Penetration Testing Report

Full Name: Muhammad Bokhtiar Uddin
Program: HCS - Penetration Testing Internship Week-3
Date: 09/03/2024

Introduction

This report document hereby describes the proceedings and results of a Black Box security assessment conducted against the **Week-3 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-3 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-Site Request Forgery, Server-Side Request Forgery**

3. Summary

Outlined is a Black Box Application Security assessment for the **Week-3 Labs**.

Total number of Sub-labs: 15 Sub-labs

High	Medium	Low
6	5	4

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

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

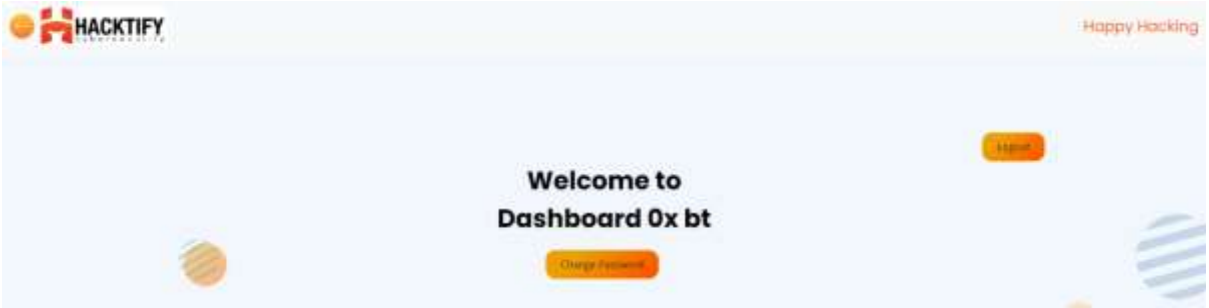
Low - **Number of Sub-labs with Easy difficulty level**

1. Cross-Site Request Forgery

1.1. Eassyy CSRF

Reference	Risk Rating
Eassyy CSRF	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Cross-Site Request Forgery (CSRF) is a type of cyber-attack where an attacker tricks a user into unintentionally executing actions on a website they are authenticated on. By exploiting the trust, a website has in a user's browser, the attacker can forge requests that seem legitimate, like transferring funds or changing account settings. This attack occurs when a user visits a malicious website or opens a malicious email, allowing the attacker to perform actions on behalf of the user without their knowledge. To mitigate CSRF attacks, developers implement measures like CSRF tokens, which are unique, random values included with each request to validate its authenticity.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/csrf_lab/lab_1/ lab_1.php	
Consequences of not Fixing the Issue	
Escalation: Problems left unaddressed tend to worsen over time. What might have been a small issue can grow into a larger, more complex problem. **Damage to Reputation:** Ignoring problems can damage your reputation, whether it's in personal relationships, professional settings, or businesses. Trust and respect can diminish. **Missed Opportunities:** Avoiding problems often means missing out on opportunities for growth, improvement, or learning from mistakes. **Strained Relationships:** Whether it's with friends, family, colleagues, or customers, unresolved issues can strain relationships and lead to misunderstandings or conflicts. **Legal Consequences:** Some issues, especially in legal or regulatory contexts, can lead to serious legal consequences if not addressed properly.	
Suggested Countermeasures	
Proactive Communication: Open communication channels should be fostered in personal relationships, work teams, and organizations to facilitate open discussions and raise concerns or issues. **Regular Assessments and Reviews:** Regular evaluations, such as work performance reviews, family check-ins, or business audits, can help identify potential issues early on. **Training and Development:** Invest in training programs to equip individuals with problem-solving skills, such as conflict resolution, leadership development, or personal growth workshops. **Establish Clear Policies and Procedures:** Clear guidelines are crucial for efficient issue resolution, including reporting protocols, escalation procedures, and resolution steps.	
References	
https://portswigger.net/web-security/csrf	

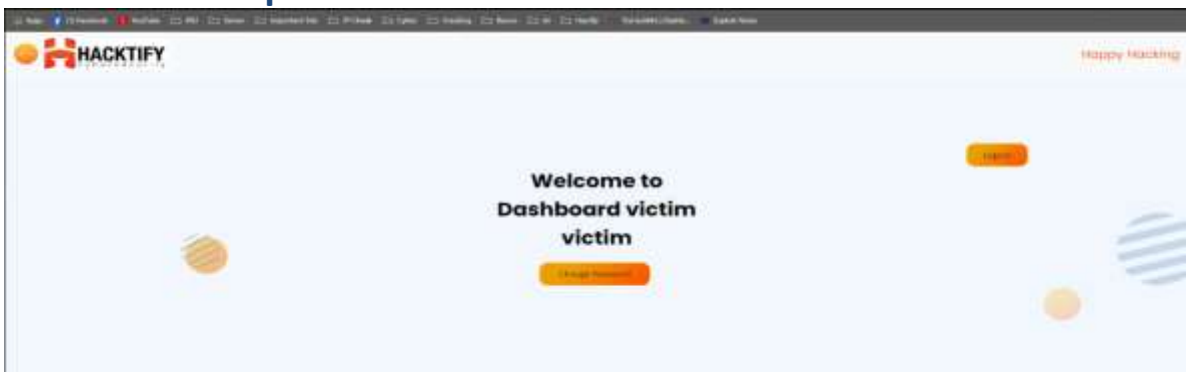
Proof of Concept



1.2. Always Validate Tokens

Reference	Risk Rating
Always Validate Token	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Cross-Site Request Forgery (CSRF) is a type of cyber-attack where an attacker tricks a user into unintentionally executing actions on a website they are authenticated on. By exploiting the trust, a website has in a user's browser, the attacker can forge requests that seem legitimate, like transferring funds or changing account settings. This attack occurs when a user visits a malicious website or opens a malicious email, allowing the attacker to perform actions on behalf of the user without their knowledge. To mitigate CSRF attacks, developers implement measures like CSRF tokens, which are unique, random values included with each request to validate its authenticity.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/csrf_lab/lab_2/ lab_2.php	
Consequences of not Fixing the Issue	
Escalation: Problems left unaddressed tend to worsen over time. What might have been a small issue can grow into a larger, more complex problem. **Damage to Reputation:** Ignoring problems can damage your reputation, whether it's in personal relationships, professional settings, or businesses. Trust and respect can diminish. **Missed Opportunities:** Avoiding problems often means missing out on opportunities for growth, improvement, or learning from mistakes. **Strained Relationships:** Whether it's with friends, family, colleagues, or customers, unresolved issues can strain relationships and lead to misunderstandings or conflicts. **Legal Consequences:** Some issues, especially in legal or regulatory contexts, can lead to serious legal consequences if not addressed properly.	
Suggested Countermeasures	
Proactive Communication: Open communication channels should be fostered in personal relationships, work teams, and organizations to facilitate open discussions and raise concerns or issues. **Regular Assessments and Reviews:** Regular evaluations, such as work performance reviews, family check-ins, or business audits, can help identify potential issues early on. **Training and Development:** Invest in training programs to equip individuals with problem-solving skills, such as conflict resolution, leadership development, or personal growth workshops. **Establish Clear Policies and Procedures:** Clear guidelines are crucial for efficient issue resolution, including reporting protocols, escalation procedures, and resolution steps.	
References	
https://portswigger.net/web-security/csrf	

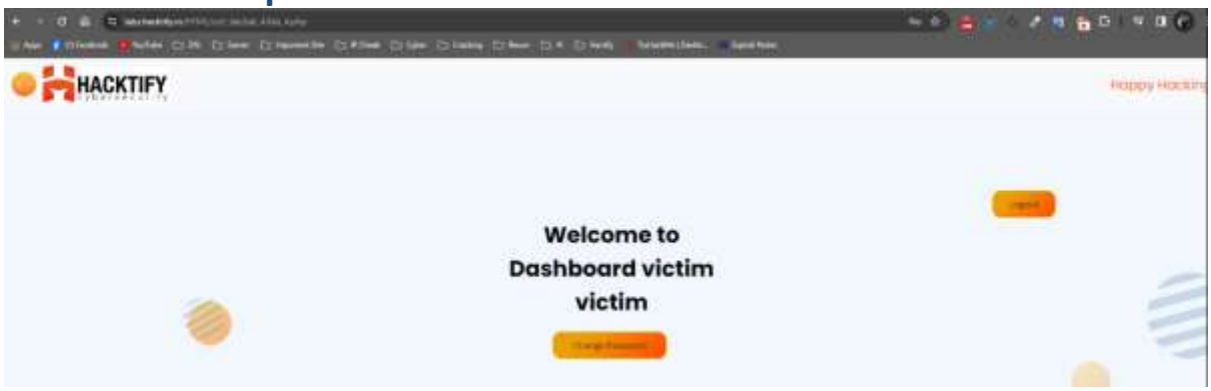
Proof of Concept



1.3. I Hate Someone Uses My Token

Reference	Risk Rating
I Hate Someone Uses My Token	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Cross-Site Request Forgery (CSRF) is a type of cyber-attack where an attacker tricks a user into unintentionally executing actions on a website they are authenticated on. By exploiting the trust, a website has in a user's browser, the attacker can forge requests that seem legitimate, like transferring funds or changing account settings. This attack occurs when a user visits a malicious website or opens a malicious email, allowing the attacker to perform actions on behalf of the user without their knowledge. To mitigate CSRF attacks, developers implement measures like CSRF tokens, which are unique, random values included with each request to validate its authenticity.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/csrf_lab/lab_3/ lab_3.php	
Consequences of not Fixing the Issue	
Escalation: Problems left unaddressed tend to worsen over time. What might have been a small issue can grow into a larger, more complex problem. **Damage to Reputation:** Ignoring problems can damage your reputation, whether it's in personal relationships, professional settings, or businesses. Trust and respect can diminish. **Missed Opportunities:** Avoiding problems often means missing out on opportunities for growth, improvement, or learning from mistakes. **Strained Relationships:** Whether it's with friends, family, colleagues, or customers, unresolved issues can strain relationships and lead to misunderstandings or conflicts. **Legal Consequences:** Some issues, especially in legal or regulatory contexts, can lead to serious legal consequences if not addressed properly.	
Suggested Countermeasures	
Proactive Communication: Open communication channels should be fostered in personal relationships, work teams, and organizations to facilitate open discussions and raise concerns or issues. **Regular Assessments and Reviews:** Regular evaluations, such as work performance reviews, family check-ins, or business audits, can help identify potential issues early on. **Training and Development:** Invest in training programs to equip individuals with problem-solving skills, such as conflict resolution, leadership development, or personal growth workshops. **Establish Clear Policies and Procedures:** Clear guidelines are crucial for efficient issue resolution, including reporting protocols, escalation procedures, and resolution steps.	
References	
https://portswigger.net/web-security/csrf	

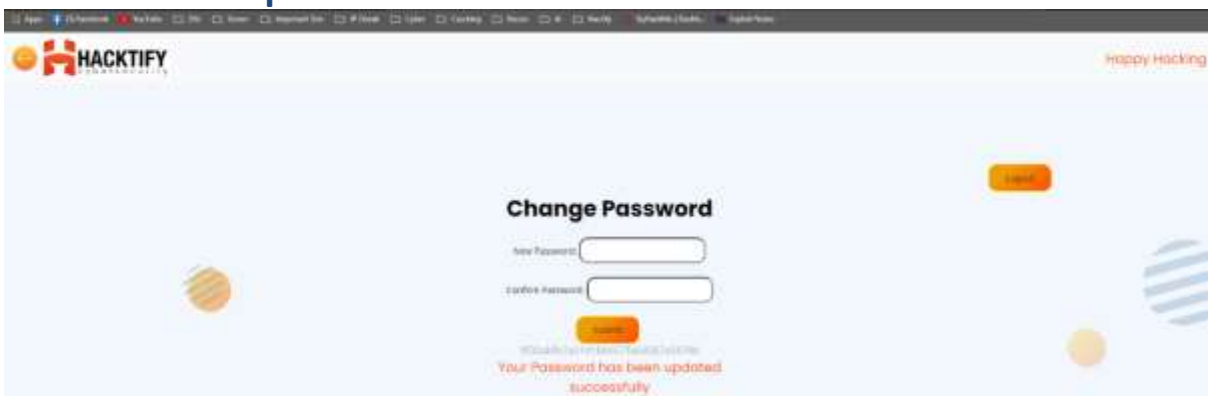
Proof of Concept



1.4. GET Me Or POST ME

Reference	Risk Rating
GET Me Or POST Me	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Cross-Site Request Forgery (CSRF) is a type of cyber-attack where an attacker tricks a user into unintentionally executing actions on a website they are authenticated on. By exploiting the trust, a website has in a user's browser, the attacker can forge requests that seem legitimate, like transferring funds or changing account settings. This attack occurs when a user visits a malicious website or opens a malicious email, allowing the attacker to perform actions on behalf of the user without their knowledge. To mitigate CSRF attacks, developers implement measures like CSRF tokens, which are unique, random values included with each request to validate its authenticity.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/csrf_lab/lab_4/ lab_4.php	
Consequences of not Fixing the Issue	
Escalation: Problems left unaddressed tend to worsen over time. What might have been a small issue can grow into a larger, more complex problem. **Damage to Reputation:** Ignoring problems can damage your reputation, whether it's in personal relationships, professional settings, or businesses. Trust and respect can diminish. **Missed Opportunities:** Avoiding problems often means missing out on opportunities for growth, improvement, or learning from mistakes. **Strained Relationships:** Whether it's with friends, family, colleagues, or customers, unresolved issues can strain relationships and lead to misunderstandings or conflicts. **Legal Consequences:** Some issues, especially in legal or regulatory contexts, can lead to serious legal consequences if not addressed properly.	
Suggested Countermeasures	
Proactive Communication: Open communication channels should be fostered in personal relationships, work teams, and organizations to facilitate open discussions and raise concerns or issues. **Regular Assessments and Reviews:** Regular evaluations, such as work performance reviews, family check-ins, or business audits, can help identify potential issues early on. **Training and Development:** Invest in training programs to equip individuals with problem-solving skills, such as conflict resolution, leadership development, or personal growth workshops. **Establish Clear Policies and Procedures:** Clear guidelines are crucial for efficient issue resolution, including reporting protocols, escalation procedures, and resolution steps.	
References	
https://portswigger.net/web-security/csrf	

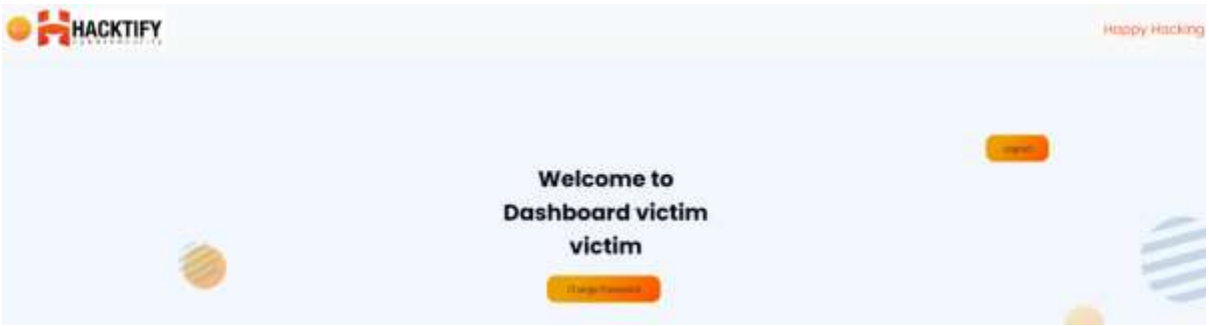
Proof of Concept



1.5. XSS The Savior

Reference	Risk Rating
XSS The Savior	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Cross-Site Request Forgery (CSRF) is a type of cyber-attack where an attacker tricks a user into unintentionally executing actions on a website they are authenticated on. By exploiting the trust, a website has in a user's browser, the attacker can forge requests that seem legitimate, like transferring funds or changing account settings. This attack occurs when a user visits a malicious website or opens a malicious email, allowing the attacker to perform actions on behalf of the user without their knowledge. To mitigate CSRF attacks, developers implement measures like CSRF tokens, which are unique, random values included with each request to validate its authenticity.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/csrf_lab/lab_5/lab_5.php	
Consequences of not Fixing the Issue	
Escalation: Problems left unaddressed tend to worsen over time. What might have been a small issue can grow into a larger, more complex problem. **Damage to Reputation:** Ignoring problems can damage your reputation, whether it's in personal relationships, professional settings, or businesses. Trust and respect can diminish. **Missed Opportunities:** Avoiding problems often means missing out on opportunities for growth, improvement, or learning from mistakes. **Strained Relationships:** Whether it's with friends, family, colleagues, or customers, unresolved issues can strain relationships and lead to misunderstandings or conflicts. **Legal Consequences:** Some issues, especially in legal or regulatory contexts, can lead to serious legal consequences if not addressed properly.	
Suggested Countermeasures	
Proactive Communication: Open communication channels should be fostered in personal relationships, work teams, and organizations to facilitate open discussions and raise concerns or issues. **Regular Assessments and Reviews:** Regular evaluations, such as work performance reviews, family check-ins, or business audits, can help identify potential issues early on. **Training and Development:** Invest in training programs to equip individuals with problem-solving skills, such as conflict resolution, leadership development, or personal growth workshops. **Establish Clear Policies and Procedures:** Clear guidelines are crucial for efficient issue resolution, including reporting protocols, escalation procedures, and resolution steps.	
References	
https://portswigger.net/web-security/csrf	

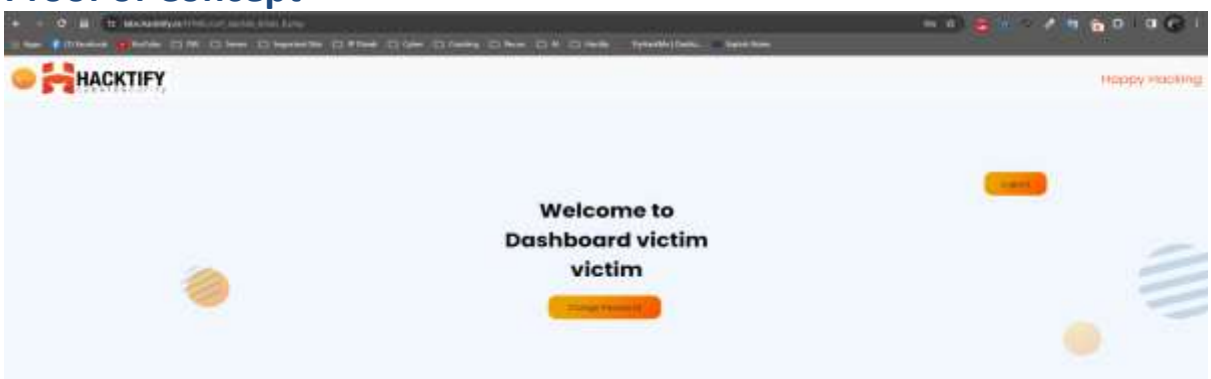
Proof of Concept



1.6. Rm -Rf Token

Reference	Risk Rating
Rm -Rf Token	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Cross-Site Request Forgery (CSRF) is a type of cyber-attack where an attacker tricks a user into unintentionally executing actions on a website they are authenticated on. By exploiting the trust, a website has in a user's browser, the attacker can forge requests that seem legitimate, like transferring funds or changing account settings. This attack occurs when a user visits a malicious website or opens a malicious email, allowing the attacker to perform actions on behalf of the user without their knowledge. To mitigate CSRF attacks, developers implement measures like CSRF tokens, which are unique, random values included with each request to validate its authenticity.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/csrf_lab/lab_6/lab_6.php	
Consequences of not Fixing the Issue	
Escalation: Problems left unaddressed tend to worsen over time. What might have been a small issue can grow into a larger, more complex problem. **Damage to Reputation:** Ignoring problems can damage your reputation, whether it's in personal relationships, professional settings, or businesses. Trust and respect can diminish. **Missed Opportunities:** Avoiding problems often means missing out on opportunities for growth, improvement, or learning from mistakes. **Strained Relationships:** Whether it's with friends, family, colleagues, or customers, unresolved issues can strain relationships and lead to misunderstandings or conflicts. **Legal Consequences:** Some issues, especially in legal or regulatory contexts, can lead to serious legal consequences if not addressed properly.	
Suggested Countermeasures	
Proactive Communication: Open communication channels should be fostered in personal relationships, work teams, and organizations to facilitate open discussions and raise concerns or issues. **Regular Assessments and Reviews:** Regular evaluations, such as work performance reviews, family check-ins, or business audits, can help identify potential issues early on. **Training and Development:** Invest in training programs to equip individuals with problem-solving skills, such as conflict resolution, leadership development, or personal growth workshops. **Establish Clear Policies and Procedures:** Clear guidelines are crucial for efficient issue resolution, including reporting protocols, escalation procedures, and resolution steps.	
References	
https://portswigger.net/web-security/csrf	

Proof of Concept



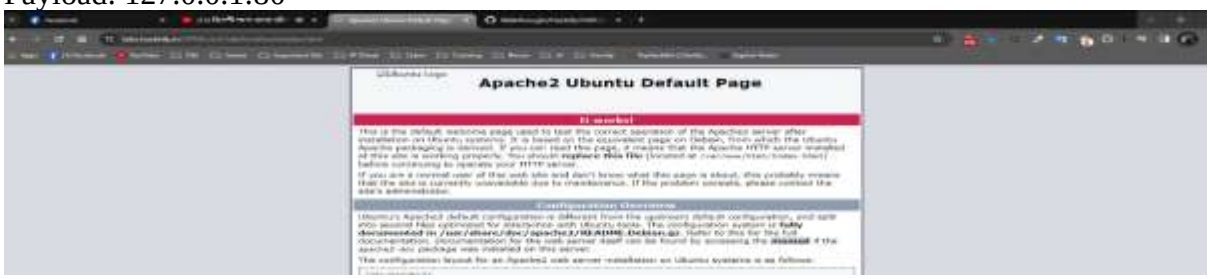
2. Server-Site Request Forgey

2.1. GET The 127.0.0.1!

Reference	Risk Rating
Get The 127.0.0.1	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Server-Side Request Forgery (SSRF) is a security vulnerability where an attacker manipulates an application to send unauthorized requests, potentially leading to data theft, server-side attacks, and access to internal systems. Web applications and APIs are particularly vulnerable, necessitating thorough input validation and access controls.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/ssrf_lab/lab_1/lab_1.php	
Consequences of not Fixing the Issue	
Data Breaches: Unresolved SSRF vulnerabilities can allow unauthorized access to internal systems' sensitive data, allowing attackers to steal confidential information like customer records, financial data, or intellectual property. **Financial Losses:** The theft of sensitive data can result in financial losses for the organization, including costs associated with legal fees, regulatory fines, and compensation for affected individuals. **Reputation Damage:** SSRF-related data breaches can severely harm an organization's reputation, potentially leading to customer trust loss, as customers may choose to relocate. **Network Compromise:** SSRF can serve as a pivot point for attackers, potentially compromising entire systems, disrupting services, and causing loss of critical infrastructure control. **Legal Ramifications:** Non-compliance with data protection regulations like GDPR and CCPA can lead to legal consequences for organizations failing to protect against SSRF attacks.	
Suggested Countermeasures	
Input Validation: Validate and sanitize user-supplied URLs to ensure they are legitimate and safe. **Whitelist Allowed Hosts:** Restrict the domains or IP addresses that the application can access, allowing only necessary and trusted resources. **Use of Redirection URLs:** Use a whitelist of allowed redirection URLs to prevent open redirects. **Firewall Configuration:** Configure firewalls to block requests to internal network resources from the web server.	
References	
https://portswigger.net/web-security/ssrf	

Proof of Concept

Payload: 127.0.0.1:80

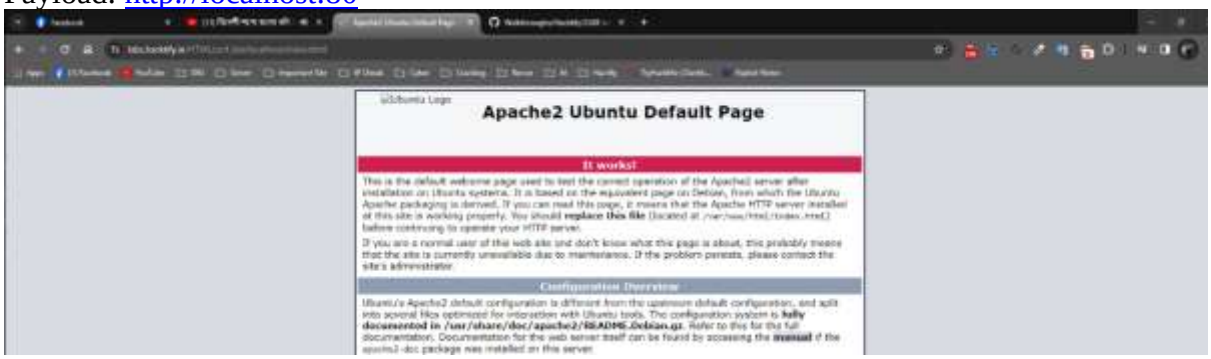


2.2. Http(S)? Nevermind!!

Reference	Risk Rating
Http(S)? Nevermind!	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Server-Side Request Forgery (SSRF) is a security vulnerability where an attacker manipulates an application to send unauthorized requests, potentially leading to data theft, server-side attacks, and access to internal systems. Web applications and APIs are particularly vulnerable, necessitating thorough input validation and access controls.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/ssrf_lab/lab_2/lab_2.php	
Consequences of not Fixing the Issue	
Data Breaches: Unresolved SSRF vulnerabilities can allow unauthorized access to internal systems' sensitive data, allowing attackers to steal confidential information like customer records, financial data, or intellectual property. **Financial Losses:** The theft of sensitive data can result in financial losses for the organization, including costs associated with legal fees, regulatory fines, and compensation for affected individuals. **Reputation Damage:** SSRF-related data breaches can severely harm an organization's reputation, potentially leading to customer trust loss, as customers may choose to relocate. **Network Compromise:** SSRF can serve as a pivot point for attackers, potentially compromising entire systems, disrupting services, and causing loss of critical infrastructure control. **Legal Ramifications:** Non-compliance with data protection regulations like GDPR and CCPA can lead to legal consequences for organizations failing to protect against SSRF attacks.	
Suggested Countermeasures	
Input Validation: Validate and sanitize user-supplied URLs to ensure they are legitimate and safe. **Whitelist Allowed Hosts:** Restrict the domains or IP addresses that the application can access, allowing only necessary and trusted resources. **Use of Redirection URLs:** Use a whitelist of allowed redirection URLs to prevent open redirects. **Firewall Configuration:** Configure firewalls to block requests to internal network resources from the web server.	
References	
https://portswigger.net/web-security/ssrf	

Proof of Concept

Payload: http://localhost:80

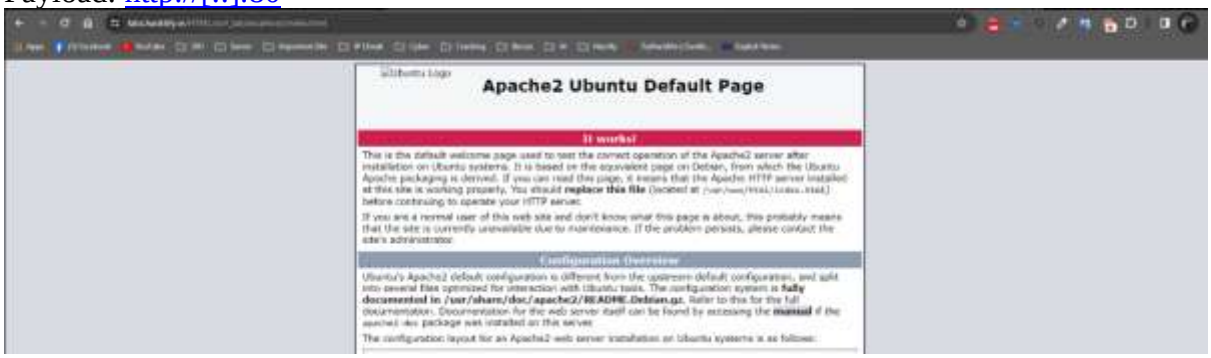


2.3. ":" The Saviour!

Reference	Risk Rating
":" The Saviour!	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Server-Side Request Forgery (SSRF) is a security vulnerability where an attacker manipulates an application to send unauthorized requests, potentially leading to data theft, server-side attacks, and access to internal systems. Web applications and APIs are particularly vulnerable, necessitating thorough input validation and access controls.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/ssrf_lab/lab_3/lab_3.php	
Consequences of not Fixing the Issue	
Data Breaches: Unresolved SSRF vulnerabilities can allow unauthorized access to internal systems' sensitive data, allowing attackers to steal confidential information like customer records, financial data, or intellectual property. **Financial Losses:** The theft of sensitive data can result in financial losses for the organization, including costs associated with legal fees, regulatory fines, and compensation for affected individuals. **Reputation Damage:** SSRF-related data breaches can severely harm an organization's reputation, potentially leading to customer trust loss, as customers may choose to relocate. **Network Compromise:** SSRF can serve as a pivot point for attackers, potentially compromising entire systems, disrupting services, and causing loss of critical infrastructure control. **Legal Ramifications:** Non-compliance with data protection regulations like GDPR and CCPA can lead to legal consequences for organizations failing to protect against SSRF attacks.	
Suggested Countermeasures	
Input Validation: Validate and sanitize user-supplied URLs to ensure they are legitimate and safe. **Whitelist Allowed Hosts:** Restrict the domains or IP addresses that the application can access, allowing only necessary and trusted resources. **Use of Redirection URLs:** Use a whitelist of allowed redirection URLs to prevent open redirects. **Firewall Configuration:** Configure firewalls to block requests to internal network resources from the web server.	
References	
https://portswigger.net/web-security/ssrf	

Proof of Concept

Payload: http://[::]:80

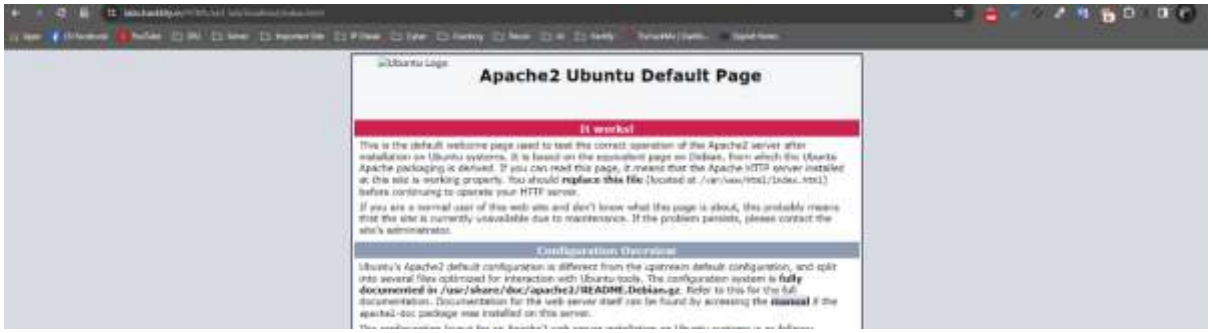


2.4. Messed Up Domain!

Reference	Risk Rating
Messed Up Domain!	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Server-Side Request Forgery (SSRF) is a security vulnerability where an attacker manipulates an application to send unauthorized requests, potentially leading to data theft, server-side attacks, and access to internal systems. Web applications and APIs are particularly vulnerable, necessitating thorough input validation and access controls.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/ssrf_lab/lab_4/lab_4.php	
Consequences of not Fixing the Issue	
Data Breaches: Unresolved SSRF vulnerabilities can allow unauthorized access to internal systems' sensitive data, allowing attackers to steal confidential information like customer records, financial data, or intellectual property. **Financial Losses:** The theft of sensitive data can result in financial losses for the organization, including costs associated with legal fees, regulatory fines, and compensation for affected individuals. **Reputation Damage:** SSRF-related data breaches can severely harm an organization's reputation, potentially leading to customer trust loss, as customers may choose to relocate. **Network Compromise:** SSRF can serve as a pivot point for attackers, potentially compromising entire systems, disrupting services, and causing loss of critical infrastructure control. **Legal Ramifications:** Non-compliance with data protection regulations like GDPR and CCPA can lead to legal consequences for organizations failing to protect against SSRF attacks.	
Suggested Countermeasures	
Input Validation: Validate and sanitize user-supplied URLs to ensure they are legitimate and safe. **Whitelist Allowed Hosts:** Restrict the domains or IP addresses that the application can access, allowing only necessary and trusted resources. **Use of Redirection URLs:** Use a whitelist of allowed redirection URLs to prevent open redirects. **Firewall Configuration:** Configure firewalls to block requests to internal network resources from the web server.	
References	
https://portswigger.net/web-security/ssrf	

Proof of Concept

Payload: http://customer1.app.localhost.my.company.127.0.0.1.nip.io

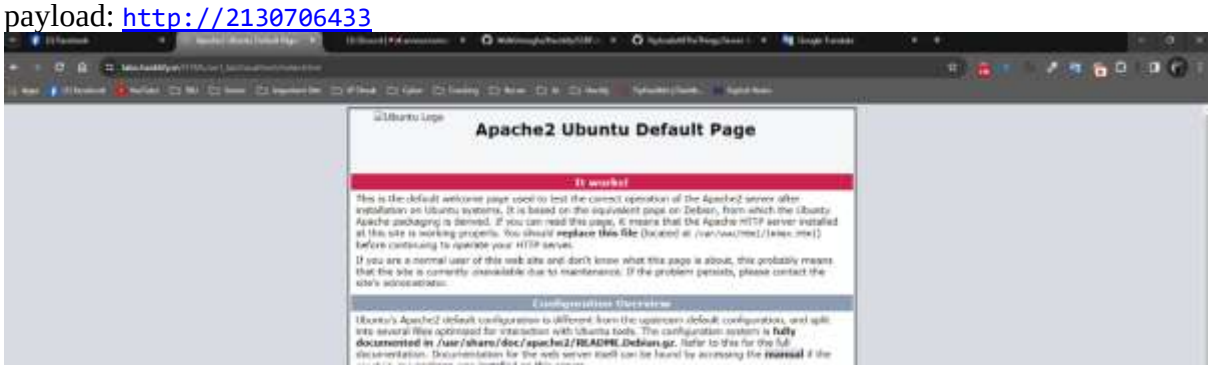


2.5. Decimal IP

Reference	Risk Rating
Decimal IP	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Server-Side Request Forgery (SSRF) is a security vulnerability where an attacker manipulates an application to send unauthorized requests, potentially leading to data theft, server-side attacks, and access to internal systems. Web applications and APIs are particularly vulnerable, necessitating thorough input validation and access controls.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/ssrf_lab/lab_5/lab_5.php	
Consequences of not Fixing the Issue	
Data Breaches: Unresolved SSRF vulnerabilities can allow unauthorized access to internal systems' sensitive data, allowing attackers to steal confidential information like customer records, financial data, or intellectual property. **Financial Losses:** The theft of sensitive data can result in financial losses for the organization, including costs associated with legal fees, regulatory fines, and compensation for affected individuals. **Reputation Damage:** SSRF-related data breaches can severely harm an organization's reputation, potentially leading to customer trust loss, as customers may choose to relocate. **Network Compromise:** SSRF can serve as a pivot point for attackers, potentially compromising entire systems, disrupting services, and causing loss of critical infrastructure control. **Legal Ramifications:** Non-compliance with data protection regulations like GDPR and CCPA can lead to legal consequences for organizations failing to protect against SSRF attacks.	
Suggested Countermeasures	
Input Validation: Validate and sanitize user-supplied URLs to ensure they are legitimate and safe. **Whitelist Allowed Hosts:** Restrict the domains or IP addresses that the application can access, allowing only necessary and trusted resources. **Use of Redirection URLs:** Use a whitelist of allowed redirection URLs to prevent open redirects. **Firewall Configuration:** Configure firewalls to block requests to internal network resources from the web server.	
References	
https://portswigger.net/web-security/ssrf	

Proof of Concept

payload: `http://2130706433`

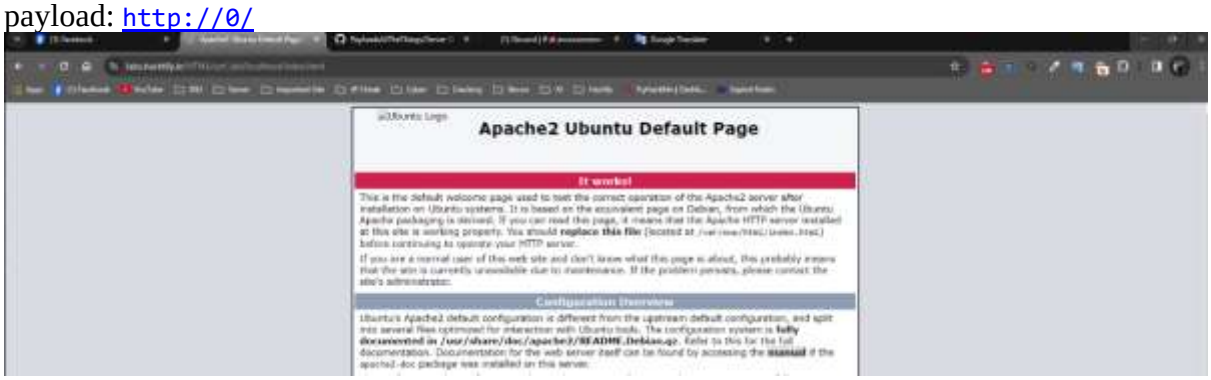


2.6. Short-Hand IP Address

Reference	Risk Rating
Short Hand IP Address	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Server-Side Request Forgery (SSRF) is a security vulnerability where an attacker manipulates an application to send unauthorized requests, potentially leading to data theft, server-side attacks, and access to internal systems. Web applications and APIs are particularly vulnerable, necessitating thorough input validation and access controls.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/ssrf_lab/lab_6/lab_6.php	
Consequences of not Fixing the Issue	
Data Breaches: Unresolved SSRF vulnerabilities can allow unauthorized access to internal systems' sensitive data, allowing attackers to steal confidential information like customer records, financial data, or intellectual property. **Financial Losses:** The theft of sensitive data can result in financial losses for the organization, including costs associated with legal fees, regulatory fines, and compensation for affected individuals. **Reputation Damage:** SSRF-related data breaches can severely harm an organization's reputation, potentially leading to customer trust loss, as customers may choose to relocate. **Network Compromise:** SSRF can serve as a pivot point for attackers, potentially compromising entire systems, disrupting services, and causing loss of critical infrastructure control. **Legal Ramifications:** Non-compliance with data protection regulations like GDPR and CCPA can lead to legal consequences for organizations failing to protect against SSRF attacks.	
Suggested Countermeasures	
Input Validation: Validate and sanitize user-supplied URLs to ensure they are legitimate and safe. **Whitelist Allowed Hosts:** Restrict the domains or IP addresses that the application can access, allowing only necessary and trusted resources. **Use of Redirection URLs:** Use a whitelist of allowed redirection URLs to prevent open redirects. **Firewall Configuration:** Configure firewalls to block requests to internal network resources from the web server.	
References	
https://portswigger.net/web-security/ssrf	

Proof of Concept

payload: `http://0/`



2.10. Look An SSRF On Cloud!

Reference	Risk Rating
Look An SSRF On Cloud!	**Low / Medium / High**
Tools Used	
Burp Suite	
Vulnerability Description	
Server-Side Request Forgery (SSRF) is a security vulnerability where an attacker manipulates an application to send unauthorized requests, potentially leading to data theft, server-side attacks, and access to internal systems. Web applications and APIs are particularly vulnerable, necessitating thorough input validation and access controls.	
How It Was Discovered	
Manual Analysis	
Vulnerable URLs	
https://labs.hacktify.in/HTML/ssrf_lab/lab_10/lab_10.php	
Consequences of not Fixing the Issue	
Data Breaches: Unresolved SSRF vulnerabilities can allow unauthorized access to internal systems' sensitive data, allowing attackers to steal confidential information like customer records, financial data, or intellectual property. **Financial Losses:** The theft of sensitive data can result in financial losses for the organization, including costs associated with legal fees, regulatory fines, and compensation for affected individuals. **Reputation Damage:** SSRF-related data breaches can severely harm an organization's reputation, potentially leading to customer trust loss, as customers may choose to relocate. **Network Compromise:** SSRF can serve as a pivot point for attackers, potentially compromising entire systems, disrupting services, and causing loss of critical infrastructure control. **Legal Ramifications:** Non-compliance with data protection regulations like GDPR and CCPA can lead to legal consequences for organizations failing to protect against SSRF attacks.	
Suggested Countermeasures	
Input Validation: Validate and sanitize user-supplied URLs to ensure they are legitimate and safe. **Whitelist Allowed Hosts:** Restrict the domains or IP addresses that the application can access, allowing only necessary and trusted resources. **Use of Redirection URLs:** Use a whitelist of allowed redirection URLs to prevent open redirects. **Firewall Configuration:** Configure firewalls to block requests to internal network resources from the web server.	
References	
https://portswigger.net/web-security/ssrf	

Proof of Concept

payload: http://metadata.google.internal/computeMetadata/v1/