Report Release Date	17th May 2025
Type of Audit	Web Application Audit
Type of Audit Report	First Audit Report
Period	14th May 2025– 17th May 2025

Document Preparation	
Document Title	Web Application Audit Report
Document ID	Qwegle Technologies
Document Version	1.0
Prepared by	Mr. Sanjeev Chavan
Reviewed by	Ms. Tasneam P
Approved by	Ms. Tasneam P
Released by	Ms. Tasneam P
Release date	17th May 2025

This document and the information contained within generated by Imperium Solutions, are intended for the use of Client personnel only. The information contained herein is confidential and is for use by the assigned personnel only. General circulation and reproduction in any form by unauthorized personnel or firm is prohibited.

Document Control

Document Distribution List			
Name	Organization	Designation	Email Id
Ms. Tasneam P	Imperium Solution	Head - Information Security and Data Privacy	tasneam@imperiumsolution.com

Document Change History		
Version	Date	Remarks / Reason of change
1.0	17th May 2025	No Change

Table of Contents

Company Profile

Imperium Solutions Profile

Imperium Solutions is a Cert-IN empanelled, ISO 9001 and ISO27001 certified organization, focused on Information Security and Data Privacy domains. Established in June 2008, we aim to partner with our clients to build and implement appropriate Information Security and Data Privacy Controls. Imperium Solutions professionals have rich domain expertise across a cross section of industry verticals and the multi-dimensional skills to meet the ever-changing business needs of our clients.

Imperium Solution’s unique blend of customer focus and consistency in delivering services of the highest quality provides a competitive edge to our clients.

Our portfolios of solutions help our clients achieve their business objectives while maintaining the security and privacy of their data.

Targeted Audience

Citing the identified risks and vulnerabilities as mentioned in the report, it should be considered as Strictly Confidential and accessible only to the Top Management, IT Manager, Software Development Team, Software Testing Team and senior staff members responsible for taking action.

Audit Objective

The objective of this vulnerability assessment and penetration testing exercise is to

1. Identify the vulnerabilities existing in the application module within the scope of the vulnerability assessment and penetration testing exercise.

2. Determine whether the vulnerabilities can be exploited and information gathered from the same.
3. Recommend solutions to plug these vulnerabilities.

Audit Scope

The following web application within the scope of the VAPT exercise.

Sr. No.	Web Service Link
1	http://44.209.151.89:5173/

S. No	Asset Description	Criticality of Asset	Internal IP Address	URL	Public IP Address	Location	Hash Value (in case of applications)	Version (in Case of applications)	Other details such as make and model in case of network devices or security devices.
1.	Web Application	Not Applicable	Not Available	http://44.209.151.89:5173/			Application SHA value		Not Applicable

Engagement Scope

Details of the Auditing team

S. No.	Name	Designation	Email Id	Professional Qualifications/ Certifications	Whether the resource has been listed in the Snapshot information published on CERT-In's website(Yes/No)
1	Mr. Sanjeev Chavan	Information Security Consultant	sanjeev.chavan@imperiumsolution.com	Masters in Computer Science	No

S. No.	Name	Designation	Email Id	Professional Qualifications/ Certifications	Whether the resource has been listed in the Snapshot information published on CERT-In’s website(Yes/No)
2	Ms. Tasneam P	Head - Information Security and Data Privacy	tasneam@imperiumsolution.com	CISA	Yes

Methodology

The vulnerability assessment and penetration testing methodology is as outlined below:

Information Gathering

1. Study & scope the IT architecture & components for assessment
2. Determine the boundary of analysis
3. Impact analysis for Active scans, which includes assessment of Services or Servers scans in the operations environment.
4. Plan for Downtime & Contingency, if applicable
5. Estimate the scan process, based on the complexity of the target networks, hosts and applications
6. Define the scan Policy for each target. Scan Policy to define the level of scan – Information gathering, Policy checking, Port scanning, Password analysis, Attack stimulation etc.
7. Scan the targeted networks, hosts and applications based on the defined scan policy
8. Collect the scan results and analyse the security loopholes, configuration errors, default installation settings, overlooked setups, password quality, firmware/software revisions, patch fixes, security policy violations etc.
9. Submission of Assessment Reports with suggestions and recommendations to fix the vulnerabilities.

Analysis

1. Analysis the scan results and details collected.

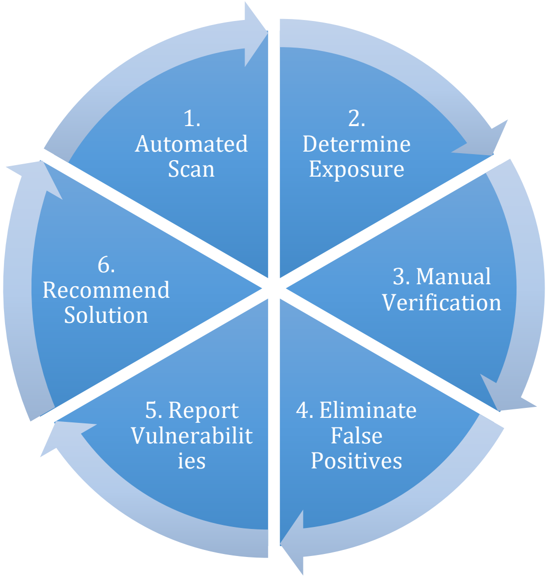
Report Preparation

The report will comprise of the following assessment components

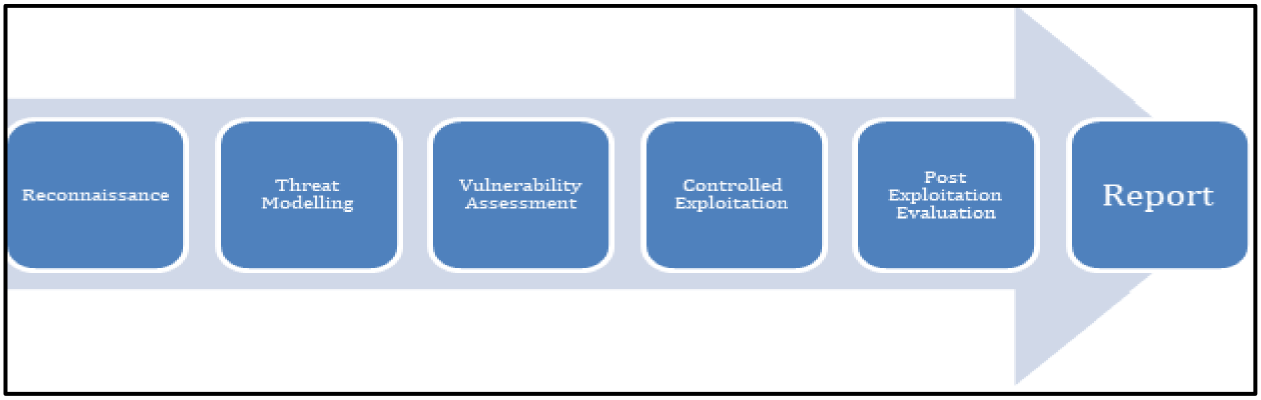
1. Vulnerability Assessment and Penetration testing Findings

2. Recommendations – Controls to be implemented to mitigate the identified vulnerabilities.

Vulnerability Assessment Methodology



Penetration Testing Methodology



Audit Conducted

Below are the details of the applications for which the Web audit was conducted.

Sr. No.	Web Service Link
1	http://44.209.151.89:5173/

Tools/ Software used

Sr. No.	Name of Tool/ Software used	Version of the tool/Software used	Open Source/Licensed
1.	BurpSuit Professional	V.2024.4	Licensed

Risk Summary

The vulnerability assessment and penetration testing exercise aims to highlight the existing vulnerabilities within the application module and the different ways in which it can be addressed and closed.

We strongly recommend expeditious removal of the inconsistencies brought out in this report.

Risks have been defined as critical, high, medium and low.

Critical risks: These risks have to be addressed on priority. The system is extremely vulnerable and needs prompt mitigation.

High risk: There is a strong need for corrective measures. The existing system may continue to operate but the corrective action needs to be implemented as soon as possible

Medium risk: Corrective actions are needed and a plan must be developed to incorporate these actions within a reasonable period of time.

Low risk: Client Shall determine whether corrective actions are still required or decide to accept the risk.

Likelihood	
Level	Definition
High	The threat's source is highly motivated and sufficiently capable, and controls that prevent the vulnerability from being exercised are ineffective.
Medium	The threat's source is motivated and capable, but controls are in place that may impede a successful exercise of the vulnerability.

Likelihood	
Level	Definition
Low	The threat's source lacks motivation or capability, and controls are in place to prevent or significantly impede the vulnerability from being exercised.

Impact and Degree	
Level	Definition
High	High impact risks may result in the high costly loss of assets; risks that significantly violate, harm, or impede operations; or risks that cause human death or serious injury.
Medium	Medium impact risks may result in the costly loss of assets; risks that violate, harm, or impede operations; or risks that cause human injury.
Low	Low impact risks may result in the loss of some assets or may noticeably affect operations.

Risk	
Mitigation	Recommendation
Critical	This is a critical vulnerability and needs to be addressed immediately
High	There is a strong need for corrective measures. The existing system may continue to operate but the corrective action needs to be put in place as early as possible
Medium	Corrective actions are needed and a plan must be developed to incorporate these actions within a reasonable period of time
Low	Company to determine corrective actions or decide to accept the risk

Management Summary

Our Information Security Assessment team would like to take this opportunity to thank all the participants for their assistance and co-operation in ensuring that the assessment could be completed smoothly and in a timely manner.

The web application module was in the scope of the VAPT activity. Critical and High vulnerabilities have been identified during the scan. The scan has been done using multiple tools followed by manual verification to reduce the risks of false positives. It is recommended that the Critical and High vulnerabilities be addressed immediately.

It is further recommended that the Application be scanned for vulnerabilities whenever there is a change in the configuration or at least once in a year, whichever is earlier. All changes to the application should be done through a proper change management process.

Below is the summarized report of the vulnerabilities.

Critical	High	Medium	Low	Information
0	03	10	11	00

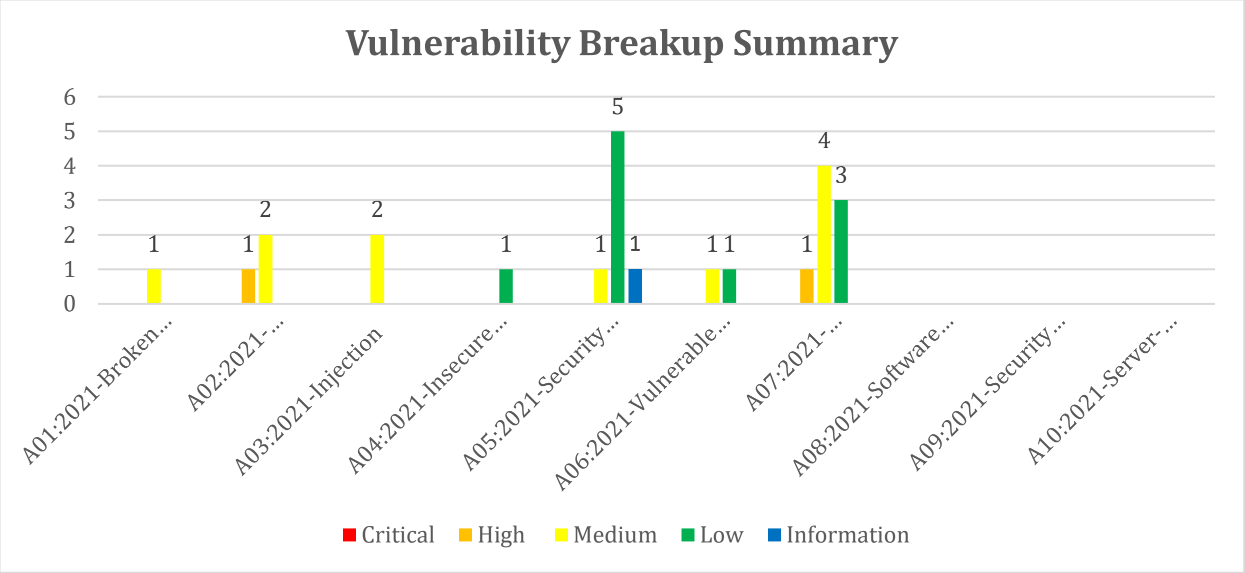


Figure 1: Vulnerability Distribution with severity

S. No	Affected Asset	Observation/ Vulnerability title	CVE/CWE	Control Objective	Control Name	Audit Requirement	Severity	Recommendation	Reference	New or Repeat observation
1	http://44.209.151.89:5173/	Brute force Attack	CWE-307	NA	NA	NA	High	Temporarily lock accounts after a number of failed login attempts.	https://owasp.org/www-community/attacks/Brute_force_attack	New Observation
2	http://44.209.151.89:5173/	Credentail transmitted to Server in Plain Text	CWE-319	NA	NA	NA	High	Use HTTPS for all web traffic, especially during login or	https://www.acunetix.com/vulnerabilities/web/password-transmi	New Observation

								API authentication.	tted-over-http/	
3	http://44.209.151.89:5173/	Vertical Previleage Escalration	CWE 434	NA	NA	NA	High	Assign roles and permissions clearly (e.g., user, moderator, admin).	https://www.acunetix.com/vulnerabilities/web/tag/privilege-escalation/	New Observation
4	http://44.209.151.89:5173/	Broken Authentication Via Response Mainupulation	CWE-345	NA	NA	NA	Medium	Never expose authentication status flags in client-controllable data (like	https://owasp.org/Top10/A07_2021-Identification_and_Authentica	New Observation

								JSON, local storage, cookies without HttpOnly).	tion_Failures/	
5	http://44.209.151.89:5173/	Captcha not Implement	CWE-799	NA	NA	NA	Medium	For better user experience, use invisible or adaptive CAPTCHA based on behavior.	https://www.acunetix.com/vulnerabilities/web/wordpress-plugin-login-no-captcha-recaptcha-security-	New Observation

									bypass-1-4-1/	
6	http://44.209.151.89:5173/	Configuration file Accessible	CWE-538	NA	NA	NA	Medium	Ensure sensitive files (e.g., .env, config.php) are not directly accessible via HTTP.	https://www.acunetix.com/vulnerabilities/web/configuration-file-disclosure/	New Observation
7	http://44.209.151.89:5173/	Sesnsitive Data not masked	CWE-359	NA	NA	NA	Medium	Filter out or redact sensitive fields before writing to logs (e.g.,	https://cwe.mitre.org/data/definitions/200.html	New Observation

								passwords, tokens, PII).		
8	http://44.209.151.89:5173/	Session Token in Url	CWE-598	NA	NA	NA	Medium	Store session tokens in secure, HttpOnly, SameSite cookies instead of URLs.	https://www.acunetix.com/vulnerabilities/web/session-id-in-url/	New Observation
9	http://44.209.151.89:5173/	Json Syntax Error	CWE-116	NA	NA	NA	Medium	Use strict schema validation (e.g., using AJV in Node.js or	https://www.invicti.com/learn/json-injection/	New Observation

								jsonschema in Python).		
10	http://44.209.151.89:5173/	Improper Error Handling	CWE-209	NA	NA	NA	Medium	Ensure that error messages shown to users do not contain sensitive information. Use generic messages like "Something went wrong.	https://owasp.org/www-community/Improper_Error_Handling	New Observation

								Please try again later" rather than exposing technical details.		
11	http://44.209.151.89:5173/	Improper Input Validation	CWE-20	NA	NA	NA	Medium	Whitelisting should be used wherever possible. Only allow known, safe inputs (e.g.,	https://cwe.mitre.org/data/definitions/20.html	New Observation

								specific characters, length, and data types).		
12	http://44.209.151.89:5173/	Weak Password Policy	CWE-521	NA	NA	NA	Medium	Prevent users from reusing their last N passwords (e.g., last 5 or 10 passwords).	https://www.acunetix.com/vulnerabilities/web/weak-password/	New Observation
13	http://44.209.151.89:5173/	Unencrypted Communication	CWE-319	NA	NA	NA	Medium	Ensure that all web traffic is encrypt	https://www.acunetix.com/vulnerabili	New Observation

								ed with TLS (Transport Layer Security), previously known as SSL.	ties/web/ssl-tls-not-implemented/	
14	http://44.209.151.89:5173/	Missing Security Headers	CWE 693	NA	NA	NA	Low	Implement Security Headers, Add recommended security headers to your web server configu	https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Headers_Cheat_Sheet.html	New Observation

								ration or application response.		
15	http://44.209.151.89:5173/	Concurrent Login	CWE-307	NA	NA	NA	Low	Implement a policy to allow only a single active session per user at a time (or a limited number of sessions). This can help	https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/06-Session_Manag	New Observation

								prevent multiple logins and control access to the user account more securely.	ement Testing/11-Testing for Concurrent Sessions	
16	http://44.209.151.89:5173/	Lack of Phone number Verification	CWE-284	NA	NA	NA	Low	When users enter a phone number, send a one-time passcode (OTP) to that number. The	https://cwe.mitre.org/data/definitions/345.html	New Observation

								user must then enter the correct OTP to verify ownership of the phone number.		
17	http://44.209.151.89:5173/	clickjacking Attack	CWE-1021	NA	NA	NA	Low	The X-Frame-Options HTTP header prevents your content from being embed	https://owasp.org/www-community/attacks/Clickjacking	New Observation

								ded into an iframe. This can block clickjacking attacks by disallowing your page from being embedded on malicious sites.		
18	http://44.209.151.89:5173/	Exposed Directory File	CWE-548	NA	NA	NA	Low	Disable directory listing on the web server	https://www.acunetix.com/vulnerabilities/we	New Observation

							🟩	(e.g., Apache, Nginx). This prevents attackers from viewing the contents of directories that do not have an index file (e.g., index.html, index.php).	b/directory-listings/	

19	http://44.209.151.89:5173/	Outdated Component Used	CWE-1104	NA	NA	NA	Low	Ensure that all components (e.g., libraries, frameworks, APIs) are updated to the latest stable version that has received security patches	https://owasp.org/Top10/A06_2021-Vulnerable_and_Outdated_Components/	New Observation
20	http://44.209.151.89:5173/	Functionality Not Working	CWE-802	NA	NA	NA	Low	Conduct unit testing, integrat	https://www.invicti.com/web-	New Observation

							🟩	ion testing, and user acceptance testing (UAT) to ensure that all features are functioning as expected in different environments and scenarios.	vulnerability-scanner/vulnerabilities/classification/cwe-22/	

21	http://44.209.151.89:5173/	Allow origin Allow origin Misconfiguration	CWE-345	NA	NA	NA	Low	If the origin must be dynamic (i.e., vary based on the request), implement logic to check the Origin header and ensure it matches a whitelis	https://www.acunetix.com/vulnerabilities/web/misconfigured-access-control-allow-origin-header/	New Observation

							🟩	t of trusted origins.		
22	http://44.209.151.89:5173/	Old Password Set as New Password	CWE-759	NA	NA	NA	Low	When users attempt to change their password, ensure that the new password is different from the old passwo	https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/04-	New Observation

								rd. This is a simple yet crucial validation check.	Authentication_Testing/09-Testing_for_Weak_Password_Change_or_Reset_Functionalities	
23	http://44.209.151.89:5173/	Unwanted http Method Enable	CWE-444	NA	NA	NA	Low	Review the HTTP methods allowed by your web server and application, and	https://owasp.org/www-project-web-security-testing-guide/v41/4-Web_A	New Observation

								disable any that are unnecessary for the functioning of the application.	pplication_Security_Testing/02-Configuration_and_Deployment_Management_Testing/06-Test_HTTP_Methods	
24	http://44.209.151.89:5173/	Open port	CWE-200	NA	NA	NA	Low	Identify and close any open ports that are not needed	https://mas.owasp.org/MASWE/MASVS-NETWORK/MA	New Observation

							🟩	for legitimate purposes. This includes ports related to unused services, outdated protocols, or any service that does not need to be exposed to the	SWE-0051/	

							🟩	internet.		

Detailed Report

Vulnerability Details

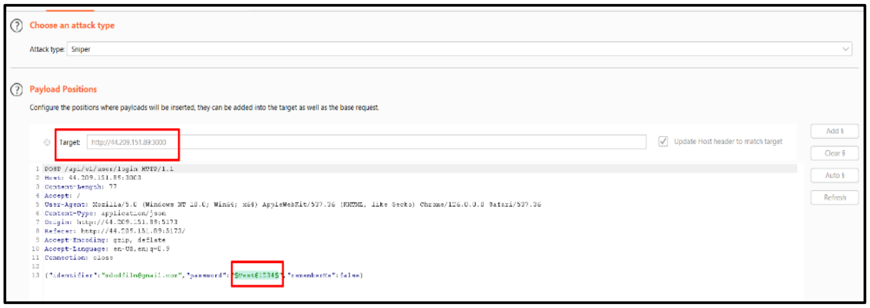
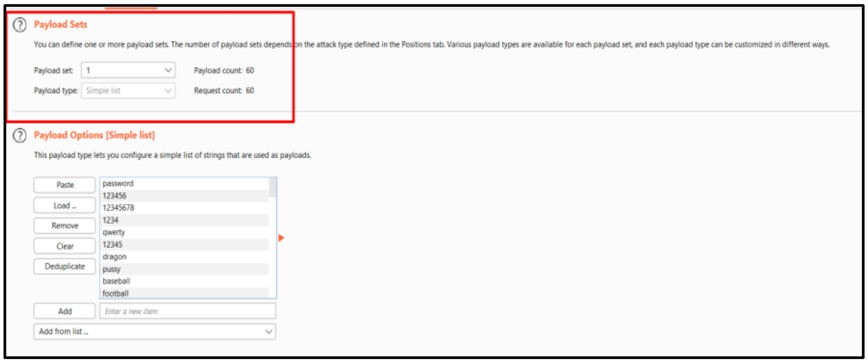
Vulnerability Severity Level:

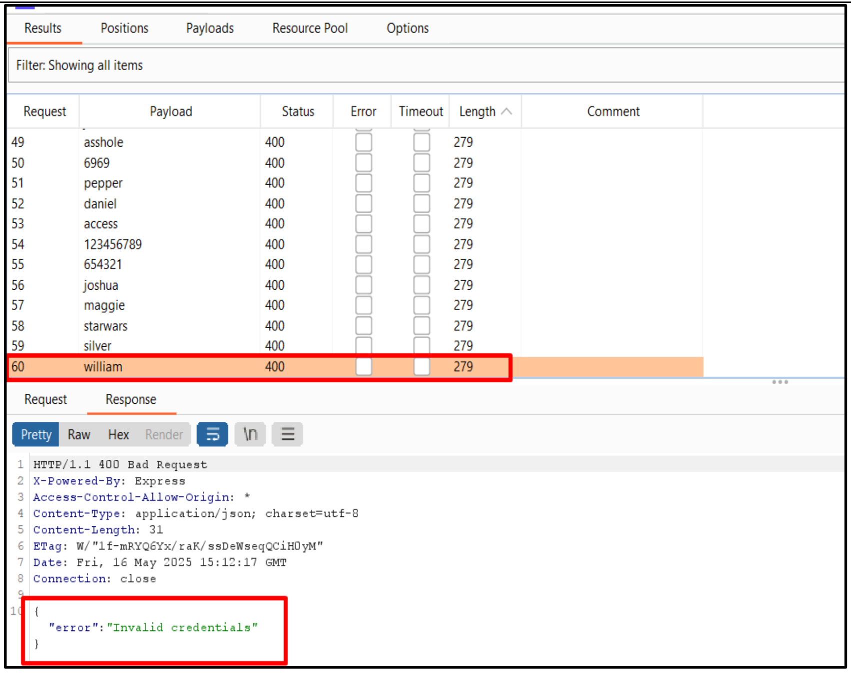
Critical	🟥
High	🟧
Medium	🟨
Low	🟩
Information	🟦

01 : Brute Force Attack

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	A Brute Force Attack occurs when an attacker systematically attempts all possible combinations of usernames and passwords until the correct one is found. This typically targets login forms, APIs, or password-protected resources that do not have adequate protection mechanisms in place.
Detailed observation / Vulnerable point	Unauthorized access to sensitive user accounts or administrative interfaces Credential stuffing when credentials from other breaches are reused
CVE/CWE	CWE-307
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	High

Recommendation	Temporarily lock accounts after a number of failed login attempts.Use exponential backoff or CAPTCHA after repeated failures
Reference	https://owasp.org/www-community/attacks/Brute_force_attack
New or Repeat observation	New Observation

References to evidences / Proof of Concept	
	Choose an attack type
	Attack type: Sniper
	Payload Positions
	Configure the positions where payloads will be inserted, they can be added into the target as well as the base request.
	Target: http://44.209.151.89:3000
	☑ Update Host header to match target
	`POST /api/v1/user/login HTTP/1.1`
	`Host: 44.209.151.89:3000`
	`Content-Length: 77`
	`Accept: /`
	`User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36`
	`Content-Type: application/json`
	`Origin: http://44.209.151.89:5173`
	`Referer: http://44.209.151.89:5173/`
	`Accept-Encoding: gzip, deflate`
	`Accept-Language: en-US,en;q=0.9`
	`Connection: close`
	
	`{"identifier":"mdodfiln@gmail.com","password":"§Test@12345§","rememberMe":false}`
	
	
	Payload Sets
	You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.
	Payload set: 1
	Payload count: 60
	Payload type: Simple list
	Request count: 60
	Payload Options [Simple list]
	This payload type lets you configure a simple list of strings that are used as payloads.
	password
	123456
	12345678
	1234
	qwerty
	12345
	dragon
	pussy
	baseball
	football
	 Paste
	Load ...
	Remove
	Clear
	Deduplicate
	Add
	Enter a new item
	Add from list ...

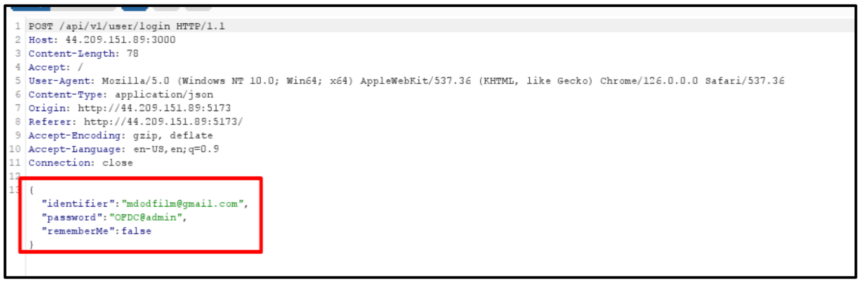


Request	Payload	Status	Error	Timeout	Length	Comment
49	asshole	400	☐	☐	279	
50	6969	400	☐	☐	279	
51	pepper	400	☐	☐	279	
52	daniel	400	☐	☐	279	
53	access	400	☐	☐	279	
54	123456789	400	☐	☐	279	
55	654321	400	☐	☐	279	
56	joshua	400	☐	☐	279	
57	maggie	400	☐	☐	279	
58	starwars	400	☐	☐	279	
59	silver	400	☐	☐	279	
60	william	400	☐	☐	279	

```
HTTP/1.1 400 Bad Request
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 31
ETag: W/"1f-mRYQ6Yx/raK/ssDeWseqQCiH0yM"
Date: Fri, 16 May 2025 15:12:17 GMT
Connection: close

{
  "error":"Invalid credentials"
}
```

02: Credential Transmitted to Server in Plain Text

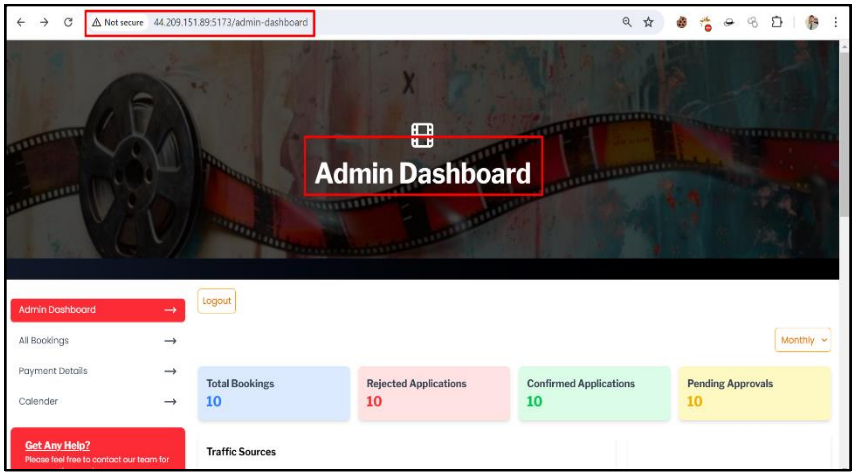
Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	This vulnerability occurs when sensitive data — such as usernames, passwords, API keys, or session tokens — is transmitted over the network without encryption, such as via HTTP instead of HTTPS, or via other insecure protocols like FTP, SMTP, or Telnet.
Detailed observation / Vulnerable point	Credential interception by attackers via sniffing (Man-in-the-Middle attacks)
CVE/CWE	CWE-319
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	High
Recommendation	Use HTTPS for all web traffic, especially during login or API authentication.

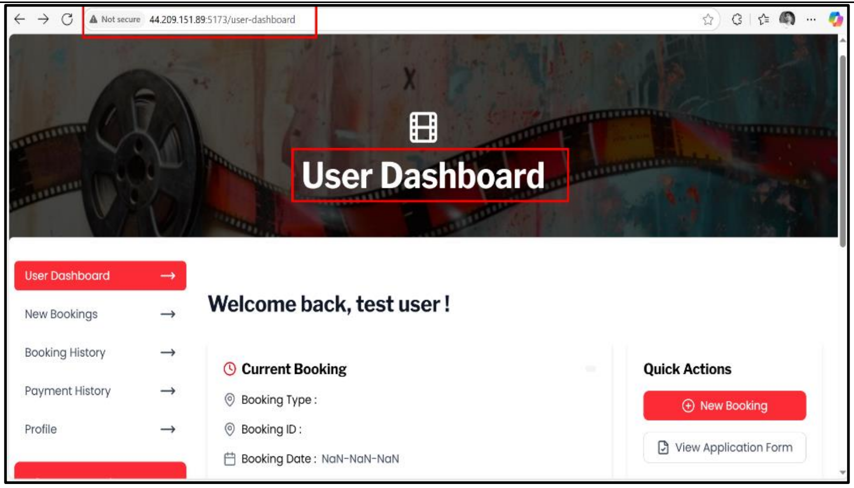
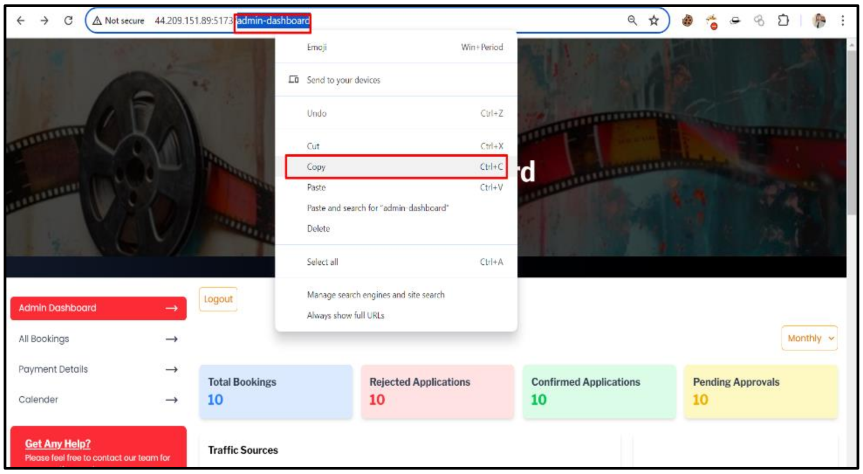
Reference	https://www.acunetix.com/vulnerabilities/web/password-transmitted-over-http/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

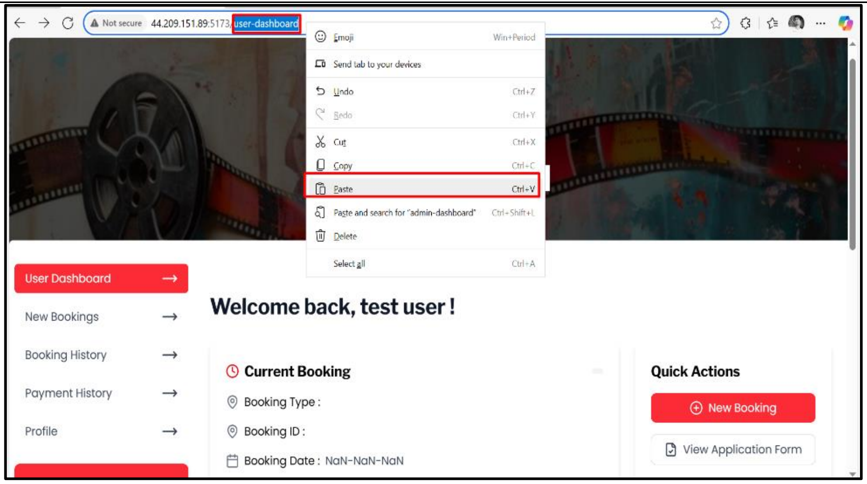
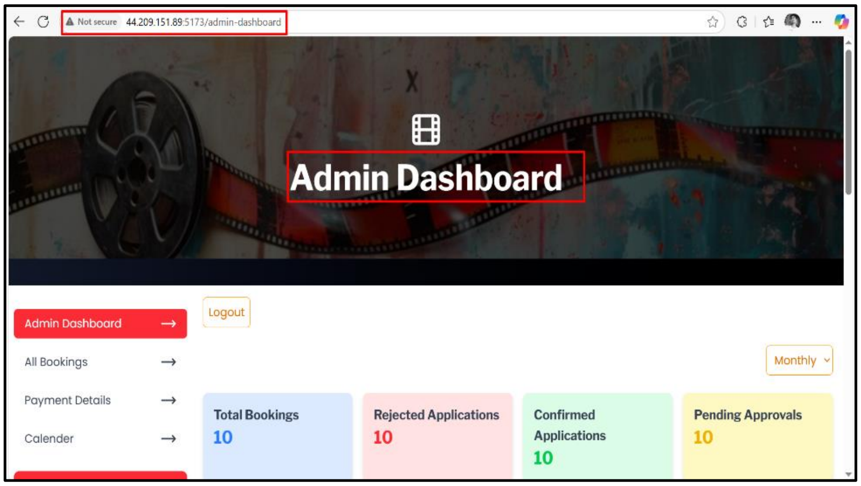
```
POST /api/v1/user/login HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 78
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "identifier":"mdodfilm@gmail.com",
  "password":"OFDC@admin",
  "rememberMe":false
}
```

03: Vertical Privilege Escalation

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Vertical privilege escalation occurs when a user gains access to higher-level privileges or functions than they are authorized for. For example, a regular user accessing an admin dashboard, or an unprivileged user performing actions that only a system administrator should be able to perform.
Detailed observation / Vulnerable point	Unauthorized access to administrative features. Data manipulation, deletion, or exfiltration
CVE/CWE	CWE-269
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	High

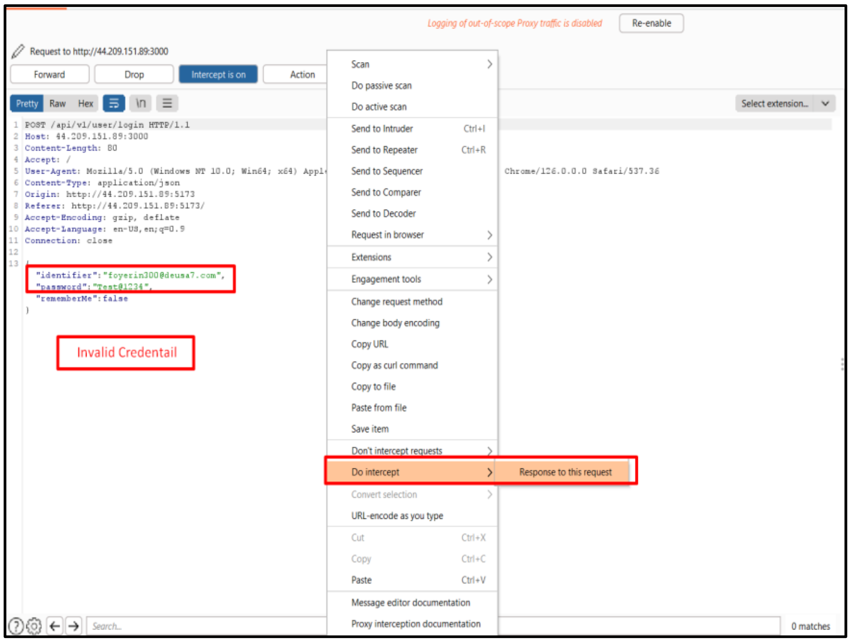
Recommendation	Assign roles and permissions clearly (e.g., user, moderator, admin). Check permissions on the server-side for every sensitive action.
Reference	https://www.acunetix.com/vulnerabilities/web/tag/privilege-escalation/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Not secure 44.209.151.89:5173/admin-dashboard Admin Dashboard Logout Admin Dashboard All Bookings Payment Details Calender Monthly Total Bookings 10 Rejected Applications 10 Confirmed Applications 10 Pending Approvals 10 Get Any Help? Please feel free to contact our team for Traffic Sources

	 Not secure 44.209.151.89:5173/user-dashboard User Dashboard User Dashboard New Bookings Booking History Payment History Profile Welcome back, test user ! Current Booking Booking Type : Booking ID : Booking Date : NaN-NaN-NaN Quick Actions New Booking View Application Form  Not secure 44.209.151.89:5173/admin-dashboard Emoji Win+Period Send to your devices Undo Ctrl+Z Cut Ctrl+X Copy Ctrl+C Paste Ctrl+V Paste and search for "admin dashboard" Delete Select all Ctrl+A Manage search engines and site search Always show full URLs Admin Dashboard All Bookings Payment Details Calender Logout Monthly Total Bookings 10 Rejected Applications 10 Confirmed Applications 10 Pending Approvals 10 Traffic Sources Get Any Help? Please feel free to contact our team for

	 Not secure 44.209.151.89:5173/user-dashboard Emoji Win+Period Send tab to your devices Undo Ctrl+Z Redo Ctrl+Y Cut Ctrl+X Copy Ctrl+C Paste Ctrl+V Paste and search for "admin-dashboard" Ctrl+Shift+L Delete Select all Ctrl+A User Dashboard New Bookings Booking History Payment History Profile Welcome back, test user ! Current Booking Booking Type : Booking ID : Booking Date : NaN-NaN-NaN Quick Actions New Booking View Application Form  Not secure 44.209.151.89:5173/admin-dashboard Admin Dashboard Admin Dashboard All Bookings Payment Details Calender Logout Monthly Total Bookings 10 Rejected Applications 10 Confirmed Applications 10 Pending Approvals 10

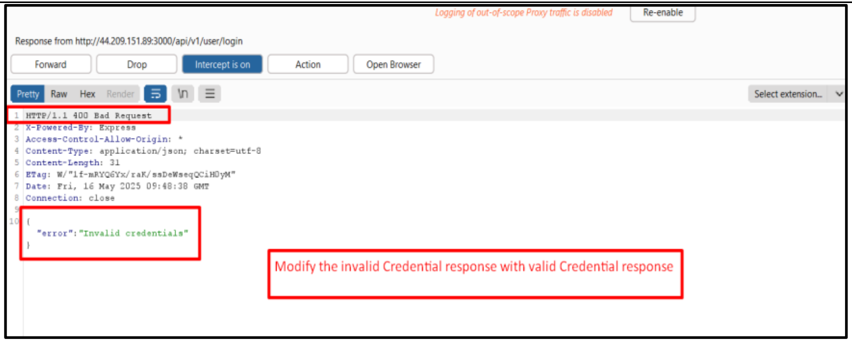
04: Broken Authentication Via Response Mainupulation

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Broken Authentication via Response Manipulation occurs when a web application incorrectly trusts user-controllable client-side data (such as HTTP responses, status codes, or headers) to confirm authentication or login success.
Detailed observation / Vulnerable point	Bypass of authentication Account takeover Privilege escalation Full compromise of application security trust model
CVE/CWE	CWE-345
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium
Recommendation	Do not rely on client-side logic to determine whether a user is logged in. Validate tokens/sessions on the server for each sensitive request.

Reference	https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

```
POST /api/v1/user/login HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 80
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) Appl      Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "identifier":"foyerin300@deusa7.com",
  "password":"Test@1234",
  "rememberMe":false
}
```

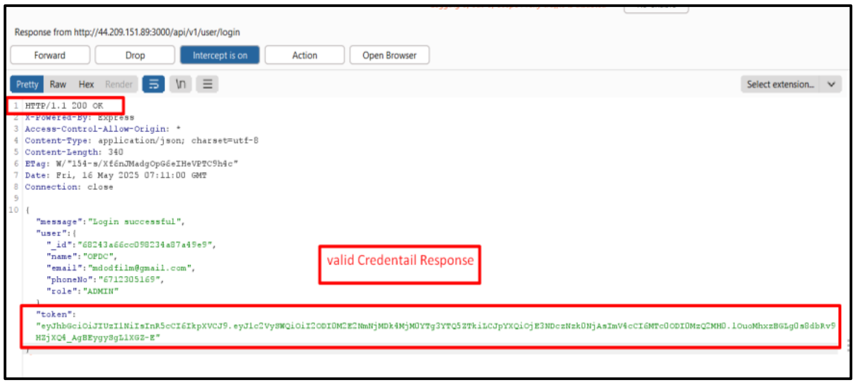
Invalid Credentail



Response from http://44.209.151.89:3000/api/v1/user/login

```
HTTP/1.1 400 Bad Request
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 31
ETag: W/"1f-mRYQ6Yx/raK/ssDeWseqQCiHDyM"
Date: Fri, 16 May 2025 09:48:38 GMT
Connection: close

{
  "error":"Invalid credentials"
}
```

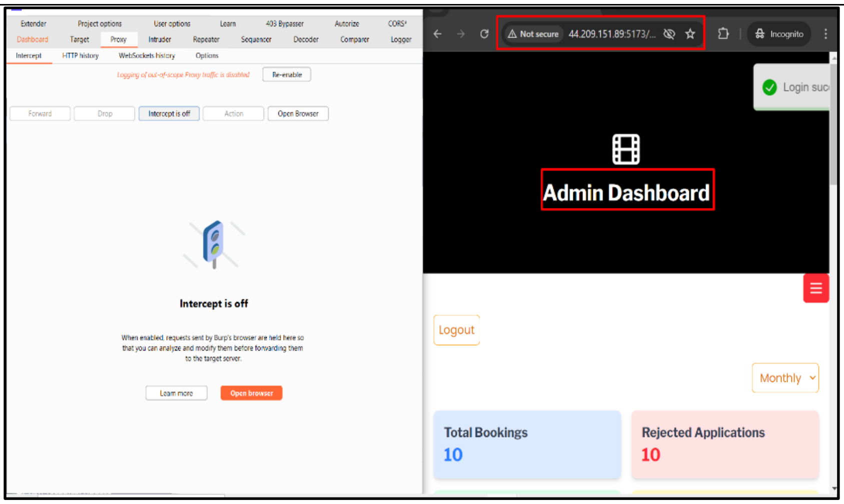
Modify the invalid Credential response with valid Credential response



Response from http://44.209.151.89:3000/api/v1/user/login

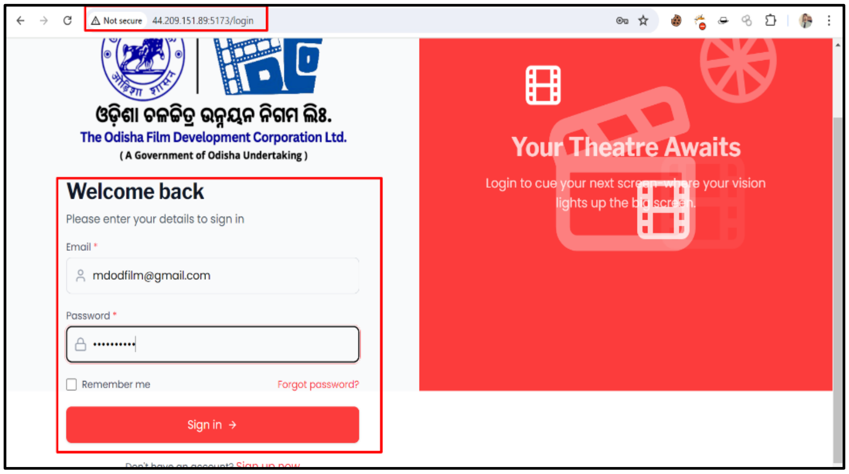
```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 340
ETag: W/"154-s/Xf6nJMadgOpG6eIHeVPTC9h4c"
Date: Fri, 16 May 2025 07:11:00 GMT
Connection: close

{
  "message":"Login successful",
  "user":{
    "_id":"68243a66cc098234a87a49e9",
    "name":"OPDC",
    "email":"mdodfilm@gmail.com",
    "phoneNo":"6712305169",
    "role":"ADMIN"
  }
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOiI2ODI0M2E2NmNjMDk4MjM0YTg3YTQ5ZTkiLCJpYXQiOjE3NDczNzk0NjAsImV4cCI6MTc0ODI0MzQ2MH0.lOuoMhxzBGLg0s8dbRv9
  HZjXQ4_AgBEygySgL1XGZ-E"
}
```

valid Credentail Response

	 Logging of out-of-scope Proxy traffic is disabled Intercept is off When enabled, requests sent by Burp's browser are held here so that you can analyze and modify them before forwarding them to the target server. Not secure 44.209.151.89:5173/... Login suc Admin Dashboard Total Bookings 10 Rejected Applications 10 Extender Project options User options Learn 403 Bypasser Autorize CORS* Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Logger Intercept HTTP history WebSockets history Options Re-enable Forward Drop Intercept is off Action Open Browser Learn more Open browser Incognito Logout Monthly

05: Captcha not Implement

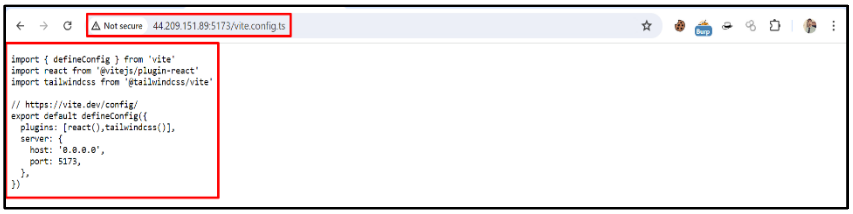
Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Failing to implement a CAPTCHA (Completely Automated Public Turing test to tell Computers and Humans Apart) on critical user input forms (e.g., login, registration, password reset) can leave the application vulnerable to automated abuse such as Brute-force login attempts Credential stuffing attacks
Detailed observation / Vulnerable point	Account takeover through automated brute-force or credential stuffing Denial of service or abuse of application features
CVE/CWE	CWE-799
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium

Recommendation	For better user experience, use invisible or adaptive CAPTCHA based on behavior. Complement CAPTCHA with IP-based rate limiting and lockouts.
Reference	https://www.acunetix.com/vulnerabilities/web/wordpress-plugin-login-no-captcha-recaptcha-security-bypass-1-4-1/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Not secure 44.209.151.89:5173/login ଓଡ଼ିଶା ଚଳଚ୍ଚିତ୍ର ଉନ୍ନୟନ ନିଗମ ଲିଃ. The Odisha Film Development Corporation Ltd. (A Government of Odisha Undertaking) Welcome back Please enter your details to sign in Email * mdodfilm@gmail.com Password * •••••••••• □ Remember me Forgot password? Sign in → Don't have an account? Sign up now Your Theatre Awaits Login to cue your next screen—where your vision lights up the big screen.

06: Configuration file Accessible

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	When sensitive configuration files (such as .env, config.php, web.config, application.yml, or .git/config) are publicly accessible over the web, attackers can retrieve them and gain access to Database credentials API keys and secrets
Detailed observation / Vulnerable point	Credential leakage (database, cloud, third-party APIs) Unauthorized access to systems
CVE/CWE	CWE-538
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium

<table>
<tr><td>Recommendation</td><td>Ensure sensitive files (e.g., .env, config.php) are not directly accessible via HTTP. Restrict read access to configuration files to only necessary system users.</td></tr>
<tr><td>Reference</td><td>https://www.acunetix.com/vulnerabilities/web/configuration-file-disclosure/</td></tr>
<tr><td>New or Repeat observation</td><td>New Observation</td></tr>
<tr><td>References to evidences / Proof of Concept</td><td>

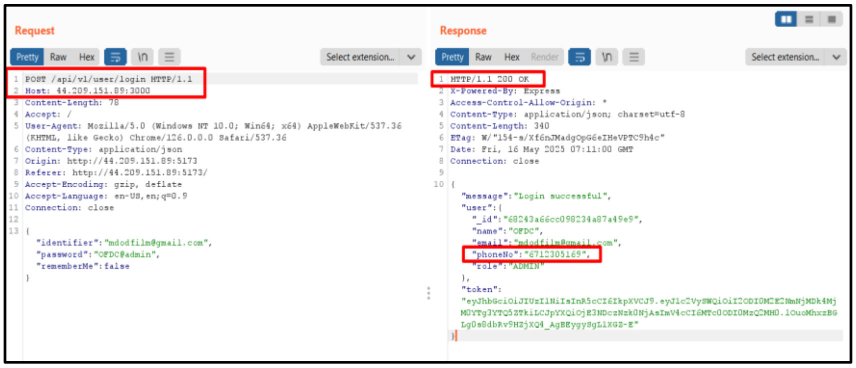


```
import { defineConfig } from 'vite'
import react from '@vitejs/plugin-react'
import tailwindcss from '@tailwindcss/vite'

// https://vite.dev/config/
export default defineConfig({
  plugins: [react(),tailwindcss()],
  server: {
    host: '0.0.0.0',
    port: 5173,
  },
})
```

</td></tr>
</table>

07: Sesnsitive Data not masked

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Sensitive data (e.g., credit card numbers, Social Security Numbers (SSNs), email addresses, authentication tokens, or medical records) should be masked when displayed in user interfaces, logs, debug tools, or transmitted in non-secure contexts.
Detailed observation / Vulnerable point	Privacy violations (especially under regulations like GDPR, HIPAA, PCI-DSS) Identity theft or financial fraud Reputation damage
CVE/CWE	CWE-359
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium

Recommendation	Filter out or redact sensitive fields before writing to logs (e.g., passwords, tokens, PII). Only authorized roles should see unmasked data.
Reference	https://cwe.mitre.org/data/definitions/200.html
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

Request

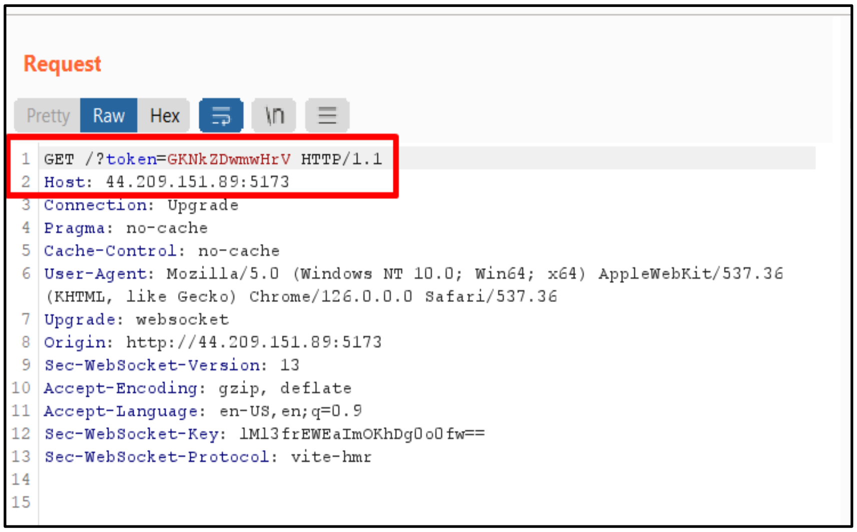
```
POST /api/v1/user/login HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 78
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "identifier":"mdodfilm@gmail.com",
  "password":"OFDC@admin",
  "rememberMe":false
}
```

Response

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 340
ETag: W/"154-s/Xf6nJMadgOpG6eIHeVPTC9h4c"
Date: Fri, 16 May 2025 07:11:00 GMT
Connection: close

{
  "message":"Login successful",
  "user":{
    "_id":"68243a66cc098234a87a49e9",
    "name":"OFDC",
    "email":"mdodfilm@gmail.com",
    "phoneNo":"6712305169",
    "role":"ADMIN"
  },
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2ODI0M2E2NmNjMDk4MjM0YTg3YTQ5ZTkiLCJpYXQiOjE3NDczNzk0NjAsImV4cCI6MTc0ODI0MzQ2MH0.lOuoMhxzBGLg0s8dbRv9HZjXQ4_AgBEygySgL1XGZ-E"
}
```

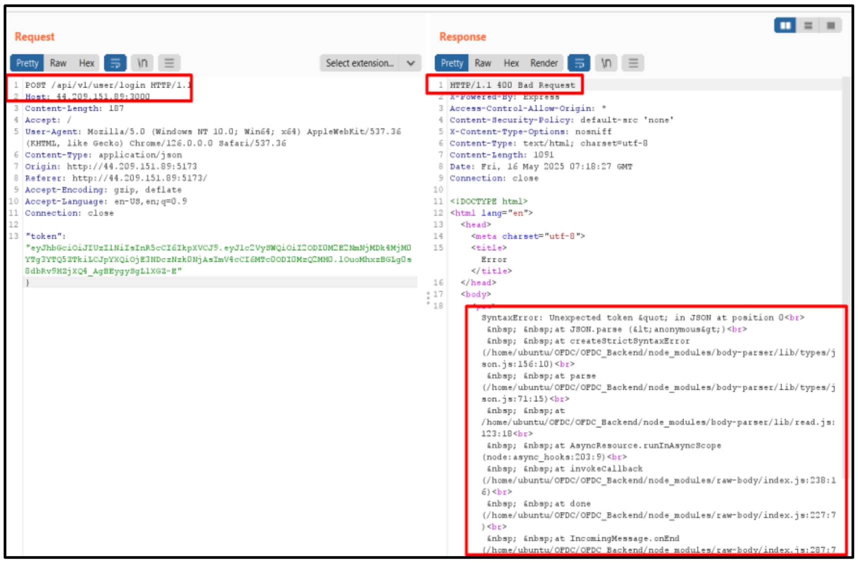
08: Session Token in Url

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Session tokens should never be included in the URL (query string) of a request. When session identifiers are passed via URLs, they may be. Stored in browser history Shared via Referer headers to third-party sites
Detailed observation / Vulnerable point	Session hijacking via leaked URLs Unauthorized access to user accounts or administrative interfaces
CVE/CWE	CWE-598
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium
Recommendation	Store session tokens in secure, HttpOnly, SameSite cookies instead of URLs. Use POST requests for login and session handling.

Reference	https://www.acunetix.com/vulnerabilities/web/session-id-in-url/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Request GET /?token=GKNkZDwmwHrV HTTP/1.1 Host: 44.209.151.89:5173 Connection: Upgrade Pragma: no-cache Cache-Control: no-cache User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36 Upgrade: websocket Origin: http://44.209.151.89:5173 Sec-WebSocket-Version: 13 Accept-Encoding: gzip, deflate Accept-Language: en-US,en;q=0.9 Sec-WebSocket-Key: lMl3frEWEaImOKhDg0o0fw== Sec-WebSocket-Protocol: vite-hmr

09: Json Syntax Error

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	A JSON Syntax Error occurs when the JSON data structure is malformed or not properly formatted, making it unreadable or improperly parsed by the server or client. Common causes include Missing commas or braces Improper escaping of quotes
Detailed observation / Vulnerable point	Service disruption or application errors Data parsing failures
CVE/CWE	CWE 693
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium
Recommendation	Use strict schema validation (e.g., using AJV in Node.js or jsonschema in Python). Reject malformed JSON before processing.

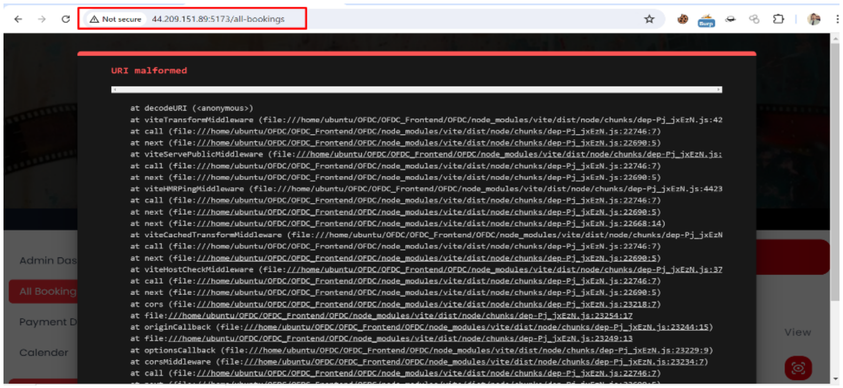
Reference	https://www.invicti.com/learn/json-injection/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

```
POST /api/v1/user/login HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 187
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

"token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOiI2ODI0M2E2NmNjMDk4MjMD
YTg3YTQ5ZTkiLCJpYXQiOjE3NDczNzk0NjAsImV4cCI6MTc0ODI0MzQ2MH0.lOuoMhxzBGLg0s
8dbRv9HZjXQ4_AgBEygySgLlXGZ-E"
}
```

```
HTTP/1.1 400 Bad Request
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Security-Policy: default-src 'none'
X-Content-Type-Options: nosniff
Content-Type: text/html; charset=utf-8
Content-Length: 1091
Date: Fri, 16 May 2025 07:18:27 GMT
Connection: close

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>
      Error
    </title>
  </head>
  <body>
    <pre>
      SyntaxError: Unexpected token &quot; in JSON at position 0<br>
       &nbsp; &nbsp;at JSON.parse (&lt;anonymous&gt;)<br>
       &nbsp; &nbsp;at createStrictSyntaxError
      (/home/ubuntu/OFDC/OFDC_Backend/node_modules/body-parser/lib/types/j
      son.js:156:10)<br>
       &nbsp; &nbsp;at parse
      (/home/ubuntu/OFDC/OFDC_Backend/node_modules/body-parser/lib/types/j
      son.js:71:15)<br>
       &nbsp; &nbsp;at
      /home/ubuntu/OFDC/OFDC_Backend/node_modules/body-parser/lib/read.js:
      123:18<br>
       &nbsp; &nbsp;at AsyncResource.runInAsyncScope
      (node:async_hooks:203:9)<br>
       &nbsp; &nbsp;at invokeCallback
      (/home/ubuntu/OFDC/OFDC_Backend/node_modules/raw-body/index.js:238:1
      6)<br>
       &nbsp; &nbsp;at done
      (/home/ubuntu/OFDC/OFDC_Backend/node_modules/raw-body/index.js:227:7
      )<br>
       &nbsp; &nbsp;at IncomingMessage.onEnd
      (/home/ubuntu/OFDC/OFDC_Backend/node_modules/raw-body/index.js:287:7
```

10: Improper Error Handling

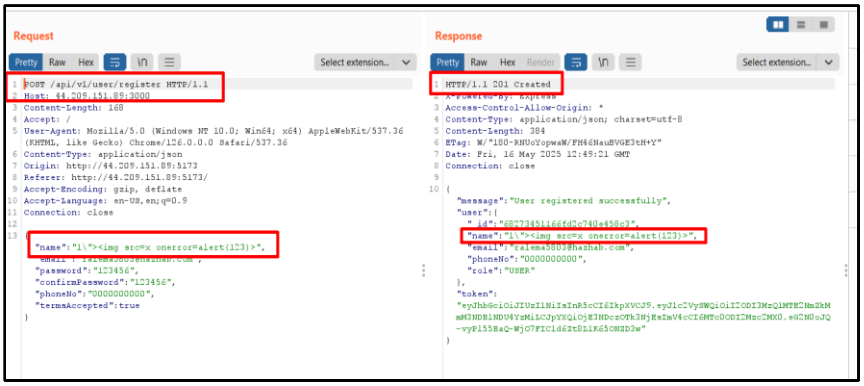
Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Improper error handling occurs when an application fails to properly manage or present error messages, potentially exposing sensitive information (e.g., stack traces, internal database queries, paths, server configurations) or allowing attackers to infer application logic. It can lead to Detailed error messages (e.g., SQL errors, stack traces) may reveal sensitive data such as database schema, application logic, or file paths.
Detailed observation / Vulnerable point	Exposure of sensitive information, like database credentials, file paths, or stack traces.
CVE/CWE	CWE-390
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium

Recommendation	Ensure that error messages shown to users do not contain sensitive information. Use generic messages like "Something went wrong. Please try again later" rather than exposing technical details.
Reference	https://owasp.org/www-community/Improper_Error_Handling
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Not secure 44.209.151.89:5173/all-bookings URI malformed at decodeURI (<anonymous>) at viteTransformMiddleware (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:42 at call (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22746:7) at next (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22690:5) at viteServePublicMiddleware (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js: at call (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22746:7) at next (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22690:5) at viteHMRPingMiddleware (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:4423 at call (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22746:7) at next (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22690:5) at next (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22668:14) at viteCachedTransformMiddleware (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN at call (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22746:7) at next (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22690:5) at viteHostCheckMiddleware (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:37 at call (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22746:7) at next (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22690:5) at cors (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:23218:7) at file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:23254:17 at originCallback (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:23244:15) at file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:23249:13 at optionsCallback (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:23229:9) at corsMiddleware (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:23234:7) at call (file:///home/ubuntu/OFDC/OFDC_Frontend/OFDC/node_modules/vite/dist/node/chunks/dep-Pj_jxEzN.js:22746:7)

This document and the information contained within generated by Imperium Solutions, are intended for the use of Client personnel only. The information contained herein is confidential and is for use by the assigned personnel only. General circulation and reproduction in any form by unauthorized personnel or firm is prohibited.

11: Improper Input Validation

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Improper Input Validation occurs when an application fails to validate input from users or external sources properly. It allows attackers to inject malicious or unexpected data that could compromise the security, integrity, or stability of the application.
Detailed observation / Vulnerable point	SQL Injection, Command Injection, Cross-Site Scripting (XSS), and other injection-based attacks. Buffer overflows leading to arbitrary code execution.
CVE/CWE	CWE-20
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium
Recommendation	Whitelisting should be used wherever possible. Only allow known, safe inputs (e.g., specific characters, length, and data types). Blacklist is

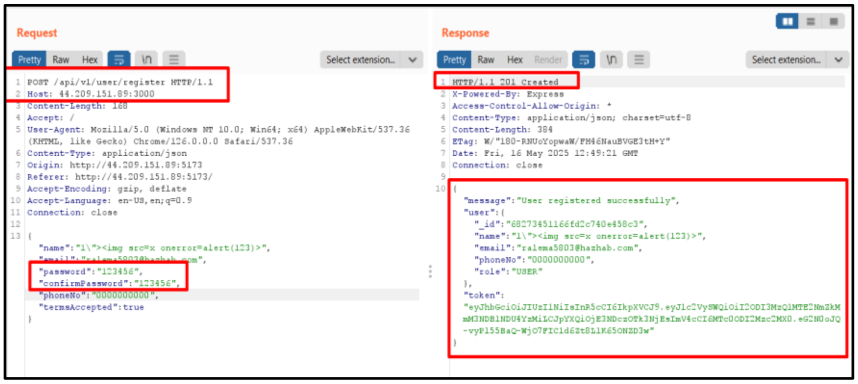
	generally a poor approach since attackers can craft payloads that bypass blacklist rules.
Reference	https://cwe.mitre.org/data/definitions/20.html
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

```
POST /api/v1/user/register HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 168
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "name":"1\"><img src=x onerror=alert(123)>",
  "email":"ralema5803@hazhab.com",
  "password":"123456",
  "confirmPassword":"123456",
  "phoneNo":"0000000000",
  "termsAccepted":true
}
```

```
HTTP/1.1 201 Created
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 384
ETag: W/"180-RNUoYopwaW/FH46NauBVGE3tH+Y"
Date: Fri, 16 May 2025 12:49:21 GMT
Connection: close

{
  "message":"User registered successfully",
  "user":{
    "_id":"68273451166fd2c740e458c3",
    "name":"1\"><img src=x onerror=alert(123)>",
    "email":"ralema5803@hazhab.com",
    "phoneNo":"0000000000",
    "role":"USER"
  },
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2ODI3MzQ1MTE2NmZkM
  mM3NDBlNDU4YzMiLCJpYXQiOjE3NDczOTk3NjEsImV4cCI6MTc0ODI2Mzc2MX0.eG2N0oJQ
  -vyP155BaQ-Wj07FIC1d6Zt8L1K65ONZD3w"
}
```

12: Weak Password Policy

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	A Weak Password Policy is a security vulnerability that arises when a system allows the creation of passwords that are too simple, too short, or easily guessable. This can make it easier for attackers to gain unauthorized access to user accounts through methods like brute force, dictionary attacks, or social engineering.
Detailed observation / Vulnerable point	Attackers can easily guess or brute-force weak passwords. Attackers gaining access to sensitive user data, internal systems, or administrative accounts.
CVE/CWE	CWE-521
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium

Recommendation	Require passwords to be at least 12 characters long. Encourage or require the use of a combination of uppercase letters, lowercase letters, numbers, and special characters.
Reference	https://www.acunetix.com/vulnerabilities/web/weak-password/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

Request

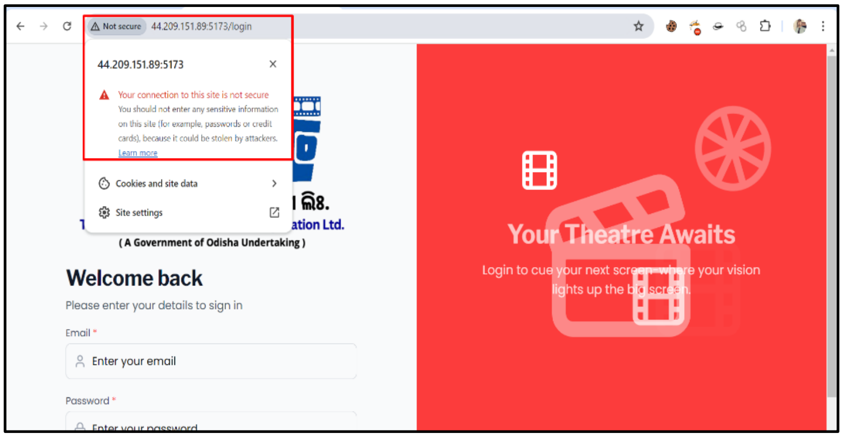
```
POST /api/v1/user/register HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 168
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "name":"1\"><img src=x onerror=alert(123)>",
  "email":"ralema5803@hazhab.com",
  "password":"123456",
  "confirmPassword":"123456",
  "phoneNo":"0000000000",
  "termsAccepted":true
}
```

Response

```
HTTP/1.1 201 Created
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 384
ETag: W/"180-RNUoYopwaW/FH46NauBVGE3tH+Y"
Date: Fri, 16 May 2025 12:49:21 GMT
Connection: close

{
  "message":"User registered successfully",
  "user":{
    "_id":"68273451166fd2c740e458c3",
    "name":"1\"><img src=x onerror=alert(123)>",
    "email":"ralema5803@hazhab.com",
    "phoneNo":"0000000000",
    "role":"USER"
  },
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOiI2ODI3MzQ1MTE2NmZkMmM3NDBlNDU4YzMiLCJpYXQiOjE3NDczOTk3NjEsImV4cCI6MTc0ODI2Mzc2MX0.eG2N0oJQ-vyPl55BaQ-WjO7FIC1d6Zt8L1K65ONZD3w"
}
```

13: Unencrypted Communication

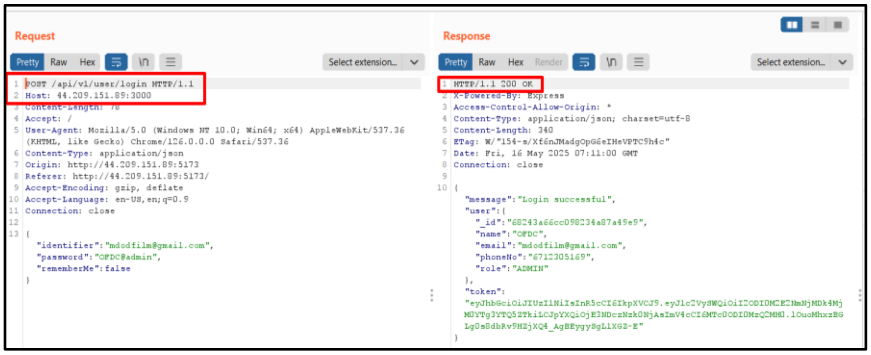
Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Unencrypted communication occurs when sensitive data, such as passwords, personal details, or payment information, is transmitted over the network without encryption. This means the data is sent in plaintext and can be intercepted and read by malicious actors.
Detailed observation / Vulnerable point	Attackers can read sensitive data transmitted over the network, including passwords, personal details, financial data, or authentication tokens.
CVE/CWE	CWE-320
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Medium

Recommendation	Ensure that all web traffic is encrypted with TLS (Transport Layer Security), previously known as SSL. Configure web servers to force HTTPS and redirect all HTTP traffic to HTTPS.
Reference	https://www.acunetix.com/vulnerabilities/web/ssl-tls-not-implemented/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Not secure 44.209.151.89:5173/login 44.209.151.89:5173 Your connection to this site is not secure You should not enter any sensitive information on this site (for example, passwords or credit cards), because it could be stolen by attackers. Learn more Cookies and site data Site settings (A Government of Odisha Undertaking) Welcome back Please enter your details to sign in Email * Enter your email Password * Enter your password Your Theatre Awaits Login to cue your next screen—where your vision lights up the big screen.

14: Missing Security Headers

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173/
Observation/ Vulnerability title	Missing Security Headers is a common web application misconfiguration where the server response does not include HTTP headers that instruct browsers to behave securely. These headers help prevent a range of attacks including cross-site scripting (XSS), clickjacking, content sniffing, and data interception.
Detailed observation / Vulnerable point	Without Content-Security-Policy, attackers may inject malicious scripts into web pages.
CVE/CWE	CWE-693
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

Recommendation	Implement Security Headers, Add recommended security headers to your web server configuration or application response. Missing Content Security Policy (CSP) Header: default-src 'self'; script-src 'self' https://trusted.cdn.com Missing HTTP Strict-Transport-Security (HSTS) Header: max-age=31536000 Missing Permissions-Policy Header: geolocation=(self), microphone=(), camera=(), Fullscreen=(self) Missing Referrer-Policy Header: no-referrer Missing X-Content-Type-Options Header: no sniff Missing X-Frame-Options Header: SAMEORIGIN or DENY Missing X-Permitted-Cross-Domain-Policies: none or master-only or by-content-type
Reference	https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Headers_Cheat_Sheet.html
New or Repeat observation	New Observation

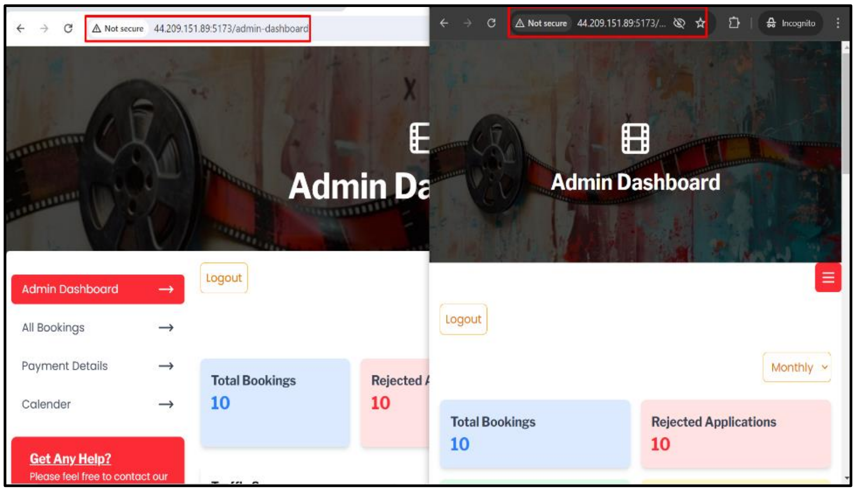
References to evidences / Proof of Concept	

```
POST /api/v1/user/login HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 78
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "identifier":"mdodfilm@gmail.com",
  "password":"OFDC@admin",
  "rememberMe":false
}
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 340
ETag: W/"154-s/Xf6nJMadgOpG6eIHeVPTC9h4c"
Date: Fri, 16 May 2025 07:11:00 GMT
Connection: close

{
  "message":"Login successful",
  "user":{
    "_id":"68243a66cc098234a87a49e9",
    "name":"OFDC",
    "email":"mdodfilm@gmail.com",
    "phoneNo":"6712305169",
    "role":"ADMIN"
  },
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOiI2ODI0M2E2NmNjMDk4Mj
  M0YTg3YTQ5ZTkiLCJpYXQiOjE3NDczNzk0NjAsImV4cCI6MTc0ODI0MzQ2MH0.lOuoMhxzBG
  LgDs8dbRv9HZjXQ4_AgBEygySgL1XGZ-E"
}
```

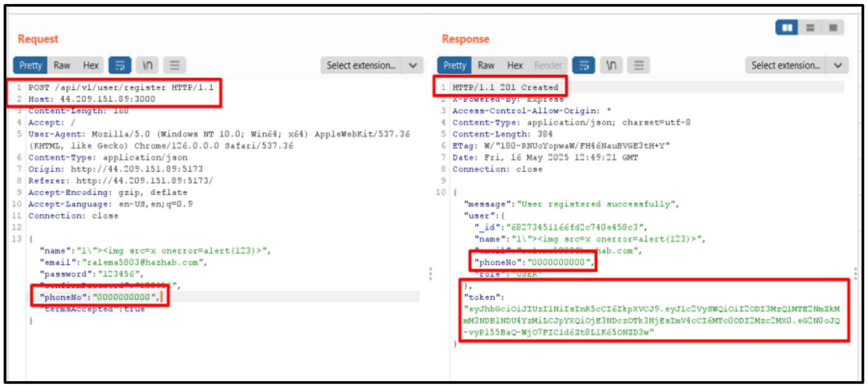
15. Concurrent Login

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	Concurrent login vulnerabilities arise when a system allows a user to be logged in from multiple devices or sessions simultaneously without any restrictions or proper controls. While some applications may require users to log out from one device before logging in from another, systems without restrictions on concurrent logins may introduce a variety of security issues.
Detailed observation / Vulnerable point	Multiple devices or sessions might allow attackers to gain persistent access even if one session is closed.
CVE/CWE	CWE-693
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

Recommendation	Implement a policy to allow only a single active session per user at a time (or a limited number of sessions). This can help prevent multiple logins and control access to the user account more securely.
Reference	https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/06-Session_Management_Testing/11-Testing_for_Concurrent_Sessions
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

16: Lack of Phone number Verification

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	Lack of phone number verification occurs when an application or system does not validate or confirm the phone number provided by the user during account registration, account recovery, or other processes. This verification typically involves sending a one-time passcode (OTP) or verification code to the phone number to confirm its ownership.
Detailed observation / Vulnerable point	Attackers can create fake or fraudulent accounts or bypass account recovery processes if they can guess or manipulate the phone number.
CVE/CWE	CWE-284
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

Recommendation	When users enter a phone number, send a one-time passcode (OTP) to that number. The user must then enter the correct OTP to verify ownership of the phone number.
Reference	https://cwe.mitre.org/data/definitions/345.html
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

Request

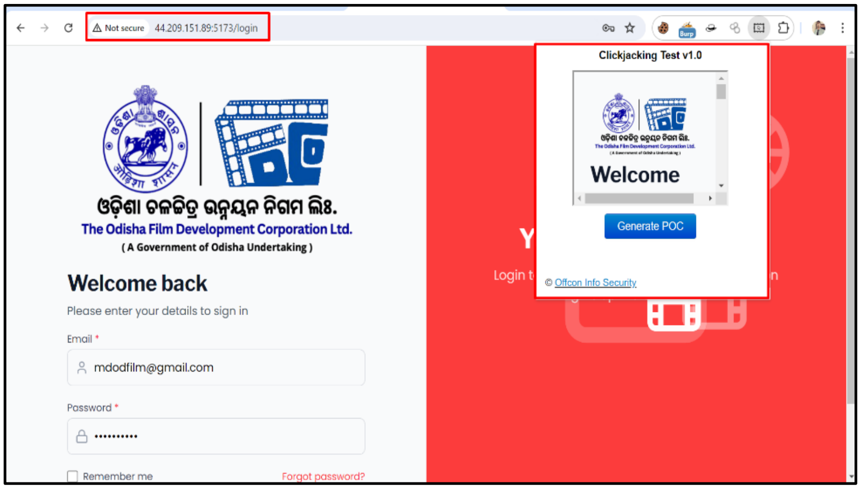
```
POST /api/v1/user/register HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 168
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "name":"1\"><img src=x onerror=alert(123)>",
  "email":"ralema5803@hazhab.com",
  "password":"123456",
  "confirmPassword":"123456",
  "phoneNo":"0000000000",
  "termsAccepted":true
}
```

Response

```
HTTP/1.1 201 Created
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 384
ETag: W/"180-RNUoYopwaW/FH46NauBVGE3tH+Y"
Date: Fri, 16 May 2025 12:49:21 GMT
Connection: close

{
  "message":"User registered successfully",
  "user":{
    "_id":"68273451166fd2c740e458c3",
    "name":"1\"><img src=x onerror=alert(123)>",
    "email":"ralema5803@hazhab.com",
    "phoneNo":"0000000000",
    "role":"USER"
  },
  "token":
"eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2ODI3MzQ1MTE2NmZkMmM3NDBlNDU4YzMiLCJpYXQiOjE3NDczOTk3NjEsImV4cCI6MTc0ODI2Mzc2MX0.eG2N0oJQ-vyPl55BaQ-WjO7FIC1d6Zt8L1K65ONZD3w"
}
```

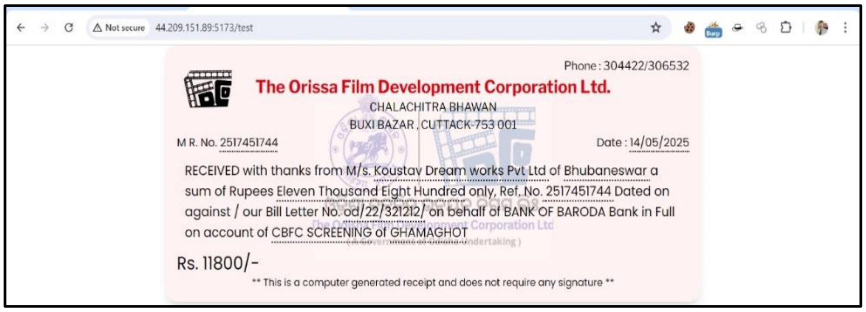
17: Clickjacking Attack

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	In a clickjacking attack, the attacker embeds a transparent frame (iframe) or element over a legitimate page. The user might think they are clicking a button or link on the page, but in reality, they are interacting with the transparent, hidden element underneath. The attacker can then perform malicious actions on behalf of the user without their knowledge.
Detailed observation / Vulnerable point	Users might unknowingly perform harmful actions, such as liking or sharing something on social media, making purchases, or sending sensitive information.
CVE/CWE	CWE-1021
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

Recommendation	The X-Frame-Options HTTP header prevents your content from being embedded into an iframe. This can block clickjacking attacks by disallowing your page from being embedded on malicious sites.
Reference	https://owasp.org/www-community/attacks/Clickjacking
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Not secure 44.209.151.89:5173/login Clickjacking Test v1.0 Welcome Generate POC © Offcon Info Security ଓଡ଼ିଶା ଚଳଚ୍ଚିତ୍ର ଉନ୍ନୟନ ନିଗମ ଲିଃ. The Odisha Film Development Corporation Ltd. (A Government of Odisha Undertaking) Welcome back Please enter your details to sign in Email * mdodfilm@gmail.com Password * •••••••••• □ Remember me Forgot password?

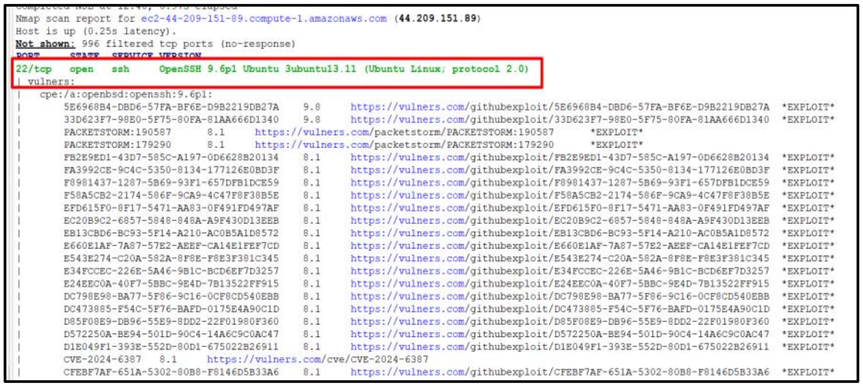
18: Exposed Directory File

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	When the server is misconfigured and reveals a list of files within a directory without restrictions, allowing attackers to browse and potentially download files.
Detailed observation / Vulnerable point	Attackers can view and download sensitive files, potentially gaining access to critical application configurations, database information, or source code.
CVE/CWE	CWE-548
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	🟩 Low
Recommendation	Disable directory listing on the web server (e.g., Apache, Nginx). This prevents attackers from viewing the contents of directories that do not have an index file (e.g., index.html, index.php).

Reference	https://www.acunetix.com/vulnerabilities/web/directory-listings/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 ⚠ Not secure 44.209.151.89:5173/test Phone : 304422/306532 The Orissa Film Development Corporation Ltd. CHALACHITRA BHAWAN BUXI BAZAR , CUTTACK-753 001 M R. No. 2517451744 Date : 14/05/2025 RECEIVED with thanks from M/s. Koustav Dream works Pvt Ltd of Bhubaneswar a sum of Rupees Eleven Thousand Eight Hundred only, Ref. No. 2517451744 Dated on against / our Bill Letter No. od/22/321212/ on behalf of BANK OF BARODA Bank in Full on account of CBFC SCREENING of GHAMAGHOT Rs. 11800/- ** This is a computer generated receipt and does not require any signature **

19: Outdated Component Used

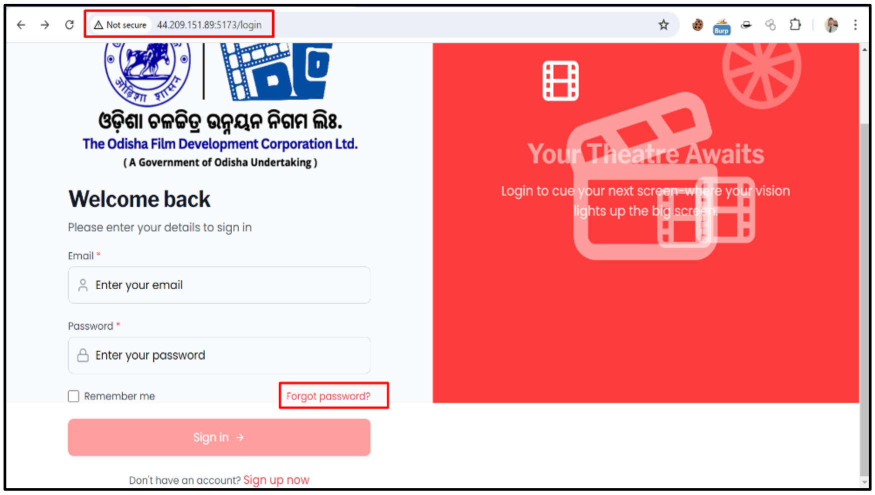
Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	An Outdated Component Used vulnerability occurs when an application relies on outdated software libraries, frameworks, or components that no longer receive security patches or updates. These components might contain known vulnerabilities that attackers can exploit, and using them can leave an application exposed to attacks.
Detailed observation / Vulnerable point	Attackers can exploit known vulnerabilities in outdated components to gain unauthorized access, execute arbitrary code, or perform other malicious activities.
CVE/CWE	CWE-1104
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

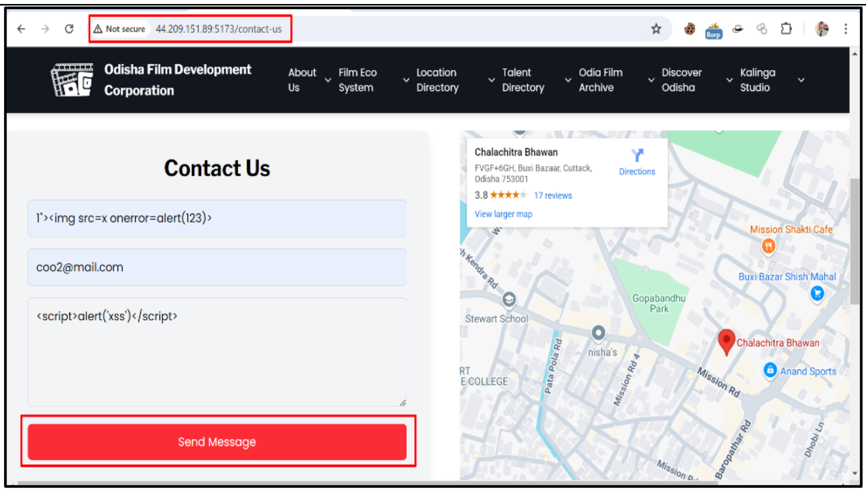
Recommendation	Ensure that all components (e.g., libraries, frameworks, APIs) are updated to the latest stable version that has received security patches.
Reference	https://owasp.org/Top10/A06_2021-Vulnerable_and_Outdated_Components/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

```
Nmap scan report for ec2-44-209-151-89.compute-1.amazonaws.com (44.209.151.89)
Host is up (0.25s latency).
Not shown: 996 filtered tcp ports (no-response)
PORT     STATE  SERVICE VERSION
22/tcp   open   ssh     OpenSSH 9.6p1 Ubuntu 3ubuntu13.11 (Ubuntu Linux; protocol 2.0)
| vulners:
|   cpe:/a:openbsd:openssh:9.6p1:
|       5E6968B4-DBD6-57FA-BF6E-D9B2219DB27A    9.8     https://vulners.com/githubexploit/5E6968B4-DBD6-57FA-BF6E-D9B2219DB27A  *EXPLOIT*
|       33D623F7-98E0-5F75-80FA-81AA666D1340    9.8     https://vulners.com/githubexploit/33D623F7-98E0-5F75-80FA-81AA666D1340  *EXPLOIT*
|       PACKETSTORM:190587      8.1     https://vulners.com/packetstorm/PACKETSTORM:190587      *EXPLOIT*
|       PACKETSTORM:179290      8.1     https://vulners.com/packetstorm/PACKETSTORM:179290      *EXPLOIT*
|       FB2E9ED1-43D7-585C-A197-0D6628B20134    8.1     https://vulners.com/githubexploit/FB2E9ED1-43D7-585C-A197-0D6628B20134  *EXPLOIT*
|       FA3992CE-9C4C-5350-8134-177126E0BD3F    8.1     https://vulners.com/githubexploit/FA3992CE-9C4C-5350-8134-177126E0BD3F  *EXPLOIT*
|       F8981437-1287-5B69-93F1-657DFB1DCE59    8.1     https://vulners.com/githubexploit/F8981437-1287-5B69-93F1-657DFB1DCE59  *EXPLOIT*
|       F58A5CB2-2174-586F-9CA9-4C47F8F38B5E    8.1     https://vulners.com/githubexploit/F58A5CB2-2174-586F-9CA9-4C47F8F38B5E  *EXPLOIT*
|       EFD615F0-8F17-5471-AA83-0F491FD497AF    8.1     https://vulners.com/githubexploit/EFD615F0-8F17-5471-AA83-0F491FD497AF  *EXPLOIT*
|       EC20B9C2-6857-5848-848A-A9F430D13EEB    8.1     https://vulners.com/githubexploit/EC20B9C2-6857-5848-848A-A9F430D13EEB  *EXPLOIT*
|       EB13CBD6-BC93-5F14-A210-AC0B5A1D8572    8.1     https://vulners.com/githubexploit/EB13CBD6-BC93-5F14-A210-AC0B5A1D8572  *EXPLOIT*
|       E660E1AF-7A87-57E2-AEEF-CA14E1FEF7CD    8.1     https://vulners.com/githubexploit/E660E1AF-7A87-57E2-AEEF-CA14E1FEF7CD  *EXPLOIT*
|       E543E274-C20A-582A-8F8E-F8E3F381C345    8.1     https://vulners.com/githubexploit/E543E274-C20A-582A-8F8E-F8E3F381C345  *EXPLOIT*
|       E34FCCEC-226E-5A46-9B1C-BCD6EF7D3257    8.1     https://vulners.com/githubexploit/E34FCCEC-226E-5A46-9B1C-BCD6EF7D3257  *EXPLOIT*
|       E24EEC0A-40F7-5BBC-9E4D-7B13522FF915    8.1     https://vulners.com/githubexploit/E24EEC0A-40F7-5BBC-9E4D-7B13522FF915  *EXPLOIT*
|       DC798E98-BA77-5F86-9C16-0CF8CD540EBB    8.1     https://vulners.com/githubexploit/DC798E98-BA77-5F86-9C16-0CF8CD540EBB  *EXPLOIT*
|       DC473885-F54C-5F76-BAFD-0175E4A90C1D    8.1     https://vulners.com/githubexploit/DC473885-F54C-5F76-BAFD-0175E4A90C1D  *EXPLOIT*
|       D85F08E9-DB96-55E9-8DD2-22F01980F360    8.1     https://vulners.com/githubexploit/D85F08E9-DB96-55E9-8DD2-22F01980F360  *EXPLOIT*
|       D572250A-BE94-501D-90C4-14A6C9C0AC47    8.1     https://vulners.com/githubexploit/D572250A-BE94-501D-90C4-14A6C9C0AC47  *EXPLOIT*
|       D1E049F1-393E-552D-80D1-675022B26911    8.1     https://vulners.com/githubexploit/D1E049F1-393E-552D-80D1-675022B26911  *EXPLOIT*
|       CVE-2024-6387   8.1     https://vulners.com/cve/CVE-2024-6387
|       CFEBF7AF-651A-5302-80B8-F8146D5B33A6    8.1     https://vulners.com/githubexploit/CFEBF7AF-651A-5302-80B8-F8146D5B33A6  *EXPLOIT*
```

20: Functionality Not Working

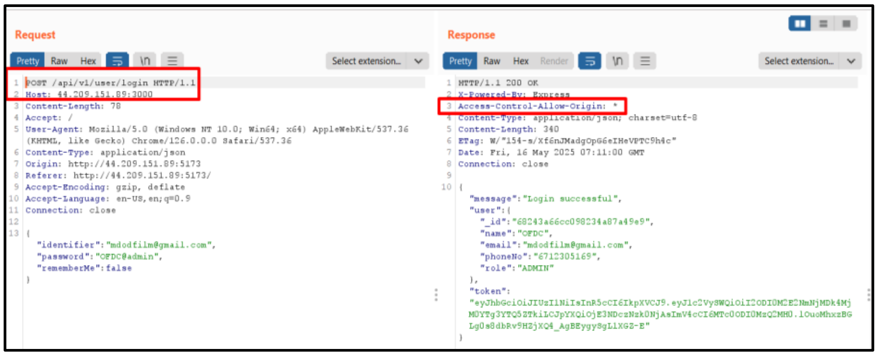
Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	A clickable element like a button or link doesn't trigger the expected action. A series of processes or steps in an application do not proceed as expected.
Detailed observation / Vulnerable point	When key functionality does not work, users may get frustrated, leading to a poor user experience. Non-functional features can directly impact business processes, such as losing the ability to process transactions or sign up users.
CVE/CWE	CWE-802
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

Recommendation	Conduct unit testing, integration testing, and user acceptance testing (UAT) to ensure that all features are functioning as expected in different environments and scenarios.
Reference	https://www.invicti.com/web-vulnerability-scanner/vulnerabilities/classification/cwe-22/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Not secure 44.209.151.89:5173/login ଓଡ଼ିଶା ଚଳଚ୍ଚିତ୍ର ଉନ୍ନୟନ ନିଗମ ଲିଃ. The Odisha Film Development Corporation Ltd. (A Government of Odisha Undertaking) Welcome back Please enter your details to sign in Email * Enter your email Password * Enter your password □ Remember me Forgot password? Sign in → Don't have an account? Sign up now Your Theatre Awaits Login to cue your next screen—where your vision lights up the big screen.

	 Not secure 44.209.151.89:5173/contact-us Contact Us 1'"><img src=x onerror=alert(123)> coo2@mail.com <script>alert('xss')</script> Odisha Film Development Corporation About Us Film Eco System Location Directory Talent Directory Odia Film Archive Discover Odisha Kalinga Studio Send Message Chalachitra Bhawan FVGF+6GH, Buxi Bazaar, Cuttack, Odisha 753001 3.8 ★★★★☆ 17 reviews Directions View larger map Mission Shakti Cafe Buxi Bazar Shish Mahal Gopabandhu Park Stewart School nisha's Anand Sports Mission Rd Pata Pola Rd Baropathar Rd Dhobi Ln

21 : Allow origin Allow origin Misconfiguration

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	The Allow-Origin Misconfiguration vulnerability is typically caused by an incorrect configuration of the CORS (Cross-Origin Resource Sharing) policy in web applications. CORS is a security feature implemented by web browsers to restrict web pages from making requests to domains different from the one that served the web page. It's a mechanism that allows a server to specify which domains can access its resources.
Detailed observation / Vulnerable point	Malicious websites can access sensitive user data (e.g., authentication tokens, user information) from the vulnerable web application, leading to data theft or information leakage.
CVE/CWE	CWE-693
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

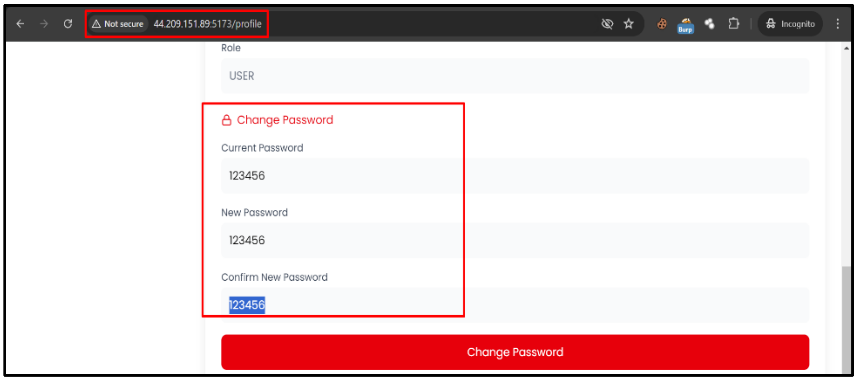
Recommendation	Avoid using the wildcard (*) for the Access-Control-Allow-Origin header. If the origin must be dynamic (i.e., vary based on the request), implement logic to check the Origin header and ensure it matches a whitelist of trusted origins.
Reference	https://www.acunetix.com/vulnerabilities/web/misconfigured-access-control-allow-origin-header/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

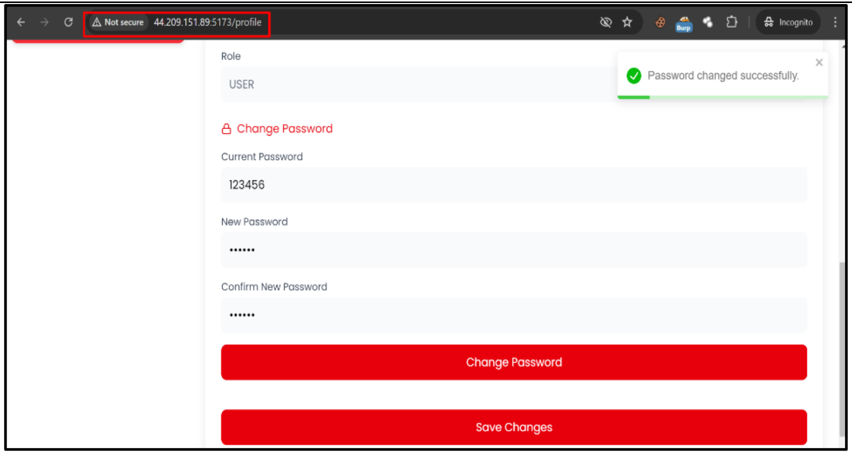
```
POST /api/v1/user/login HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 78
Accept: /
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "identifier":"mdodfilm@gmail.com",
  "password":"OFDC@admin",
  "rememberMe":false
}
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 340
ETag: W/"154-s/Xf6nJMadgOpG6eIHeVPTC9h4c"
Date: Fri, 16 May 2025 07:11:00 GMT
Connection: close

{
  "message":"Login successful",
  "user":{
    "_id":"68243a66cc098234a87a49e9",
    "name":"OFDC",
    "email":"mdodfilm@gmail.com",
    "phoneNo":"6712305169",
    "role":"ADMIN"
  },
  "token":
  "eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJlc2VySWQiOiI2ODI0M2E2NmNjMDk4MjM0YTg3YTQ5ZTkiLCJpYXQiOjE3NDczNzk0NjAsImV4cCI6MTc0ODI0MzQ2MH0.lOuoMhxzBGLg0s8dbRv9HZjXQ4_AgBEygySgLlXGZ-E"
}
```

22 : Old Password Set as New Password

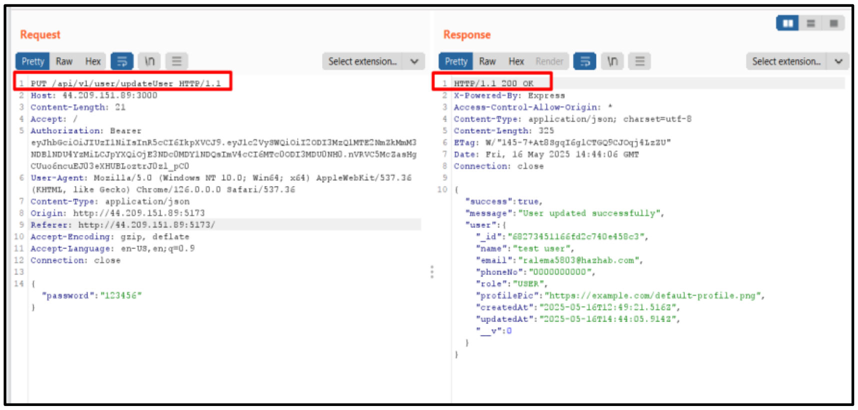
Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	The Old Password Set as New Password vulnerability occurs when a user can set their old password as their new password during the password change process. This happens because the application does not properly validate that the new password is different from the current password, which allows the user to essentially keep their old password.
Detailed observation / Vulnerable point	An attacker who gains access to an account will not be forced to change the password if they exploit this vulnerability. They could retain full control of the account, even if the user or the system attempts to enforce a password change.
CVE/CWE	CWE-640
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	🟩 Low

Recommendation	When users attempt to change their password, ensure that the new password is different from the old password. This is a simple yet crucial validation check.
Reference	https://owasp.org/www-project-web-security-testing-guide/latest/4-Web_Application_Security_Testing/04-Authentication_Testing/09-Testing_for_Weak_Password_Change_or_Reset_Functionalities
New or Repeat observation	New Observation
References to evidences / Proof of Concept	 Not secure 44.209.151.89:5173/profile Role USER Change Password Current Password 123456 New Password 123456 Confirm New Password 123456

	 Not secure 44.209.151.89:5173/profile Role USER Password changed successfully. Change Password Current Password 123456 New Password •••••• Confirm New Password ••••••

23: Unwanted http Method Enable

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	The Unwanted HTTP Methods Enabled vulnerability occurs when an application or web server is configured to accept HTTP methods that are unnecessary or unsafe, such as DELETE, PUT, TRACE, or OPTIONS, for endpoints that should only support GET or POST. This misconfiguration can expose the application to a variety of attacks.
Detailed observation / Vulnerable point	If a server accepts HTTP methods that should not be available to the public, attackers could modify or delete sensitive resources. For example, if an attacker sends a DELETE request to a vulnerable resource endpoint, they might be able to remove critical data.
CVE/CWE	CWE-444
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low

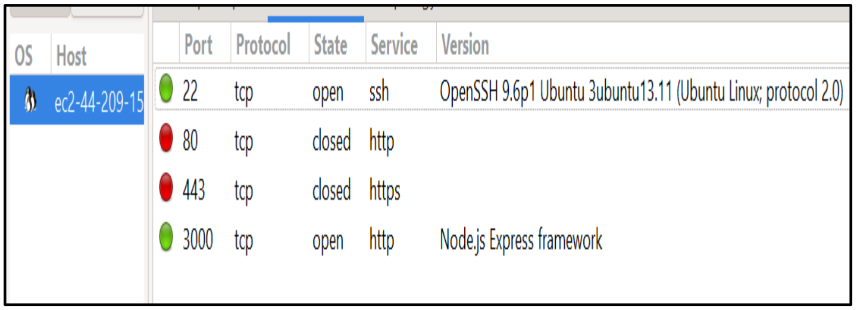
Recommendation	Review the HTTP methods allowed by your web server and application, and disable any that are unnecessary for the functioning of the application.
Reference	https://owasp.org/www-project-web-security-testing-guide/v41/4-Web_Application_Security_Testing/02-Configuration_and_Deployment_Management_Testing/06-Test_HTTP_Methods
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

```
PUT /api/v1/user/updateUser HTTP/1.1
Host: 44.209.151.89:3000
Content-Length: 21
Accept: /
Authorization: Bearer
eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJ1c2VySWQiOiI2ODI3MzQ1MTE2NmZkMmM3
NDBlNDU4YzMiLCJpYXQiOjE3NDc0MDY1NDQsImV4cCI6MTc0ODI3MDU0NH0.nVRVC5McZasHg
CUuo6ncuEJ03eXHUBLoztrJ0zl_pC0
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
(KHTML, like Gecko) Chrome/126.0.0.0 Safari/537.36
Content-Type: application/json
Origin: http://44.209.151.89:5173
Referer: http://44.209.151.89:5173/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Connection: close

{
  "password":"123456"
}
```

```
HTTP/1.1 200 OK
X-Powered-By: Express
Access-Control-Allow-Origin: *
Content-Type: application/json; charset=utf-8
Content-Length: 325
ETag: W/"145-7+At8SgqI6glCTGQ9CJOqj4LzZU"
Date: Fri, 16 May 2025 14:44:06 GMT
Connection: close

{
  "success":true,
  "message":"User updated successfully",
  "user":{
    "_id":"68273451166fd2c740e458c3",
    "name":"test user",
    "email":"ralema5803@hazhab.com",
    "phoneNo":"0000000000",
    "role":"USER",
    "profilePic":"https://example.com/default-profile.png",
    "createdAt":"2025-05-16T12:49:21.516Z",
    "updatedAt":"2025-05-16T14:44:05.914Z",
    "__v":0
  }
}
```

24 : Open port

Affected Asset i.e. IP / URL / Application etc.	http://44.209.151.89:5173//
Observation/ Vulnerability title	An Open Port vulnerability occurs when unnecessary or unsecured network ports are left open on a system, device, or server. These open ports can be exploited by attackers to gain unauthorized access to the system, steal sensitive data, or perform malicious activities.
Detailed observation / Vulnerable point	If an attacker can exploit an open port, they might gain unauthorized access to the system, bypassing security controls. For example, an open SSH port might allow an attacker to brute force a login and gain access to a server.
CVE/CWE	CWE-200
Control Objective	NA
Control Name	NA
Audit Requirement	NA
Severity	Low
Recommendation	Identify and close any open ports that are not needed for legitimate purposes. This includes ports related to unused services, outdated

	protocols, or any service that does not need to be exposed to the internet.
Reference	https://mas.owasp.org/MASWE/MASVS-NETWORK/MASWE-0051/
New or Repeat observation	New Observation
References to evidences / Proof of Concept	

OS	Host
	ec2-44-209-15

	Port	Protocol	State	Service	Version
●	22	tcp	open	ssh	OpenSSH 9.6p1 Ubuntu 3ubuntu13.11 (Ubuntu Linux; protocol 2.0)
●	80	tcp	closed	http	
●	443	tcp	closed	https	
●	3000	tcp	open	http	Node.js Express framework