MIRA BHAYANDAR MUNICIPAL CORPORATION

Visitor Management
Web Application

Report Release Date	February 12, 2026
Type of Audit	Application Security Assessment
Type of Audit Report	First Audit Report
Period	January 28, 2026 to January 31, 2026

Document Control

Document Preparation	
Document Title	VAPT Report of Visitor Management Web Application
Document ID	A3S/MBMC/ Visitor Management /2526/00011
Document Version	1.0
Prepared by	Jasmeet Singh
Reviewed by	Sagar Gupta
Approved by	Sagar Gupta
Released by	Jasmeet Singh
Release date	February 12, 2026

Document Change History		
Version	**Date**	**Remarks / Reason of change**
1.0	February 12, 2026	New Report

Document Distribution List			
Name	**Organization**	**Designation**	**Email Id**
Rajkumar Gharat	Mira Bhayandar Municipal Corporation	System Manager	it@mbmc.gov.in

SAGAR GUPTA

Digitally signed by SAGAR GUPTA
Date: 2026.02.12 14:06:38 +05'30'

Contents

Table of Contents

1. Introduction

A3S Tech & Co. (A3S) was engaged by Mira Bhayandar Municipal Corporation to perform VAPT, for Visitor Management Web Application. The report highlights gaps identified during the review and recommendations to remediate the gaps.

The objective of Web Application VAPT was to provide independent evaluation of the vulnerabilities in scope to fulfil the objectives of confidentiality, integrity, and availability and to perform controlled attack to assess the immunity level, to assess the overall level of security, discover weak links and provide recommendations and compliance status to vulnerable entities discovered. The report highlights gaps identified during the VAPT review, recommendations, risk ratings and impact of the vulnerabilities.

2. Engagement Scope

Below are the details of assets covered in the scope:

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 – Visitor Management	Not available	Not Available	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Not Available	MUMBAI	Not available	Not available	Not Applicable

3. 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.	Jasmeet Singh	Senior IS Consultant	jasmeet@a3stech.co.in	CEH	Yes

4. Audit Activities and Timelines

The audit was conducted in the following phases:

S. no.	Audit Activity	Timeline
1.	Information Gathering	January 29, 2026
2.	Scanning	January 29, 2026
3.	Information Analysis	January 29, 2026
4.	Vulnerability Assessment	January 30, 2026
5.	Penetration Testing	January 30, 2026
6.	Revalidation Testing	NA

5. Audit Methodology and Criteria / Standard referred for audit

The Audit Approach and Methodology was a Risk based Audit Approach. In a risk-based audit approach, IS auditors are not just relying on risk; they also are also relying on internal and operational controls as well as knowledge of the organization and its business. The audit was conducted based on combination of tools and manual testing. The audit methodology and approach are based on global best practice framework such as OWASP Top 10 Vulnerabilities, OSSTMM, SANS 25, CIS benchmarks. These are globally accepted standard and a benchmark for IT security across a large number of organizations.

List of OWASP vulnerabilities (Web Application) is:

S. no	Attack Type	Description
1.	A1- Broken Access Control	Improperly configured or missing restrictions on authenticated users allow them to access unauthorized functionality or data, such as accessing other users' accounts, viewing sensitive documents, and modifying data and access rights
2.	A2- Cryptographic Failures	Applications and APIs that don't properly protect sensitive data such as financial data, usernames and passwords, or health information, could enable attackers to access such information to commit fraud or steal identities.
3.	A3- Injection	Injection flaws, such as SQL, OS, and LDAP injection, occur when untrusted data is sent to an interpreter as part of a command or query. The attacker's hostile data can trick the interpreter into executing unintended commands or accessing unauthorized data
4.	A4- Insecure Design	Insecure design is a broad category representing different weaknesses, expressed as "missing or

S. no	Attack Type	Description
		ineffective control design". An insecure design cannot be fixed by a perfect implementation as by definition, needed security controls were never created to defend against specific attacks. One of the factors that contribute to insecure design is the lack of business risk profiling inherent in the software or system being developed, and thus the failure to determine what level of security design is required.
5.	A5- Security Misconfiguration	Good security requires having a secure configuration defined and deployed for the application, frameworks, application server, web server, database server, and platform. All these settings should be defined, implemented, and maintained as many are not shipped with secure defaults. This includes keeping all software up to date, including all code libraries used by the application
6.	A6- Vulnerable and Outdated Components	Developers frequently don't know which open source and third-party components are in their applications, making it difficult to update components when new vulnerabilities are discovered. Attackers can exploit an insecure component to take over the server or steal sensitive data.
7.	A7- Identification and Authentication Failures	Application functions related to authentication and session management are often not implemented correctly, allowing attackers to compromise passwords, keys, session tokens, or exploit other implementation flaws to assume other users' identities
8.	A8- Software and Data	Software and data integrity failures relate to code and infrastructure that does not protect against integrity violations. An insecure CI/CD pipeline can introduce the potential for unauthorized access, malicious code,

S. no	Attack Type	Description
	Integrity Failures	or system compromise. Lastly, many applications now include auto-update functionality, where updates are downloaded without sufficient integrity verification and applied to the previously trusted application. Attackers could potentially upload their own updates to be distributed and run on all installations
9.	A9- Security Logging and Monitoring Failures	The time to detect a breach is frequently measured in weeks or months. Insufficient logging and ineffective integration with security incident response systems allow attackers to pivot to other systems and maintain persistent threats
10.	A10- Server-Side Request Forgery	SSRF flaws occur whenever a web application is fetching a remote resource without validating the user-supplied URL. It allows an attacker to coerce the application to send a crafted request to an unexpected destination, even when protected by a firewall, VPN, or another type of network access control list (ACL).

This document is an exception report highlighting the vulnerabilities and their compliance status.

Our review has been based on the assumption that the information provided to us was accurate and complete, as existing at the time of review, and that all relevant information, system access for review, and supporting documents, as asked for by A3S, were shared with us for the area that was subject of the review.

6. Tools/ Software used

S. no.	Name of Tool/Software used	Version of the tool /Software used	Open Source/Licensed
1.	Burp Suite	2025.12.5	Licensed

7. Executive Summary

The details of the vulnerabilities identified during the testing as mentioned as below:

S. No.	Affected Asset i.e. IP/URL/Application etc.	Observation/Vulnerability title	CVE/CWE	Severity	Recommendation	Reference	New or Repeat observation
1	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Brute Force Attack – Improper Restriction of Authentication Attempts	CWE-307	**High**	To mitigate this vulnerability, it is recommended to Implement account lockout after 5–10 failed login attempts. Apply rate limiting on authentication endpoints. Introduce CAPTCHA after multiple failed attempts. Add progressive delay between login attempts. Monitor and log suspicious login behavior. Enforce strong password policies	https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/	New
2	https://nagarkaryav	Missing Rate	CWE-307	**Medium**	It is recommended	https://owasp.org/Top	New

	aliuat.com/ancl_VisitorMGMT/Login.aspx	Limiting on Authentication (Password) Functionality			to implement the following controls. Apply rate limiting on login endpoints (e.g., 5–10 attempts per minute). Enforce temporary IP blocking after threshold breach. Introduce CAPTCHA after multiple failed attempts. Implement progressive delay (exponential backoff). Monitor and alert on abnormal login behavior.	10/A07_2021-Identification_and_Authentication_Failures/	
3	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Improper Input Validation	CWE-20	**Medium**	Implement server-side input validation using strict regex for email fields. Apply output encoding before rendering user input. Use HTML sanitization libraries. Enable Content-Security-Policy (CSP) header. Perform both client-side and server-side validation	https://owasp.org/Top10/A03_2021-Injection/	New

					(server-side is mandatory)		
4	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Improper Input Validation (Missing Input Length Validation)	CWE-20	**Medium**	Enforce strict server-side validation for phone number length Example (India): exactly 10 digits. Reject inputs exceeding the defined minimum and maximum length. Allow only numeric characters. Implement both client-side and server-side validation (server-side mandatory). Display proper validation error messages	https://owasp.org/Top10/A04_2021-Insecure_Design/	New
5	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Clear Text Password Transmission in Login Request	CWE-319	**Medium**	It is strongly recommended to: Enforce HTTPS (TLS 1.2 or higher) across the application. Avoid transmitting passwords in clear text. Implement secure encryption mechanisms for data in transit. Ensure passwords are hashed and salted on the server side. Disable login access over	https://owasp.org/Top10/A02_2021-Cryptographic_Failures/	New

					HTTP. Use secure authentication frameworks (OAuth, SSO, etc.)		
6	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Missing Security Headers	CWE-693	**Medium**	It is recommended to configure and enable the required HTTP security headers at the web server or application level. At a minimum, implement the following: Content-Security-Policy: default-src 'self'; X-Frame-Options: DENY X-Content-Type-Options: nosniff Strict-Transport-Security: max-age=31536000; includeSubDomains Referrer-Policy: no-referrer Permissions-Policy: geolocation=(), camera=(), microphone=()	https://www.invicti.com/blog/web-security/missing-http-security-headers	New
7	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Out of date (jQuery Version)	CWE-1104	**Medium**	Upgrade jQuery to the latest stable version (v3.7.1 or above). Remove unused or legacy jQuery functions. Regularly review third-party libraries for security updates. Implement dependency	https://github.com/jquery/jquery/security/advisories	New

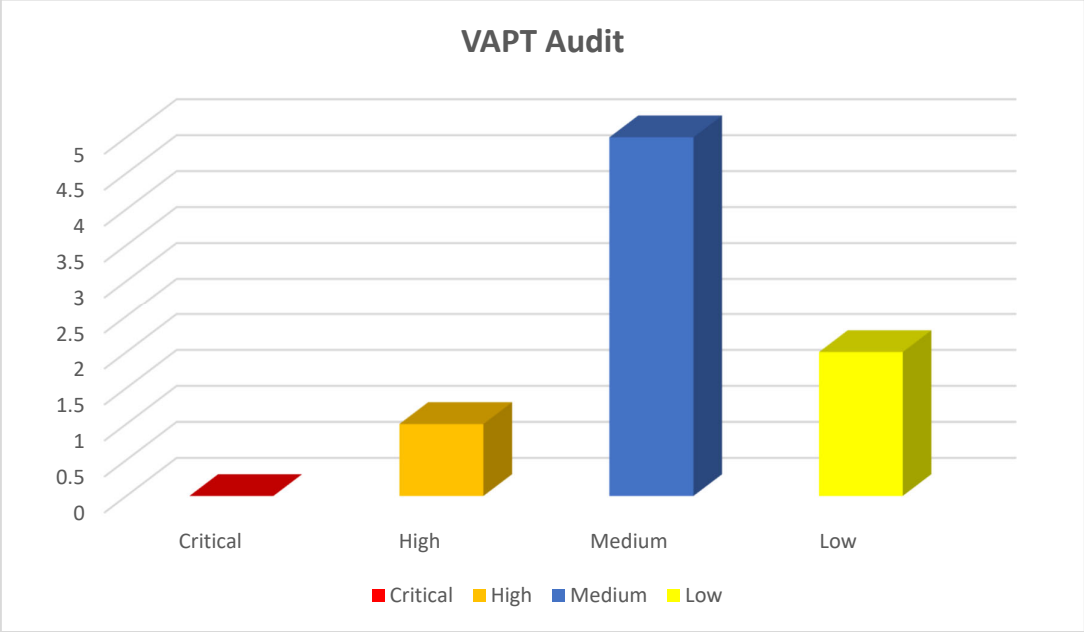
					monitoring as part of the SDLC.		
8	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Weak Ciphers	CWE-326	**Medium**	Disable all weak and legacy cipher suites. Remove CBC-based and RSA key exchange cipher suites. Allow only strong modern cipher suites such as: AES-GCM, CHACHA20-POLY1305. Enforce TLS 1.2 (secure ciphers only) and TLS 1.3. Regularly review SSL/TLS configurations Recommended Cipher Examples: TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256	https://owasp.org/Top10/A02_2021-Cryptographic_Failures/	New
9	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Version Disclosure (jQuery)	CWE-200	**Low**	Avoid exposing exact jQuery version details in production environments. Minify and bundle JavaScript files. Remove version comments and banners from client-side	https://owasp.org/www-community/attacks/Information_Disclosure	New

					resources. Keep jQuery updated to the latest stable version.		
10	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx	Concurrent Login	CWE-306	**Low**	Limit concurrent sessions per account (e.g., 1–3) or allow user-controlled session termination. Invalidate or rotate old sessions when new logins occur (if policy requires single session). Notify users on new-device logins and provide a session management UI.	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

Tabular Representation of the vulnerabilities:

Risk Rating	Count of Observations
Critical	**-**
High	**1**
Medium	**7**
Low	**2**

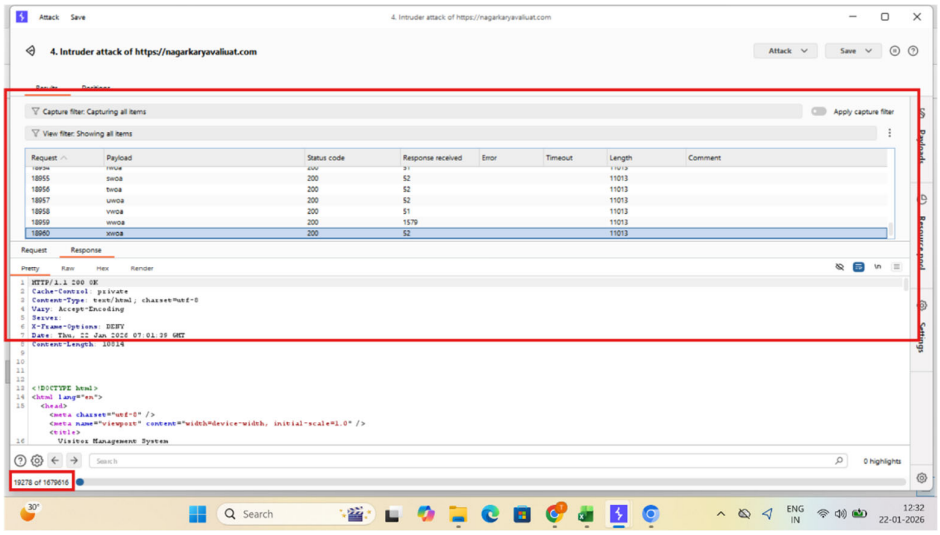
Graphical Representation of vulnerabilities


VAPT Audit
5
4.5
4
3.5
3
2.5
2
1.5
1
0.5
0
Critical
High
Medium
Low
Critical
High
Medium
Low

8. Detailed Observations

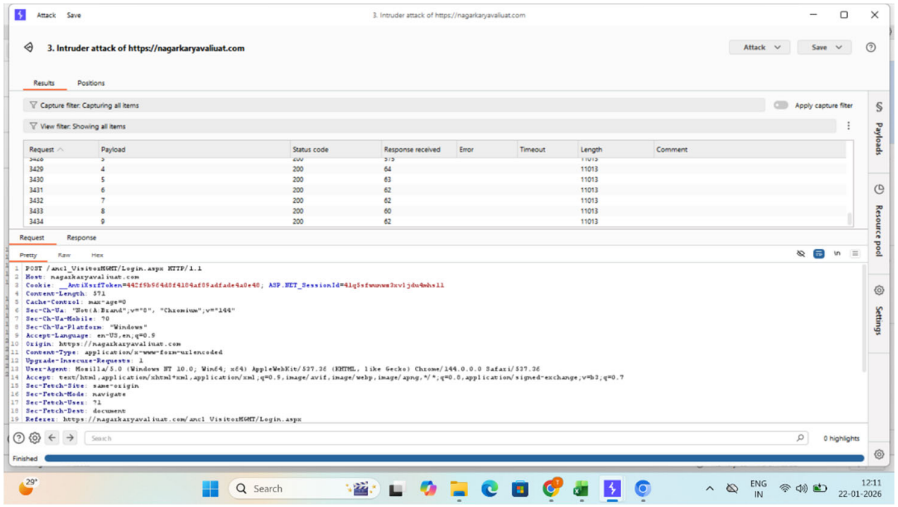
1. **Brute Force Attack – Improper Restriction of Authentication Attempts**

Vulnerability Title	***Affected URLs/IP***
Brute Force Attack – Improper Restriction of Authentication Attempts	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	Multiple login requests were sent with different payloads. The application returned HTTP 200 OK responses for all attempts. No account lockout mechanism was triggered. No CAPTCHA challenge was enforced. No request rate-limiting was observed. Response length and behavior remained consistent across attempts. This confirms that the application allows unlimited authentication attempts.
Vulnerability Reference (CWE/CVE)	CWE-307
Severity	**High**
Recommendation	To mitigate this vulnerability, it is recommended to Implement account lockout after 5–10 failed login attempts. Apply rate limiting on authentication endpoints. Introduce CAPTCHA after multiple failed attempts. Add progressive delay between login attempts. Monitor and log suspicious login behavior. Enforce strong password policies
Reference	https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/
New/ Repeat Observation	New Observation

References to evidences / Proof of Concept (POCs)	

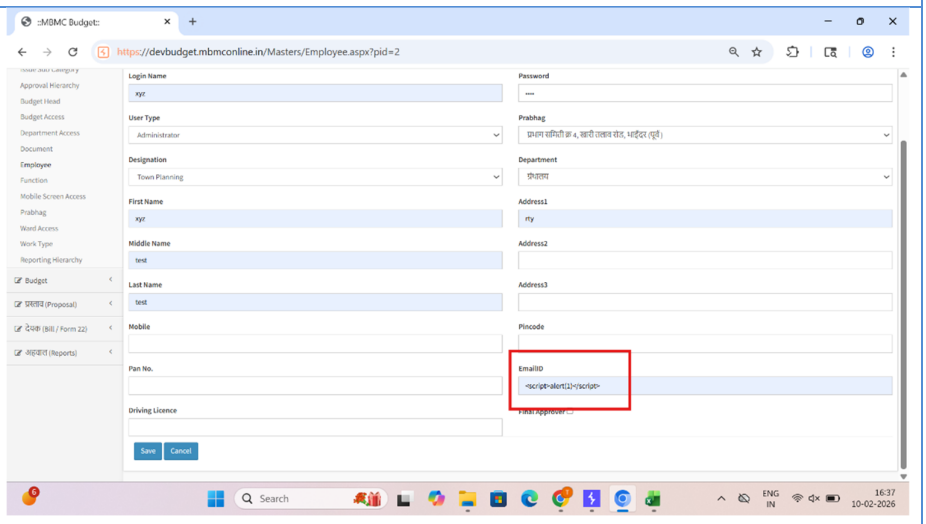
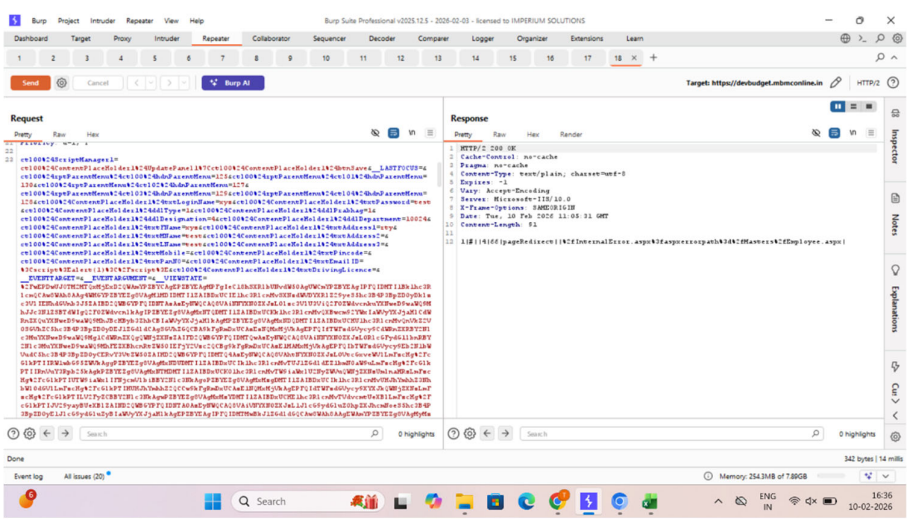
2. Missing Rate Limiting on Authentication (Password) Functionality

Vulnerability Title	***Affected URLs/IP***
Missing Rate Limiting on Authentication (Password) Functionality	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	Rate limiting is a security control used to restrict the number of requests a user or IP address can make within a specific time period. During the security assessment, it was observed that the application does not enforce rate limiting on the login/password endpoint, allowing an attacker to send multiple authentication requests continuously.
Vulnerability Reference (CWE/CVE)	CWE-307
Severity	**Medium**
Recommendation	It is recommended to implement the following controls. Apply rate limiting on login endpoints (e.g., 5–10 attempts per minute). Enforce temporary IP blocking after threshold breach. Introduce CAPTCHA after multiple failed attempts. Implement progressive delay (exponential backoff). Monitor and alert on abnormal login behavior.
Reference	https://owasp.org/Top10/A07_2021-Identification_and_Authentication_Failures/

New/ Repeat Observation	New Observation
References to evidences / Proof of Concept (POCs)	

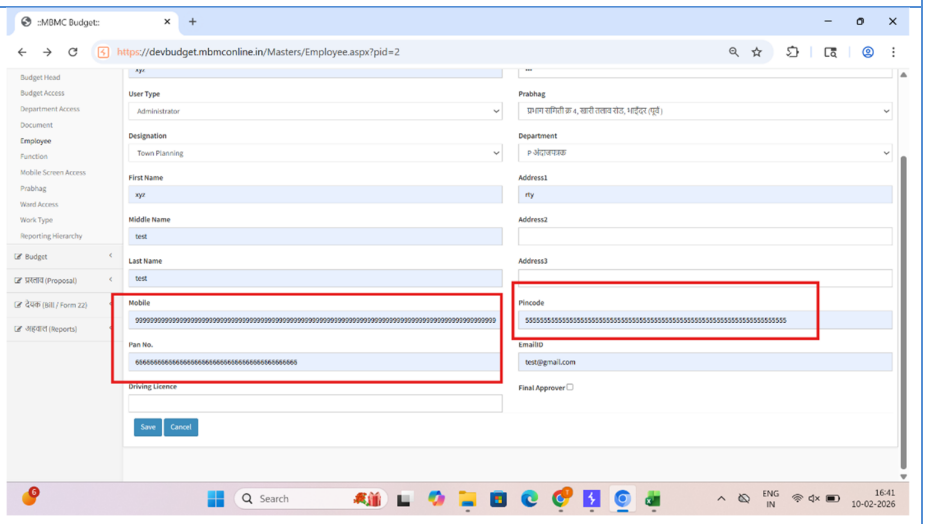
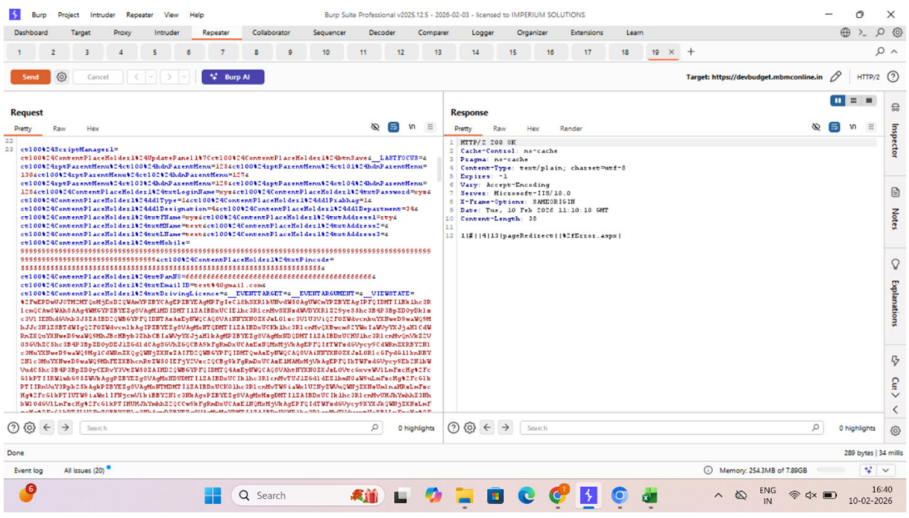
3. Improper Input Validation

Vulnerability Title	***Affected URLs/IP***
Improper Input Validation	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	The application does not properly validate or sanitize user-supplied input in the Email ID field. During testing, HTML/JavaScript payloads were accepted as valid input. This indicates a lack of server-side input validation and output encoding, which may allow attackers to inject and execute malicious scripts in the user's browser.
Vulnerability Reference (CWE/CVE)	CWE-20
Severity	**Medium**
Recommendation	Implement server-side input validation using strict regex for email fields. Apply output encoding before rendering user input. Use HTML sanitization libraries. Enable Content-Security-Policy (CSP) header. Perform both client-side and server-side validation (server-side is mandatory)
Reference	https://owasp.org/Top10/A03_2021-Injection/

New/ Repeat Observation	New Observation
References to evidences / Proof of Concept (POCs)	 

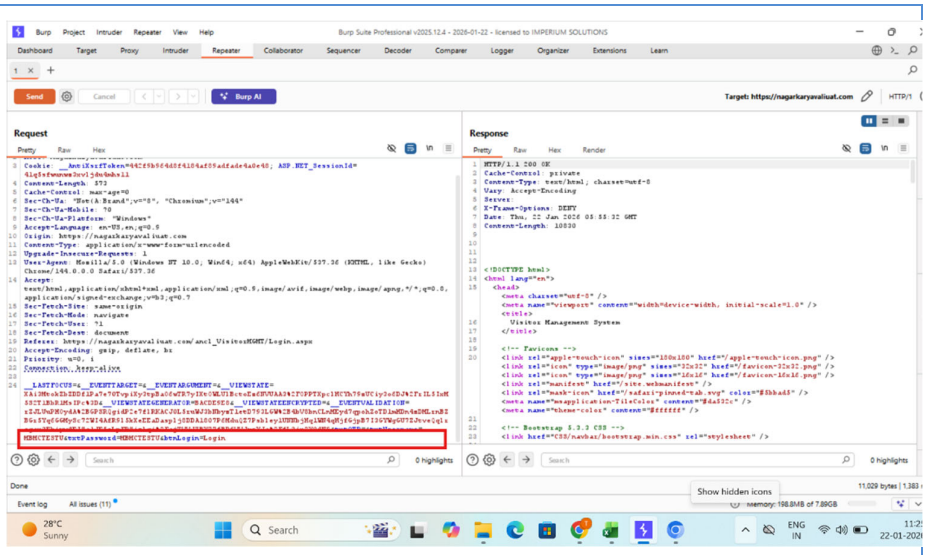
4. Improper Input Validation (Missing Input Length Validation)

Vulnerability Title	***Affected URLs/IP***
Improper Input Validation (Missing Input Length Validation)	https://nagarkaryavaliuat.com/ancl VisitorMGMT/Login.aspx
Detailed Observation	The application does not enforce proper length validation on the phone number field. During testing, the application accepted phone numbers with excessive digits, which indicates improper input validation. A phone number field should strictly accept only a fixed number of digits as per the business and regional requirements.

Vulnerability Reference (CWE/CVE)	CWE-20
Severity	**Medium**
Recommendation	Enforce strict server-side validation for phone number length Example (India): exactly 10 digits. Reject inputs exceeding the defined minimum and maximum length. Allow only numeric characters. Implement both client-side and server-side validation (server-side mandatory). Display proper validation error messages
Reference	https://owasp.org/Top10/A04_2021-Insecure_Design/
New/ Repeat Observation	New Observation
References to evidences / Proof of Concept (POCs)	 

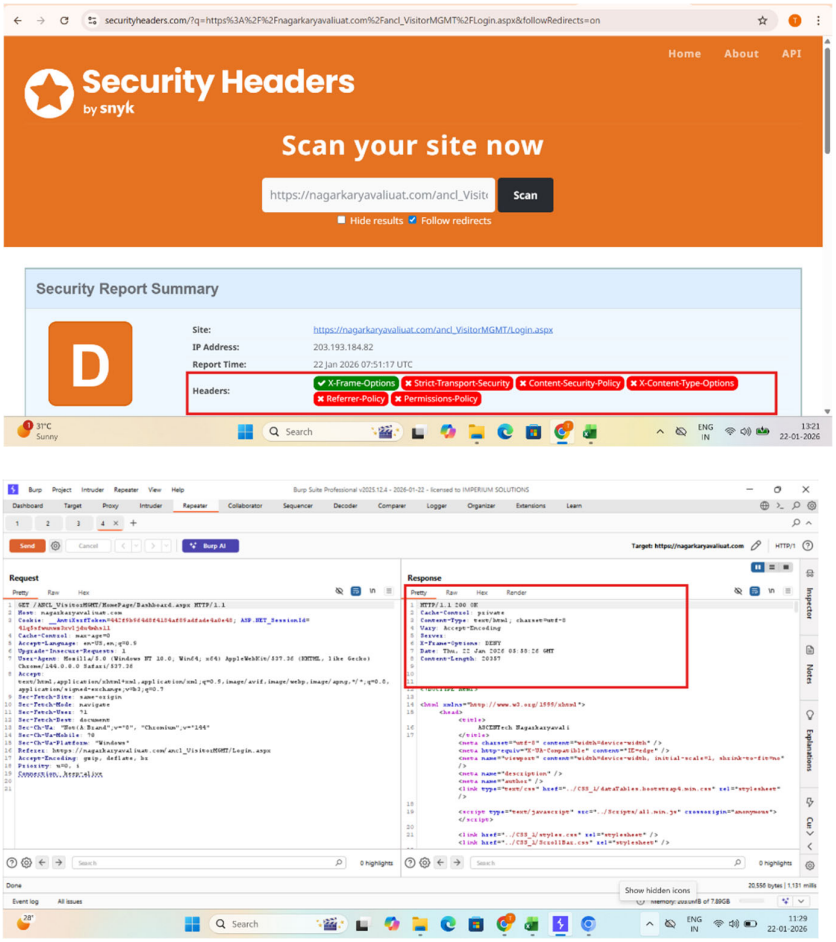
5. Clear Text Password Transmission in Login Request

Vulnerability Title	***Affected URLs/IP***
Clear Text Password Transmission in Login Request	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	Clear text password transmission occurs when user credentials are sent from the client to the server without proper encryption or protection. This allows attackers to view sensitive information such as usernames and passwords in readable format. During the security assessment, it was observed that the application transmits the password parameter in plain readable text within the HTTP request.
Vulnerability Reference (CWE/CVE)	CWE-319
Severity	**Medium**
Recommendation	It is strongly recommended to: Enforce HTTPS (TLS 1.2 or higher) across the application. Avoid transmitting passwords in clear text. Implement secure encryption mechanisms for data in transit. Ensure passwords are hashed and salted on the server side. Disable login access over HTTP. Use secure authentication frameworks (OAuth, SSO, etc.)
Reference	https://owasp.org/Top10/A02_2021-Cryptographic_Failures/
New/ Repeat Observation	New Observation

References to evidences / Proof of Concept (POCs)	

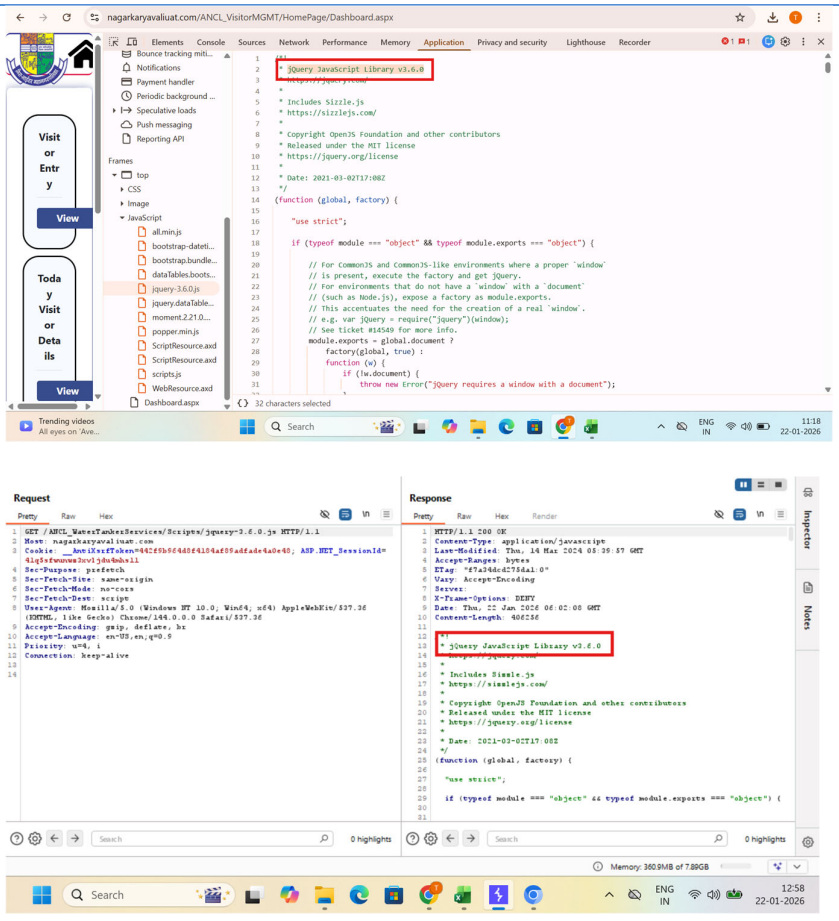
6. Missing Security Headers

Vulnerability Title	***Affected URLs/IP***
Missing Security Headers	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	The application does not implement one or more recommended HTTP security headers. Security headers help browsers enforce security controls that reduce the risk of common web attacks such as Cross-Site Scripting (XSS), Clickjacking, MIME-type sniffing, and information disclosure. Absence of these headers weakens the overall security posture of the application.
Vulnerability Reference (CWE/CVE)	CWE-693
Severity	**Medium**
Recommendation	It is recommended to configure and enable the required HTTP security headers at the web server or application level. At a minimum, implement the following: Content-Security-Policy: default-src 'self'; X-Content-Type-Options: nosniff Strict-Transport-Security: max-age=31536000; includeSubDomains Referrer-Policy: no-referrer Permissions-Policy: geolocation=(), camera=(), microphone=()

Reference	https://www.invicti.com/blog/web-security/missing-http-security-headers
New/ Repeat Observation	New Observation
References to evidences / Proof of Concept (POCs)	

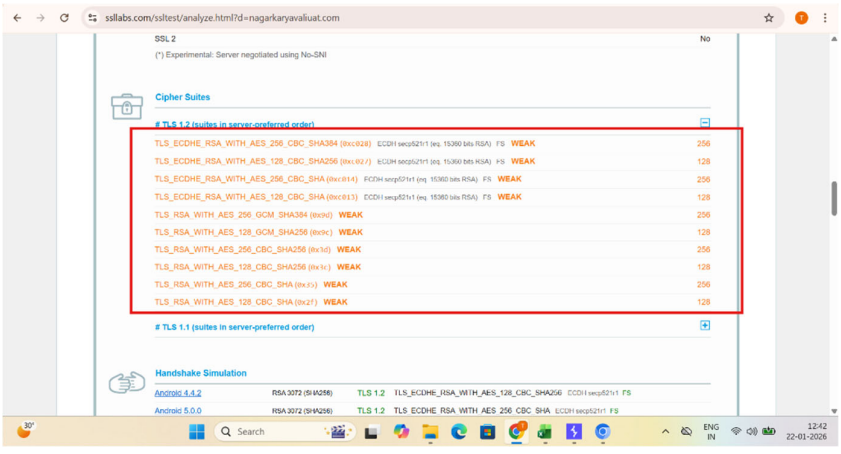
7. Out of Date (jQuery Version)

Vulnerability Title	***Affected URLs/IP***
Out of date (jQuery Version)	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	The web application is using jQuery version 3.6.0, which is an outdated JavaScript library. This version is affected by known security vulnerabilities, including issues related to Cross-Site Scripting (XSS), which have been addressed in later releases. Use of outdated client-side libraries may allow attackers to exploit known weaknesses in the application's front-end components.

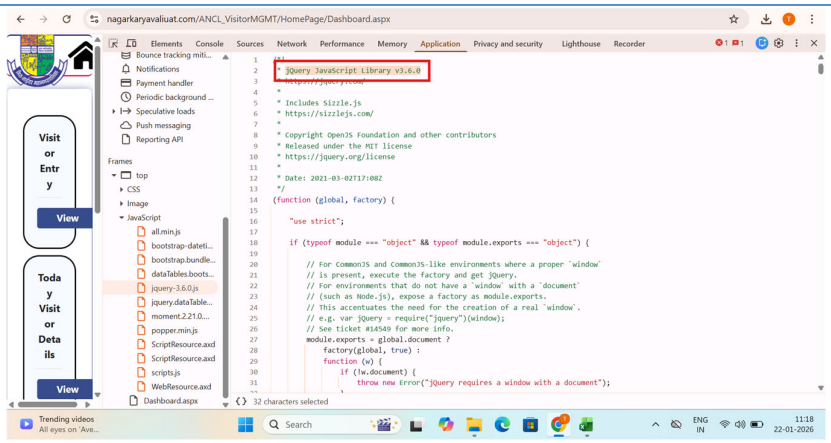
Vulnerability Reference (CWE/CVE)	CWE-1104
Severity	**Medium**
Recommendation	Upgrade jQuery to the latest stable version (v3.7.1 or above). Remove unused or legacy jQuery functions. Regularly review third-party libraries for security updates. Implement dependency monitoring as part of the SDLC.
Reference	https://github.com/jquery/jquery/security/advisories
New/ Repeat Observation	New Observation
References to evidences / Proof of Concept (POCs)	

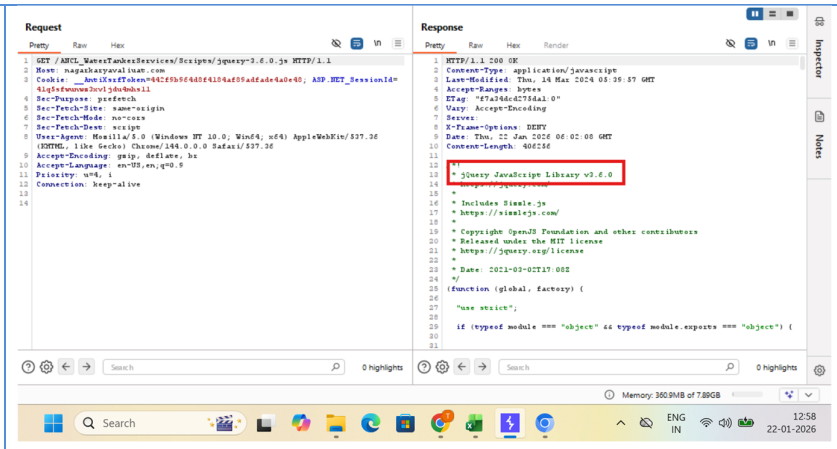
8. Weak Ciphers

Vulnerability Title	***Affected URLs/IP***

Weak Ciphers	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	The application server is configured to support multiple weak SSL/TLS cipher suites under TLS 1.2. These cipher suites rely on outdated cryptographic mechanisms such as CBC mode encryption and RSA key exchange, which are vulnerable to known cryptographic attacks. The presence of these weak cipher suites may allow an attacker to downgrade the encryption strength and potentially compromise the confidentiality and integrity of data transmitted between the client and the server.
Vulnerability Reference (CWE/CVE)	CWE-326
Severity	**Medium**
Recommendation	Disable all weak and legacy cipher suites. Remove CBC-based and RSA key exchange cipher suites. Allow only strong modern cipher suites such as: AES-GCM, CHACHA20-POLY1305. Enforce TLS 1.2 (secure ciphers only) and TLS 1.3. Regularly review SSL/TLS configurations Recommended Cipher Examples: TLS_AES_256_GCM_SHA384 TLS_CHACHA20_POLY1305_SHA256 TLS_AES_128_GCM_SHA256
Reference	https://owasp.org/Top10/A02_2021-Cryptographic_Failures/
New/ Repeat Observation	New Observation
References to evidences / Proof of Concept (POCs)	

9. Version Disclosure (jQuery)

Vulnerability Title	***Affected URLs/IP***
Version Disclosure (jQuery)	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Login.aspx
Detailed Observation	it was observed that the application discloses the jQuery library version information within client-side JavaScript files or page source code. Exposing the exact jQuery version allows attackers to fingerprint the application technology stack and identify known vulnerabilities associated with that specific version, which may aid in targeted attacks.
Vulnerability Reference (CWE/CVE)	CWE-200
Severity	**Low**
Recommendation	Avoid exposing exact jQuery version details in production environments. Minify and bundle JavaScript files. Remove version comments and banners from client-side resources. Keep jQuery updated to the latest stable version.
Reference	https://owasp.org/www-community/attacks/Information_Disclosure
New/ Repeat Observation	New Observation
References to evidences / Proof of Concept (POCs)	



10. Concurrent Login

Vulnerability Title	***Affected URLs/IP***
Concurrent Login	https://nagarkaryavaliuat.com/ancl_VisitorMGMT/Lin.aspx
Detailed Observation	The application permits multiple simultaneous sessions for the same user account from different devices/IPs without restriction or session management. This can allow credential sharing, session hijacking persistence, or unauthorized access if credentials are compromised.
Vulnerability Reference (CWE/CVE)	CWE-306
Severity	**Low**
Recommendation	Limit concurrent sessions per account (e.g., 1–3) or allow user-controlled session termination. Invalidate or rotate old sessions when new logins occur (if policy requires single session). Notify users on new-device logins and provide a session management UI.
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/ Repeat Observation	New Observation

References to evidences / Proof of Concept (POCs)