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**

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**

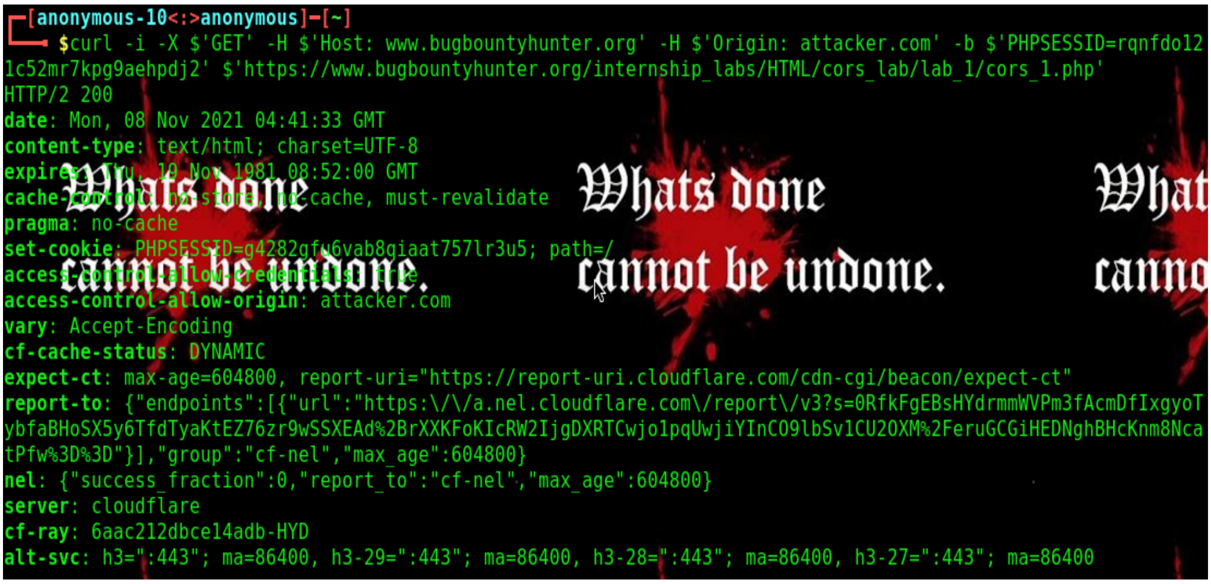
1. Cross-Origin Resource Sharing

1.1. CORS With Arbitrary Origin

Reference	**Risk Rating**
CORS With Arbitrary Origin	**Low**
Tools Used	
Browser, CURL	
Vulnerability Description	
The vulnerability is CORS that makes an attacker to access domain resource using different domain.	
How It Was Discovered	
Manual Analysis - Use Origin header with the attacker domain and pass it using curl.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_1/cors_1.php	
Consequences of not Fixing the Issue	
Account takeover and stealing credentials.	
Suggested Countermeasures	
Allow only trusted sites	
References	
https://portswigger.net/web-security/cors	

Proof of Concept

The proof of the above vulnerability.



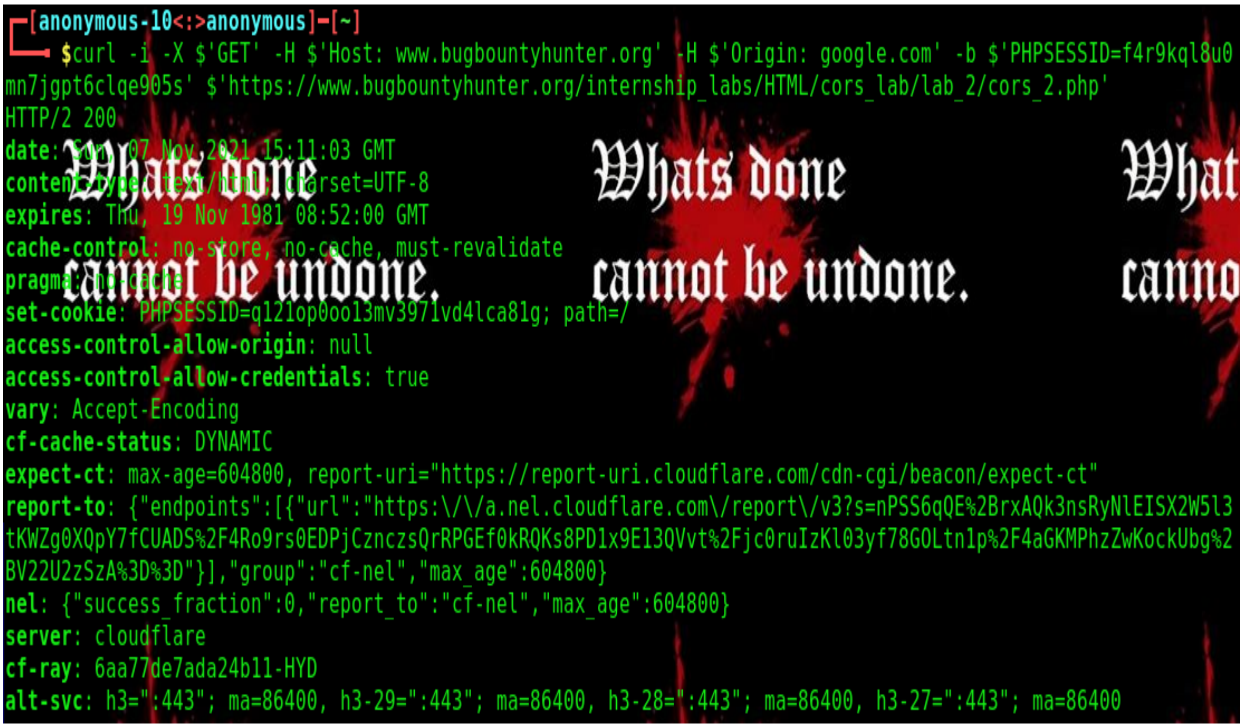
```
┌─[anonymous-10<:>anonymous]─[~]
└──╼ $curl -i -X $'GET' -H $'Host: www.bugbountyhunter.org' -H $'Origin: attacker.com' -b $'PHPSESSID=rqnfdo12
1c52mr7kpg9aehpdj2' $'https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_1/cors_1.php'
HTTP/2 200
date: Mon, 08 Nov 2021 04:41:33 GMT
content-type: text/html; charset=UTF-8
expires: Thu, 19 Nov 1981 08:52:00 GMT
cache-control: no-store, no-cache, must-revalidate
pragma: no-cache
set-cookie: PHPSESSID=g4282gfu6vab8giaat757lr3u5; path=/
access-control-allow-credentials: true
access-control-allow-origin: attacker.com
vary: Accept-Encoding
cf-cache-status: DYNAMIC
expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
report-to: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=0RfkFgEBsHYdrmmWVPm3fAcmDfIxgyoT
ybfaBHoSX5y6TfdTyaKtEZ76zr9wSSXEAd%2BrXXKFoKIcRW2IjgDXRTCwjo1pqUwjiYInC09lbSv1CU20XM%2FeruGCGiHEDNghBHcKnm8Nca
tPfw%3D%3D"}],"group":"cf-nel","max_age":604800}
nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
server: cloudflare
cf-ray: 6aac212dbce14adb-HYD
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400
```

1.2. CORS With Null Origin

Reference	Risk Rating
CORS With Null Origin	**Low**
Tools Used	
Browser, CURL	
Vulnerability Description	
The vulnerability is CORS that makes an attacker to access domain resource using different domain.	
How It Was Discovered	
Manual Analysis - Use Origin header with the attacker domain or null and pass it using curl.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_2/cors_2.php	
Consequences of not Fixing the Issue	
Account takeover and stealing credentials.	
Suggested Countermeasures	
Allow only trusted sites	
References	
https://portswigger.net/web-security/cors	

Proof of Concept

The proof of the above vulnerability.



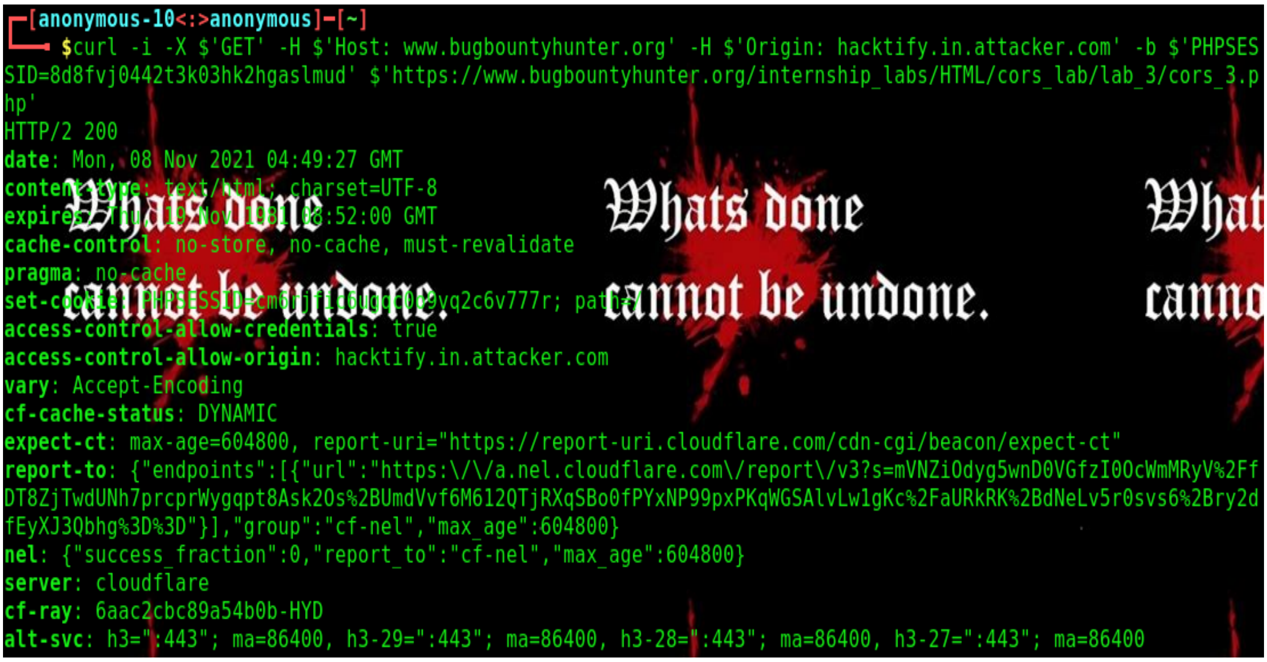
```
┌─[anonymous-10<:>anonymous]─[~]
└──╼ $curl -i -X $'GET' -H $'Host: www.bugbountyhunter.org' -H $'Origin: google.com' -b $'PHPSESSID=f4r9kql8u0
mn7jgpt6clqe905s' $'https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_2/cors_2.php'
HTTP/2 200
date: Sun, 07 Nov 2021 15:11:03 GMT
content-type: text/html; charset=UTF-8
expires: Thu, 19 Nov 1981 08:52:00 GMT
cache-control: no-store, no-cache, must-revalidate
pragma: no-cache
set-cookie: PHPSESSID=q121op0oo13mv3971vd4lca81g; path=/
access-control-allow-origin: null
access-control-allow-credentials: true
vary: Accept-Encoding
cf-cache-status: DYNAMIC
expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
report-to: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=nPSS6qQE%2BrxAQk3nsRyNlEISX2W5l3
tKWZg0XQpY7fCUADS%2F4Ro9rs0EDPjCznczsQrRPGEf0kRQKs8PD1x9E13QVvt%2Fjc0ruIzKl03yf78GOLtn1p%2F4aGKMPhzZwKockUbg%2
BV22U2zSzA%3D%3D"}],"group":"cf-nel","max_age":604800}
nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
server: cloudflare
cf-ray: 6aa77de7ada24b11-HYD
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400
```

1.3. CORS With Prefix Match

Reference	**Risk Rating**
CORS With Prefix Match	**Medium**
Tools Used	
Browser, CURL	
Vulnerability Description	
The vulnerability is CORS that makes an attacker to access domain resource using different domain.	
How It Was Discovered	
Manual Analysis - Use Origin header with the attacker domain along with some prefix accepted by the domain and pass it using curl.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_3/cors_3.php	
Consequences of not Fixing the Issue	
Account takeover and stealing credentials.	
Suggested Countermeasures	
Allow only trusted sites	
References	
https://portswigger.net/web-security/cors	

Proof of Concept

The proof of the above vulnerability.



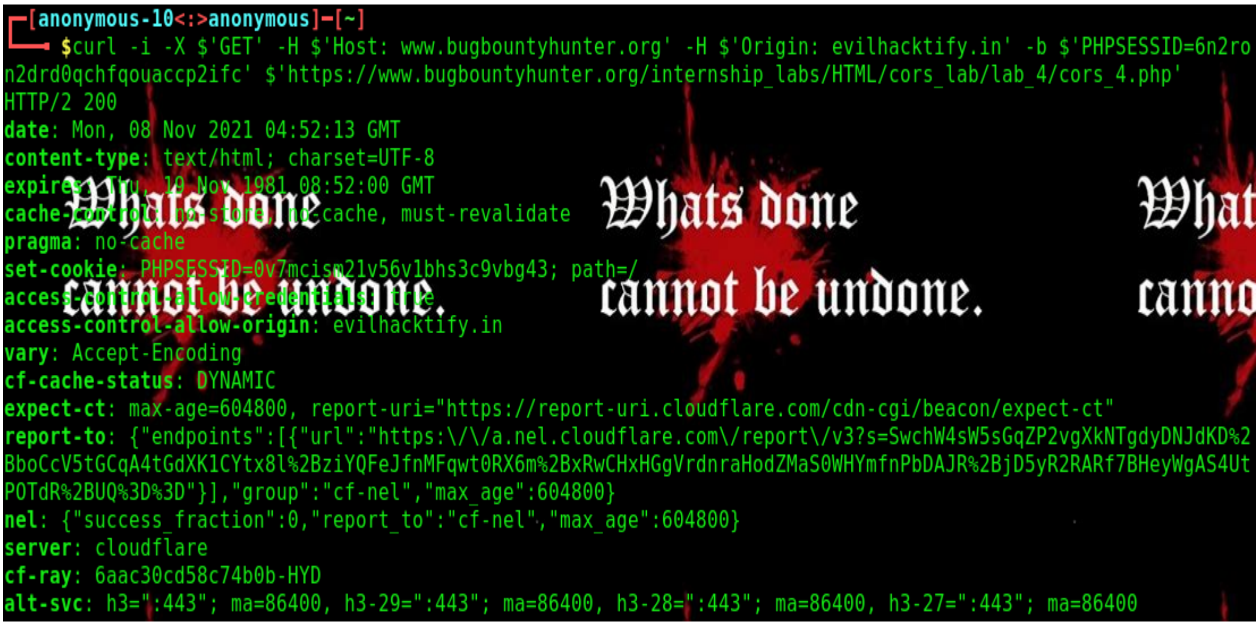
```
┌─[anonymous-10<:>anonymous]─[~]
└──╼ $curl -i -X $'GET' -H $'Host: www.bugbountyhunter.org' -H $'Origin: hacktify.in.attacker.com' -b $'PHPSES
SID=8d8fvj0442t3k03hk2hgaslmud' $'https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_3/cors_3.p
hp'
HTTP/2 200
date: Mon, 08 Nov 2021 04:49:27 GMT
content-type: text/html; charset=UTF-8
expires: Thu, 19 Nov 1981 08:52:00 GMT
cache-control: no-store, no-cache, must-revalidate
pragma: no-cache
set-cookie: PHPSESSID=cm6rjfic6ugqc0c9yq2c6v777r; path=/
access-control-allow-credentials: true
access-control-allow-origin: hacktify.in.attacker.com
vary: Accept-Encoding
cf-cache-status: DYNAMIC
expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
report-to: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=mVNZiOdyg5wnD0VGfzI0OcWmMRyV%2Ff
DT8ZjTwdUNh7prcprWygqpt8Ask2Os%2BUmdVvf6M612QTjRXqSBo0fPYxNP99pxPKqWGSAlvLw1gKc%2FaURkRK%2BdNeLv5r0svs6%2Bry2d
fEyXJ3Qbhg%3D%3D"}],"group":"cf-nel","max_age":604800}
nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
server: cloudflare
cf-ray: 6aac2cbc89a54b0b-HYD
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400
```

1.4. CORS With Suffix Match

Reference	**Risk Rating**
CORS With Suffix Match	**Medium**
Tools Used	
Browser, CURL	
Vulnerability Description	
The vulnerability is CORS that makes an attacker to access domain resource using different domain.	
How It Was Discovered	
Manual Analysis - Use Origin header with the attacker domain along with some suffix accepted by the domain and pass it using curl.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_4/cors_4.php	
Consequences of not Fixing the Issue	
Account takeover and stealing credentials.	
Suggested Countermeasures	
Allow only trusted sites	
References	
https://portswigger.net/web-security/cors	

Proof of Concept

The proof of the above vulnerability.



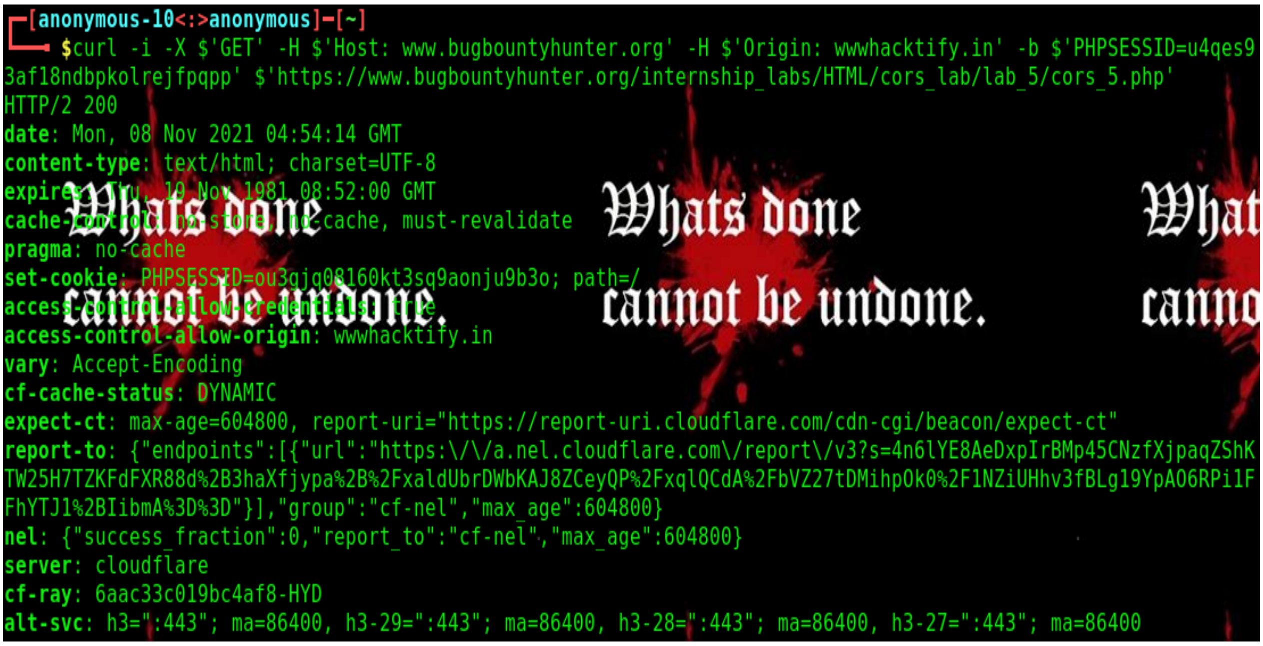
```
┌─[anonymous-10<:>anonymous]─[~]
└──╼ $curl -i -X $'GET' -H $'Host: www.bugbountyhunter.org' -H $'Origin: evilhacktify.in' -b $'PHPSESSID=6n2ro
n2drd0qchfqouaccp2ifc' $'https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_4/cors_4.php'
HTTP/2 200
date: Mon, 08 Nov 2021 04:52:13 GMT
content-type: text/html; charset=UTF-8
expires: Thu, 19 Nov 1981 08:52:00 GMT
cache-control: no-store, no-cache, must-revalidate
pragma: no-cache
set-cookie: PHPSESSID=0v7mcism21v56v1bhs3c9vbg43; path=/
access-control-allow-credentials: true
access-control-allow-origin: evilhacktify.in
vary: Accept-Encoding
cf-cache-status: DYNAMIC
expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
report-to: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=SwchW4sW5sGqZP2vgXkNTgdyDNJdKD%2
BboCcV5tGCqA4tGdXK1CYtx8l%2BziYQFeJfnMFqwt0RX6m%2BxRwCHxHGgVrdnraHodZMaS0WHYmfnPbDAJR%2BjD5yR2RARf7BHeyWgAS4Ut
POTdR%2BUQ%3D%3D"}],"group":"cf-nel","max_age":604800}
nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
server: cloudflare
cf-ray: 6aac30cd58c74b0b-HYD
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400
```

1.5. CORS With Escape Dot

Reference	**Risk Rating**
CORS With Escape Dot	**High**
Tools Used	
Browser, CURL	
Vulnerability Description	
The vulnerability is CORS that makes an attacker to access domain resource using different domain.	
How It Was Discovered	
Manual Analysis - Use Origin header with the attacker domain without dot and pass it using curl.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_5/cors_5.php	
Consequences of not Fixing the Issue	
Account takeover and stealing credentials.	
Suggested Countermeasures	
Allow only trusted sites	
References	
https://portswigger.net/web-security/cors	

Proof of Concept

The proof of the above vulnerability.



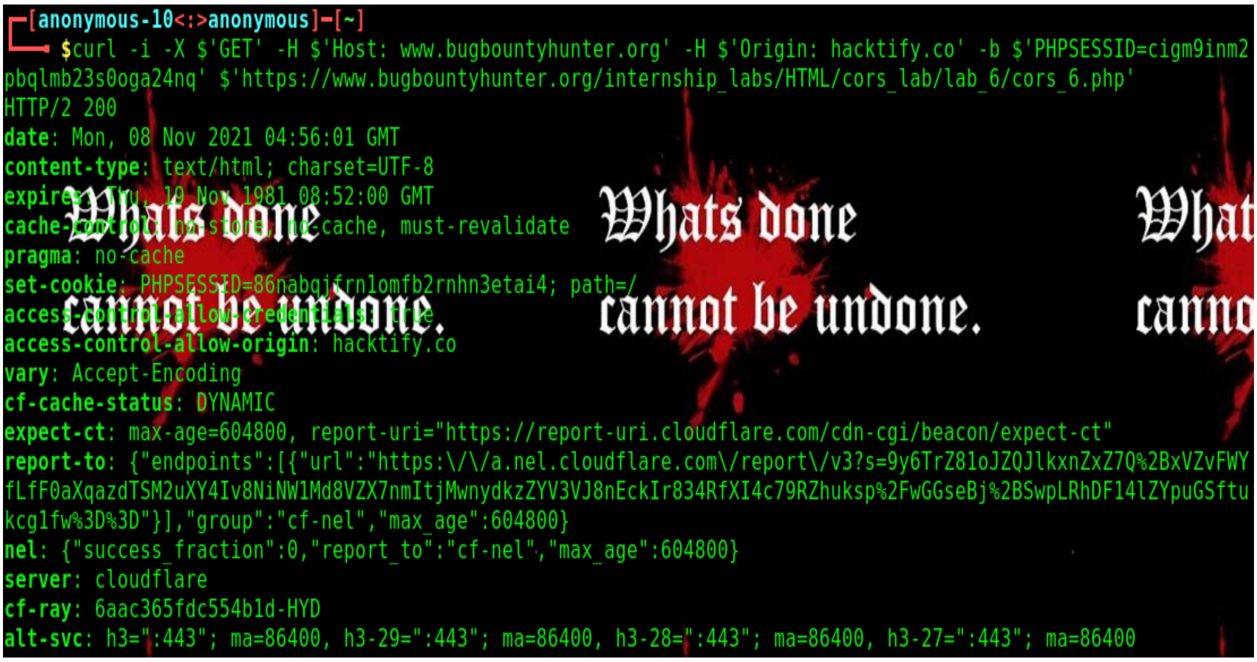
```
┌─[anonymous-10<:>anonymous]─[~]
└──╼ $curl -i -X $'GET' -H $'Host: www.bugbountyhunter.org' -H $'Origin: wwwhacktify.in' -b $'PHPSESSID=u4qes9
3af18ndbpkolrejfpqpp' $'https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_5/cors_5.php'
HTTP/2 200
date: Mon, 08 Nov 2021 04:54:14 GMT
content-type: text/html; charset=UTF-8
expires: Thu, 19 Nov 1981 08:52:00 GMT
cache-control: no-store, no-cache, must-revalidate
pragma: no-cache
set-cookie: PHPSESSID=ou3gjq08160kt3sq9aonju9b3o; path=/
access-control-allow-credentials: true
access-control-allow-origin: wwwhacktify.in
vary: Accept-Encoding
cf-cache-status: DYNAMIC
expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
report-to: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=4n6lYE8AeDxpIrBMp45CNzfXjpaqZShK
TW25H7TZKFdFXR88d%2B3haXfjypa%2B%2FxaldUbrDWbKAJ8ZCeyQP%2FxqlQCdA%2FbVZ27tDMihpOk0%2F1NZiUHhv3fBLg19YpAO6RPi1F
FhYTJ1%2BIibmA%3D%3D"}],"group":"cf-nel","max_age":604800}
nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
server: cloudflare
cf-ray: 6aac33c019bc4af8-HYD
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400
```

1.6. CORS With Substring Match

Reference	Risk Rating
CORS With Substring Match	**High**
Tools Used	
Browser, CURL	
Vulnerability Description	
The vulnerability is CORS that makes an attacker to access domain resource using different domain.	
How It Was Discovered	
Manual Analysis - Use Origin header with the attacker domain along with some substring accepted by the domain and pass it using curl.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_6/cors_6.php	
Consequences of not Fixing the Issue	
Account takeover and stealing credentials.	
Suggested Countermeasures	
Allow only trusted sites	
References	
https://portswigger.net/web-security/cors	

Proof of Concept

The proof of the above vulnerability.



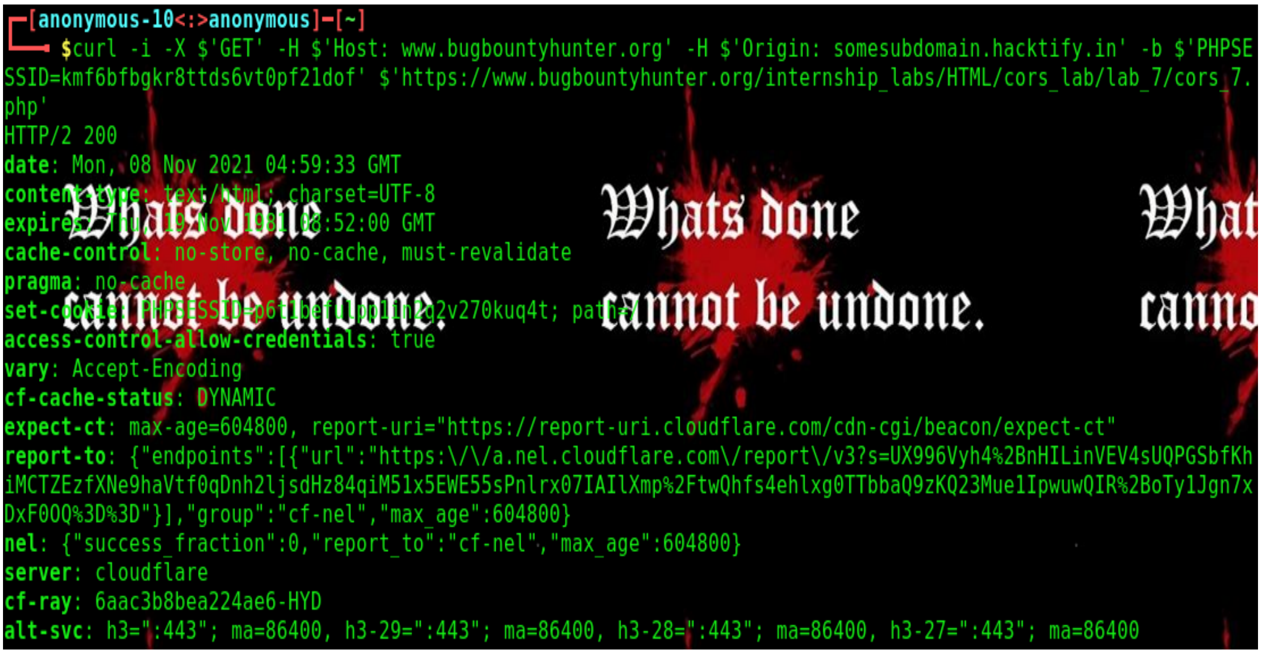
```
┌─[anonymous-10<:>anonymous]─[~]
└──╼ $curl -i -X $'GET' -H $'Host: www.bugbountyhunter.org' -H $'Origin: hacktify.co' -b $'PHPSESSID=cigm9inm2
pbqlmb23s0oga24nq' $'https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_6/cors_6.php'
HTTP/2 200
date: Mon, 08 Nov 2021 04:56:01 GMT
content-type: text/html; charset=UTF-8
expires: Thu, 19 Nov 1981 08:52:00 GMT
cache-control: no-store, no-cache, must-revalidate
pragma: no-cache
set-cookie: PHPSESSID=86nabqjfrnlomfb2rnhn3etai4; path=/
access-control-allow-credentials: true
access-control-allow-origin: hacktify.co
vary: Accept-Encoding
cf-cache-status: DYNAMIC
expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
report-to: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=9y6TrZ81oJZQJlkxnZxZ7Q%2BxVZvFWY
fLfF0aXqazdTSM2uXY4Iv8NiNW1Md8VZX7nmItjMwnydkzZYV3VJ8nEckIr834RfXI4c79RZhuksp%2FwGGseBj%2BSwpLRhDF14lZYpuGSftu
kcg1fw%3D%3D"}],"group":"cf-nel","max_age":604800}
nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
server: cloudflare
cf-ray: 6aac365fdc554b1d-HYD
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400
```

1.7. CORS With Arbitrary Subdomain

Reference	**Risk Rating**
CORS With Arbitrary Subdomain	**High**
Tools Used	
Browser, CURL	
Vulnerability Description	
The vulnerability is CORS that makes an attacker to access domain resource using different domain.	
How It Was Discovered	
Manual Analysis - Use Origin header with the attacker domain along the subdomain of the trusted domain and pass it using curl.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_7/cors_7.php	
Consequences of not Fixing the Issue	
Account takeover and stealing credentials.	
Suggested Countermeasures	
Allow only trusted sites	
References	
https://portswigger.net/web-security/cors	

Proof of Concept

The proof of the above vulnerability.



```
┌─[anonymous-10<:>anonymous]─[~]
└──╼ $curl -i -X $'GET' -H $'Host: www.bugbountyhunter.org' -H $'Origin: somesubdomain.hacktify.in' -b $'PHPSE
SSID=kmf6bfbgkr8ttds6vt0pf21dof' $'https://www.bugbountyhunter.org/internship_labs/HTML/cors_lab/lab_7/cors_7.
php'
HTTP/2 200
date: Mon, 08 Nov 2021 04:59:33 GMT
content-type: text/html; charset=UTF-8
expires: Thu, 19 Nov 1981 08:52:00 GMT
cache-control: no-store, no-cache, must-revalidate
pragma: no-cache
set-cookie: PHPSESSID=p6t1bef1lpp1in2q2v270kuq4t; path=/
access-control-allow-credentials: true
vary: Accept-Encoding
cf-cache-status: DYNAMIC
expect-ct: max-age=604800, report-uri="https://report-uri.cloudflare.com/cdn-cgi/beacon/expect-ct"
report-to: {"endpoints":[{"url":"https:\/\/a.nel.cloudflare.com\/report\/v3?s=UX996Vyh4%2BnHILinVEV4sUQPGSbfKh
iMCTZEzfXNe9haVtf0qDnh2ljsdHz84qiM51x5EWE55sPnlrx07IAIlXmp%2FtwQhfs4ehlxg0TTbbaQ9zKQ23Mue1IpwuwQIR%2BoTy1Jgn7x
DxF0OQ%3D%3D"}],"group":"cf-nel","max_age":604800}
nel: {"success_fraction":0,"report_to":"cf-nel","max_age":604800}
server: cloudflare
cf-ray: 6aac3b8bea224ae6-HYD
alt-svc: h3=":443"; ma=86400, h3-29=":443"; ma=86400, h3-28=":443"; ma=86400, h3-27=":443"; ma=86400
```