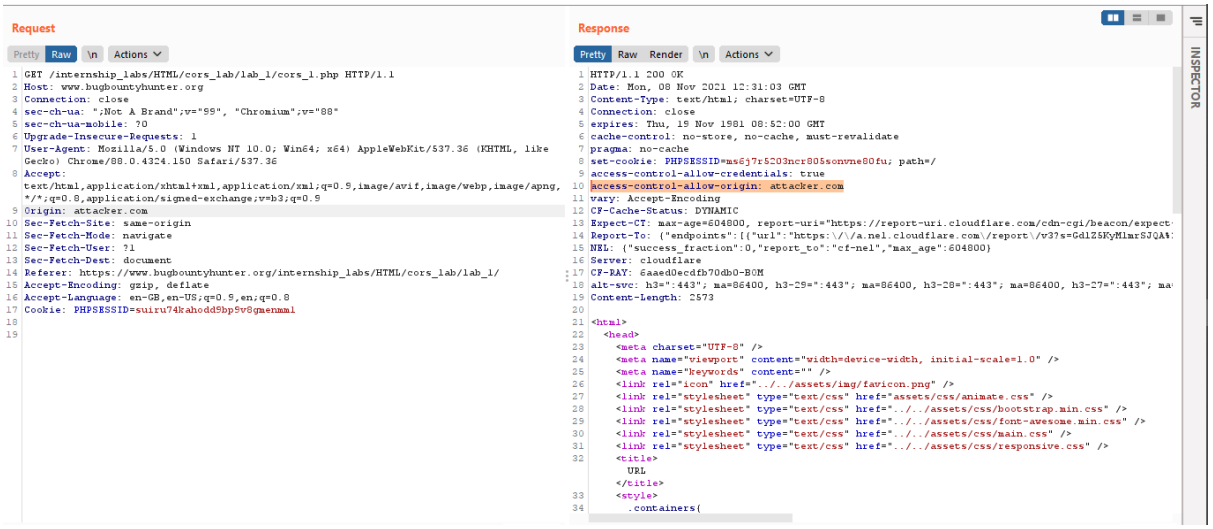
Low - **Number of Sub-labs with Easy difficulty level**

1. {Cross-Origin Resource Sharing Labs}

1.1. {CORS with Arbitrary Origin}

Reference	Risk Rating
CORS with Arbitrary Origin	**Low**
Tools Used	
Google Chrome Browser, CORS tool, Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and adding origin header to the interception and forwarding it and successfully getting expected output.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_1/login.php	
Consequences of not Fixing the Issue	
Attackers would treat many victims to visit the attacker's website, if the victim is logged in, then his personal information is recorded in the attacker's server. Attackers can perform any action in the user's account, by bypassing CSRF tokens.	
Suggested Countermeasures	
Only Allow trusted sites.	
References	
https://portswigger.net/web-security/cors	

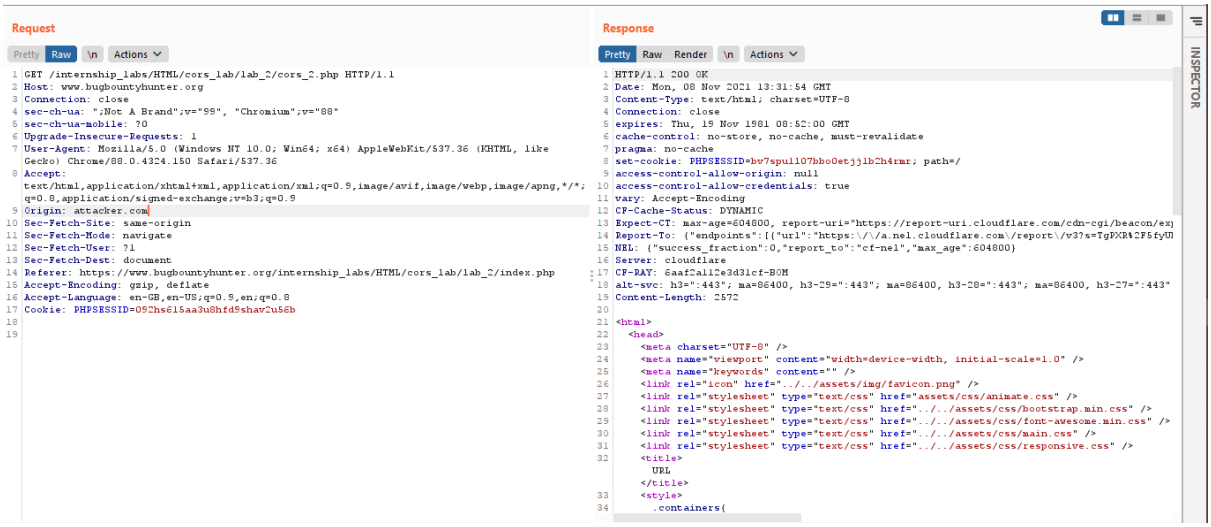
Proof of Concept



1.2. {CORS with null origin}

Reference	Risk Rating
CORS with null origin	**Low**
Tools Used	
Google Chrome Browser, CORS tool, Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and adding origin header to the interception and forwarding it and successfully getting expected output.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_2/login.php	
Consequences of not Fixing the Issue	
Attackers would treat many victims to visit the attacker's website, if the victim is logged in, then his personal information is recorded in the attacker's server. Attackers can perform any action in the user's account, by bypassing CSRF tokens.	
Suggested Countermeasures	
Avoid whitelisting null.	
References	
https://owasp.org/www-community/attacks/CORS_OriginHeaderScrutiny	

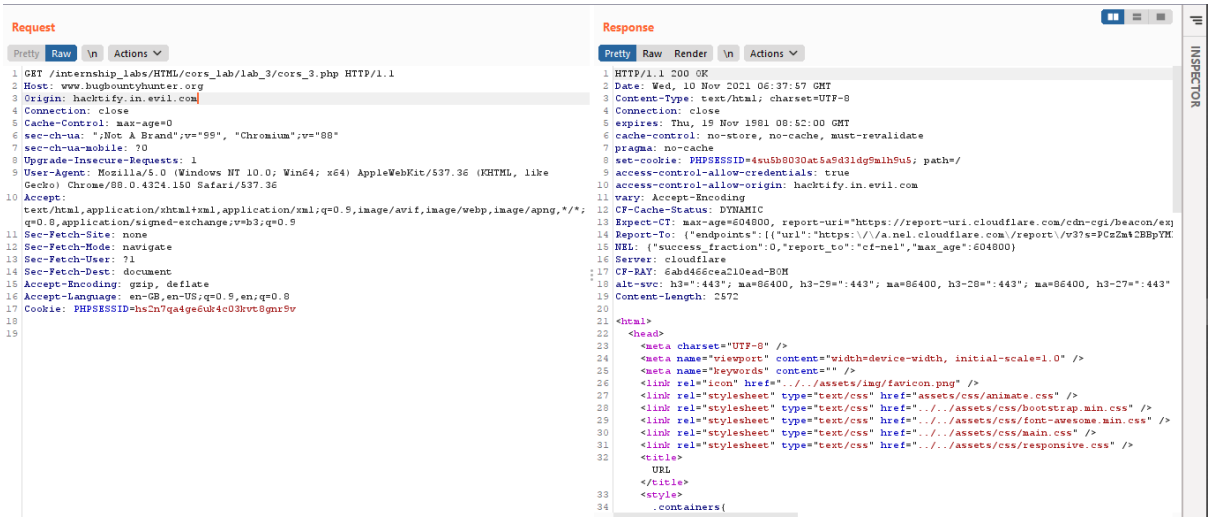
Proof of Concept



1.3. {CORS with Prefix Match}

Reference	Risk Rating
CORS with Prefix Match	**Medium**
Tools Used	
Google Chrome Browser, CORS tool, Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and adding origin header to the interception and forwarding it and successfully getting expected output.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_3/cors_3.php	
Consequences of not Fixing the Issue	
Attackers would treat many victims to visit the attacker's website, if the victim is logged in, then his personal information is recorded in the attacker's server. Attackers can perform any action in the user's account, by bypassing CSRF tokens.	
Suggested Countermeasures	
Proper configuration of cross-domains requests.	
References	
https://portswigger.net/web-security/cors	

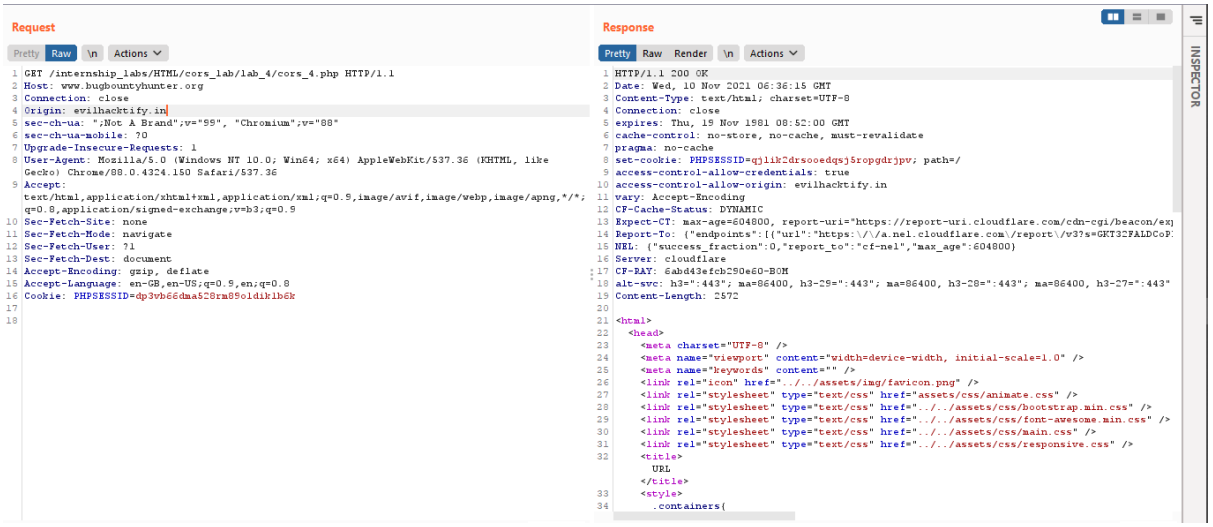
Proof of Concept



1.4. {CORS with Suffix Match}

Reference	Risk Rating
CORS with Suffix Match	**Medium**
Tools Used	
Google Chrome Browser, CORS tool, Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and adding origin header to the interception and forwarding it and successfully getting expected output.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_4/cors_4.php	
Consequences of not Fixing the Issue	
Attackers would treat many victims to visit the attacker's website, if the victim is logged in, then his personal information is recorded in the attacker's server. Attackers can perform any action in the user's account, by bypassing CSRF tokens.	
Suggested Countermeasures	
Only Allow trusted sites.	
References	
https://owasp.org/www-community/attacks/CORS_OriginHeaderScrutiny	

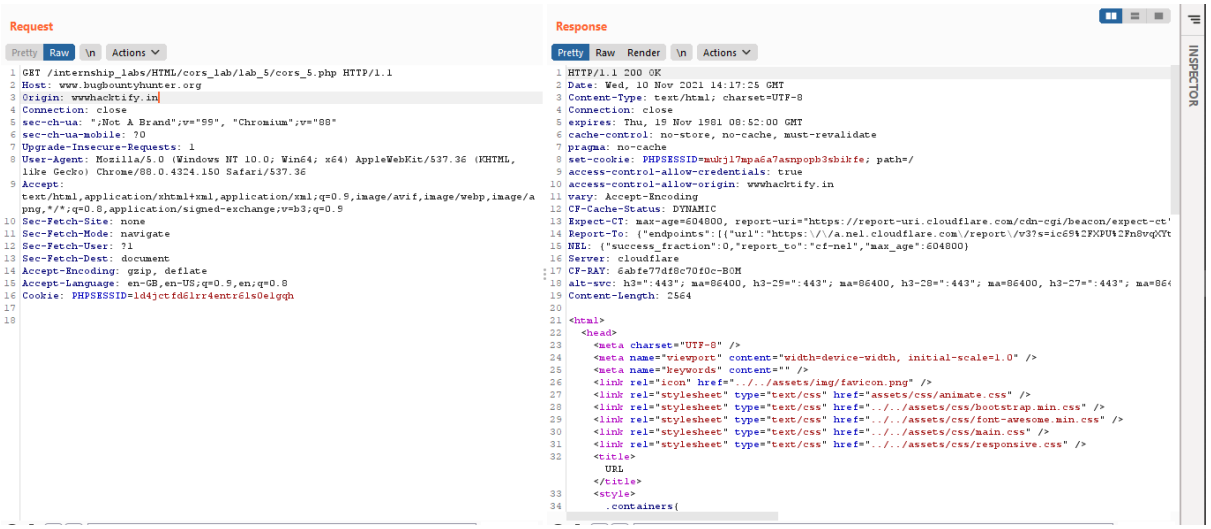
Proof of Concept



1.5. {CORS with Escape Dot}

Reference	Risk Rating
CORS with Escape Dot	**Hard**
Tools Used	
Google Chrome Browser, CORS tool, Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and adding origin header to the interception and forwarding it and successfully getting expected output.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_5/cors_5.php	
Consequences of not Fixing the Issue	
Attackers would treat many victims to visit the attacker's website, if the victim is logged in, then his personal information is recorded in the attacker's server. Attackers can perform any action in the user's account, by bypassing CSRF tokens.	
Suggested Countermeasures	
Avoid wildcards in internal networks.	
References	
https://portswigger.net/web-security/cors	

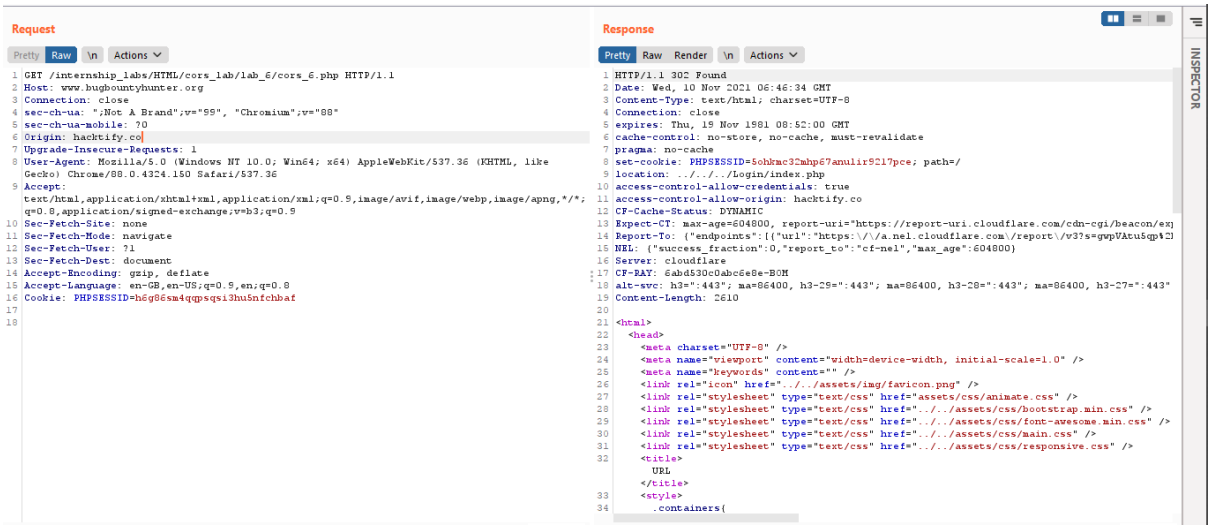
Proof of Concept



1.6. {CORS with Substring Match}

Reference	Risk Rating
CORS with Substring Match	**Hard**
Tools Used	
Google Chrome Browser, CORS tool, Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and adding origin header to the interception and forwarding it and successfully getting expected output.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_6/cors_6.php	
Consequences of not Fixing the Issue	
Attackers would treat many victims to visit the attacker's website, if the victim is logged in, then his personal information is recorded in the attacker's server. Attackers can perform any action in the user's account, by bypassing CSRF tokens.	
Suggested Countermeasures	
Proper configuration of cross-domains requests.	
References	
https://owasp.org/www-community/attacks/CORS_OriginHeaderScrutiny	

Proof of Concept



1.7. {CORS with Arbitrary Subdomain}

Reference	Risk Rating
CORS with Arbitrary Subdomain	**Hard**
Tools Used	
Google Chrome Browser, CORS tool, Burp Suite	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and adding origin header to the interception and forwarding it and successfully getting expected output.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_7/login.php	
Consequences of not Fixing the Issue	
Attackers would treat many victims to visit the attacker's website, if the victim is logged in, then his personal information is recorded in the attacker's server. Attackers can perform any action in the user's account, by bypassing CSRF tokens.	
Suggested Countermeasures	
Avoid wildcards in internal networks.	
References	
https://developer.mozilla.org/en-US/docs/Web/HTTP/CORS	

Proof of Concept