12-Jan-24

Web Application Security Testing Report

of

https://gazeteer.onlinesystems.link/

MAVERICK QUALITY ADVISORY SERVICES PRIVATE LIMITED
123 RADHEY SHYAM PARK SAHIBABAD, GHAZIABAD
UTTAR PRADESH – 201005
TELE FAX: +91- 120-4117140
EMAIL: CONTACT@MQASGLOBAL.COM, INFO@MQASGLOBAL.COM
WEBSITE: WWW.MQASGLOBAL.COM

Disclaimer:

This report and any supplements are **HIGHLY CONFIDENTIAL** and may be protected by one or more legal privileges. It is intended solely for the use of the relevant IT personnel in the organization. This report is prepared based on the IT environment that prevailed in the period of assessment.

This report is not a guarantee or certification that all vulnerabilities have been discovered and reported in the findings. Subsequent reviews may report on previously unidentified findings or on new vulnerabilities. The samples screen shot should not be treated as the final vulnerabilities. Gaps which we have identified can also get replicated in any part of the infrastructure. Organization should ensure that Vulnerability Management Program should be adapted continuously rather than fixing just the issues identified within the areas which MQAS has highlighted.

Issues identified should be closed and resubmitted within one month of publication of the report. Any changes in the build post testing completion shall result in cancellation of “safe to host certificate”.

Document Details:

Item	Description
Document Type	Web Application Security Testing Report
Starting Date	10 July 2023
Document Version	1.3

Authors:

Prepared By		
Name	**Testing Start Date**	**Testing Completion Date**
Rahul Kumar	12 January 2024	12 January 2024
Reviewed & Approved By		
Name	**Approval Date**	
Shivam Singh	12 January 2024	

TABLE OF CONTENTS

1. Introduction

This document summarizes the results of Vulnerability and Penetration tests conducted on the complete web application.

2. Scope

The aim of this project was to conduct the following activities

i. Gather Information
ii. Vulnerabilities Detection
iii. Analysis of Vulnerabilities
iv. Reporting details based on the information gathered

S No.	Application URL
1	https://gazeteer.onlinesystems.link/

3. Methodology

Web Application Security Testing Methodolgy

OUR APPROACH FOR WEB APPLICATION VULNERABILITY ASSESSMENT AND PENETRATION TESTING

STAGE 1: PLANNING AND INFORMATION GATHERING

- Scope to decide potential impact of scanning activities.
- Get the contact details of stake holders that need to be kept posted of the testing activities.
- Share contact details of Team Leads and Project Manager from company.

STAGE 2: VULNERABILITY DETECTION

- WHITE BOX TESTING ☐
- BLACK BOX TESTING ☐
- GREY BOX TESTING ☑

STAGE 3: APPLICATION SECURITY ASSESSMENT

- Assess the security of the selected applications, focusing on remotely exploitable vulnerabilities, application security architecture, design and implementation.
- Assess the controls with respect to user access, privilege levels, development and delivery, and overall design of the applications

STAGE 4: REPORTING AND KNOWLEDGE TRANSFER

- Submit the final and detailed set of reports with in-depth information to fix the vulnerabilities and an efficient and effective follow-up plan
- Conduct a knowledge transfer exercise to the technical team
- Present the findings to the technical and management teams
- Hand over final set of deliverables to the client

4. Standards

OWASP: The OWASP web application security testing methodology and explains how to test for evidence of vulnerabilities within the application due to deficiencies with identified security controls.

The test is divided into 2 phases:

Phase 1 Passive mode:

In the passive mode the tester tries to understand the application's logic and plays with the application.

Phase 2 Active mode:

- Information Gathering
- Configuration and Deployment Management Testing
- Identity Management Testing
- Authentication Testing
- Authorization Testing
- Session Management Testing
- Input Validation Testing
- Error Handling
- Cryptography
- Business Logic Testing
- Client-Side Testing

OWASP TOP 10 Vulnerabilities

- **A01:2021-Broken Access Control:**
  Access control enforces policy such that users cannot act outside of their intended permissions. Failures typically lead to unauthorized information disclosure, modification, or destruction of all data or performing a business function outside the user's limits.

- **A02:2021-Cryptographic Failures:**
  The protection needs of data in transit and at rest. For example, passwords, credit card numbers, health records, personal information, and business secrets require extra protection, mainly if that data falls under privacy laws, e.g., EU's General Data Protection Regulation (GDPR), or regulations, e.g., financial data protection such as PCI Data Security Standard (PCI DSS)

- **A03:2021-Injection:**
  Common injections are SQL, NoSQL, OS command, Object Relational Mapping (ORM), LDAP, and Expression Language (EL) or Object Graph Navigation Library (OGNL) injection. The concept is identical among all interpreters.

- **A04:2021-Insecure Design:**
  A secure design can still have implementation defects leading to vulnerabilities that may be exploited. An insecure design cannot be fixed by a perfect implementation as by definition, needed security controls were never created to defend against specific attacks.

- **A05:2021-Security Misconfiguration:**
  Missing appropriate security hardening across any part of the application stack or improperly configured permissions on cloud services. Unnecessary features are enabled or installed (e.g., unnecessary ports, services, pages, accounts, or privileges). Default accounts and their passwords are still enabled and unchanged. Server does not send security headers or directives, or they are not set to secure values. The software is out of date or vulnerable.

- **A06:2021-Vulnerable and Outdated Components:**
  Versions of all components you use (both client-side and server-side). This includes components you directly use as well as nested dependencies. the software is vulnerable, unsupported, or out of date. This includes the OS, web/application server, database management system (DBMS), applications, APIs and all components, runtime environments, and libraries.

- **A07:2021-Identification and Authentication Failures:**
  Confirmation of the user's identity, authentication, and session management is critical to protect against authentication-related attacks. Using weak or ineffective credential recovery and forgot-password processes, such as "knowledge-based answers". Permitting brute force or other automated attacks. Permitting default, weak, or well-known passwords, such as "Password1" or "admin/admin".

- **A08:2021-Software and Data Integrity Failures:**
  Software and data integrity failures relate to code and infrastructure that does not protect against integrity violations. An example of this is where an application relies upon plugins, libraries, or modules from untrusted sources, repositories, and content delivery networks (CDNs). An insecure CI/CD pipeline can introduce the potential for unauthorized access, malicious code, or system compromise.

- **A09:2021-Security Logging and Monitoring Failures:**
  This category is to help detect, escalate, and respond to active breaches. Without logging and monitoring, breaches cannot be detected. Insufficient logging, detection, monitoring, and active response occurs any time: Appropriate alerting thresholds and response escalation processes are not in place or effective, Auditable events, such as logins, failed logins, and high-value transactions, are not logged. Warnings and errors generate no, inadequate, or unclear log messages. Logs of applications and APIs are not monitored for suspicious activity.

- **A10:2021-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).

5. Ratings

The table below gives a key to the risk naming and colours used throughout this report to provide a clear and concise risk scoring system.

It should be noted that quantifying the overall business risk posed by any of the issues found in any test is outside our scope. This means that some risks may be reported as high from a technical perspective but may, as a result of other controls unknown to us, be considered acceptable by the business.

S No.	Risk Rating	Description
1	**Critical**	A vulnerability was discovered that has been rated as critical. This requires resolution as quickly as possible.
2	**High**	A vulnerability was discovered that has been rated as high. This requires resolution in a short term.
3	**Medium**	A vulnerability was discovered that has been rated as medium. This should be resolved throughout the ongoing maintenance process.
4	**Low**	A vulnerability was discovered that has been rated as low. This should be addressed as part of routine maintenance tasks.

6. Findings Summary

All the issues identified during the assessment are listed below with a brief description and risk rating for each issue. The risk ratings used in this report are defined in Risk Ratings Section.

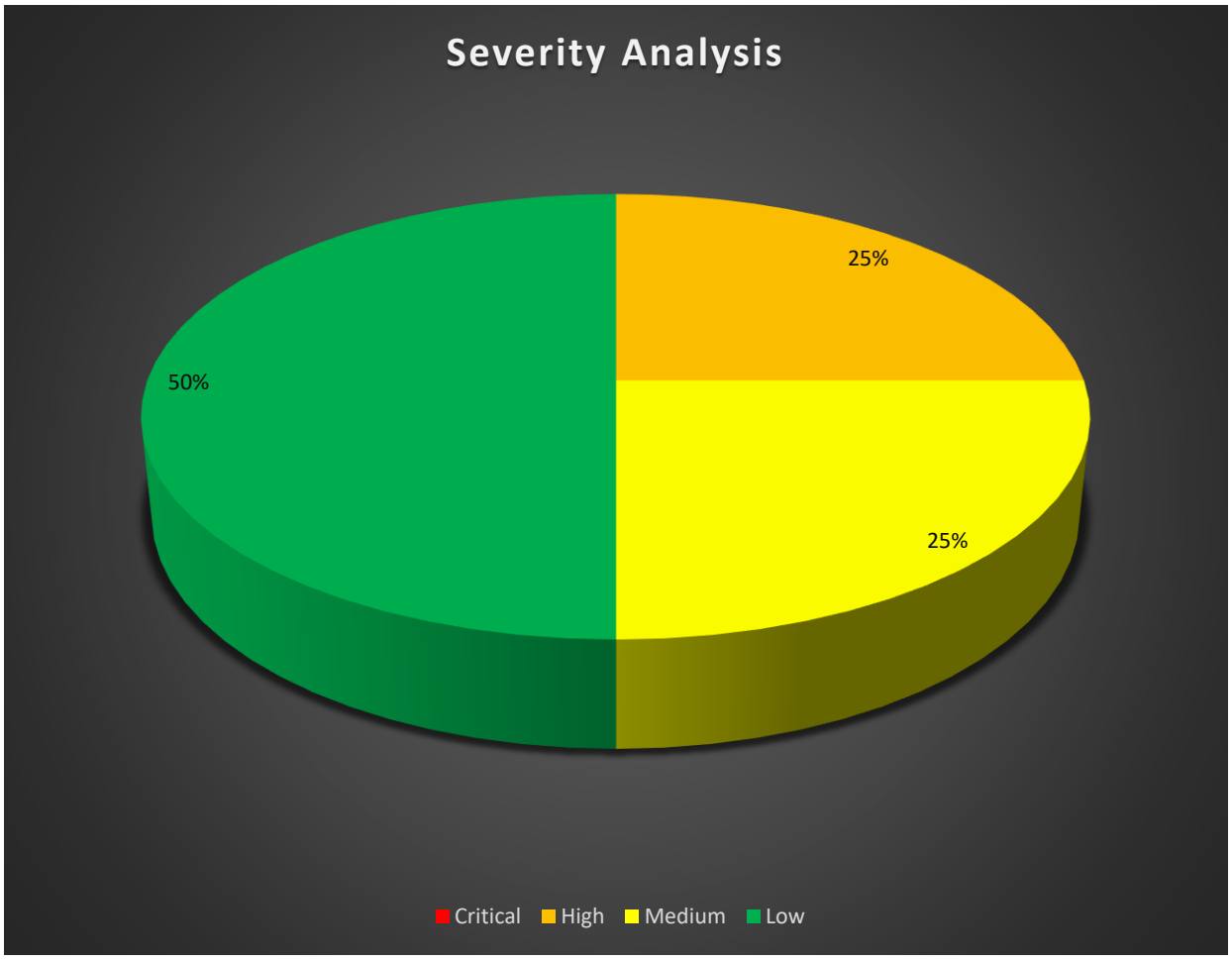
URL	Vulnerabilities	Severity	Status
https://gazeteer.onlinesystems.link/	Broken Access Control	**High**	**Open**
https://gazeteer.onlinesystems.link/	Anonymous/Weak Ciphers Supported	**Medium**	Closed*
https://gazeteer.onlinesystems.link/	Unsupported Version (Bootstrap)	**Medium**	Closed*
https://gazeteer.onlinesystems.link/	No Rate Limiting	**Medium**	**Open**
https://gazeteer.onlinesystems.link/	Out-of-date Version (jQuery)	**Low**	Closed*
https://gazeteer.onlinesystems.link/	Out-of-date Version (Bootstrap)	**Low**	Closed
https://gazeteer.onlinesystems.link/	Server Banner Grabbing	**Low**	**Open**
https://gazeteer.onlinesystems.link/	Server Leaks Information via "X-Powered-By"	**Low**	**Open**
https://gazeteer.onlinesystems.link/	Cookies Without SameSite Flag Detected	**Low**	Closed
https://gazeteer.onlinesystems.link/	Cookie Not Flagged ‘Secure’	**Low**	Closed
https://gazeteer.onlinesystems.link/	Cookie Without HttpOnly Flag Set	**Low**	Closed

*As Per Client

6.1 Overall Finding Summary

Severity Level	Counts of Vulnerability
Critical Level	0
High Level	1
Medium Level	1
Low Level	2
Total	**4**

6.2 Overall Severity



7. Detailed Findings

7.1 Broken Access Control

Affected URL : https://gazeteer.onlinesystems.link/app/writereaddata/uploadedCirculars/C_20230710151645894 1.pdf
https://gazeteer.onlinesystems.link/app/writereaddata/UploadedPressRelease/pdf/C_20230803121326414 8.jpg
https://gazeteer.onlinesystems.link/app/writereaddata/uploadedCirculars/Corrigendum/pdf/C_20230803122118610 7.pdf

Severity : **High**

CVE/CWE ID : CWE-284

Description : Application does not restrict or incorrectly restricts access to a resource from an unauthorized user.

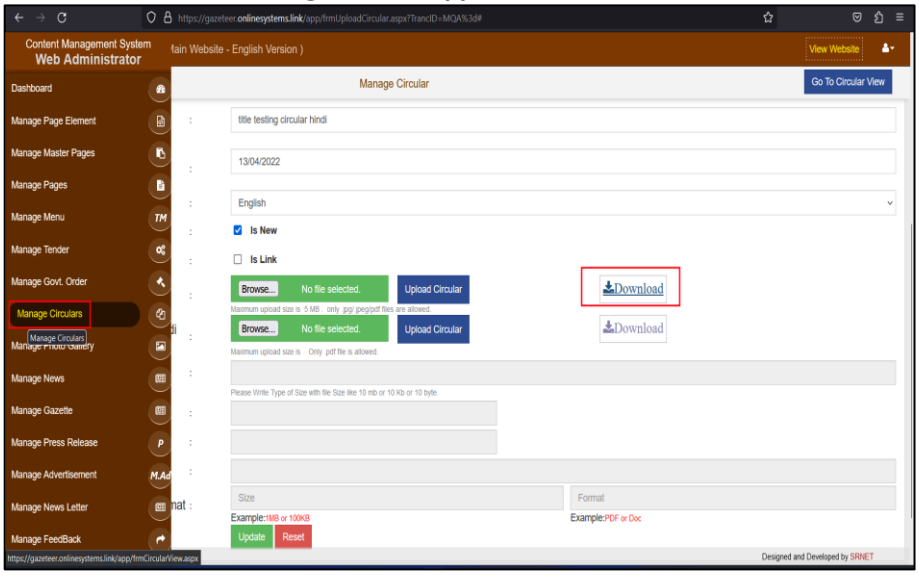
Impact : Anonymous user can access private functions that aren't protected by just copying the URL.

Recommendation :

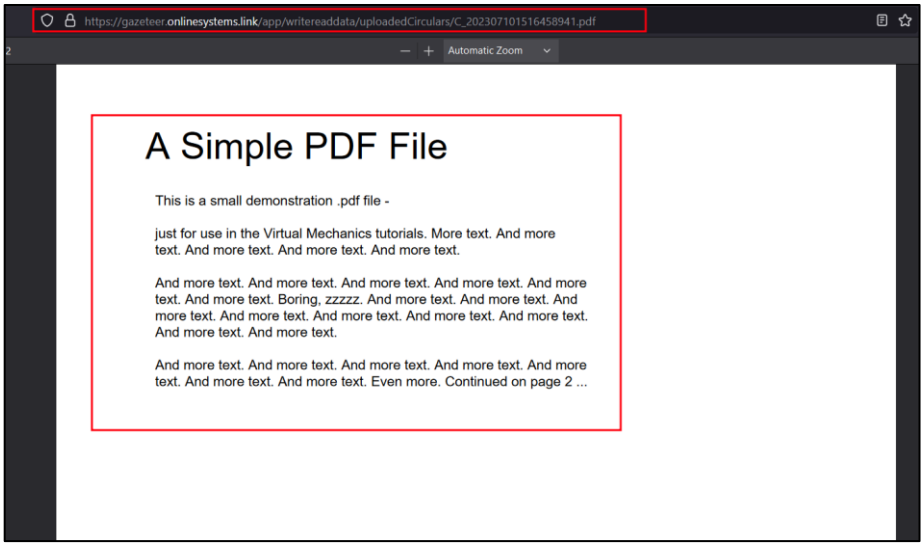
- The authentication mechanism should deny all access by default and provide access to specific roles for every function.
- In a workflow-based application, verify the users' state before allowing them to access any resources.

Note: Fix this issue throughout the application.

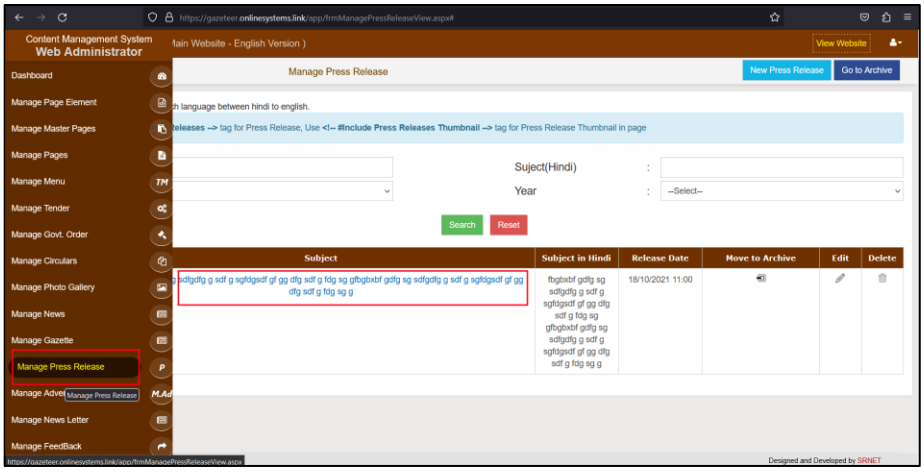
Reference :



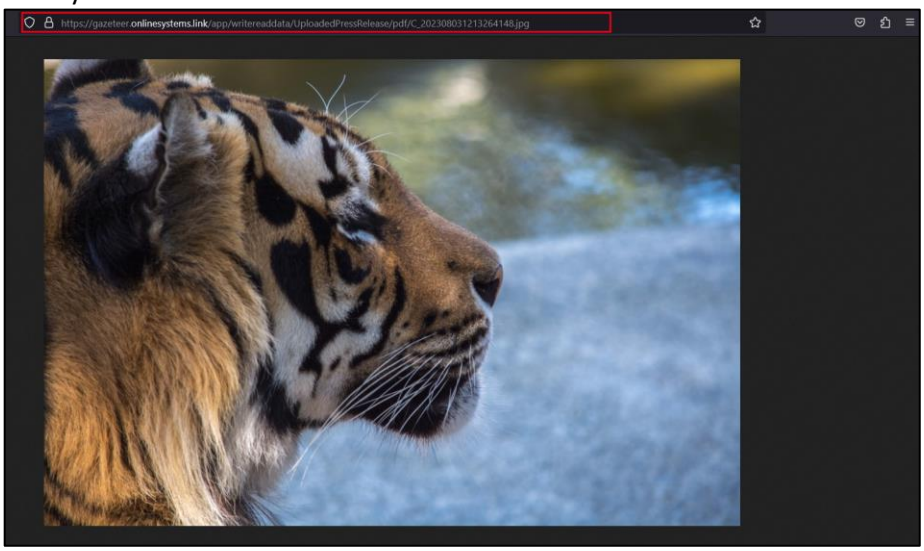
Click on download, the file will open in new tab. Copy Url and paste it on any browser.

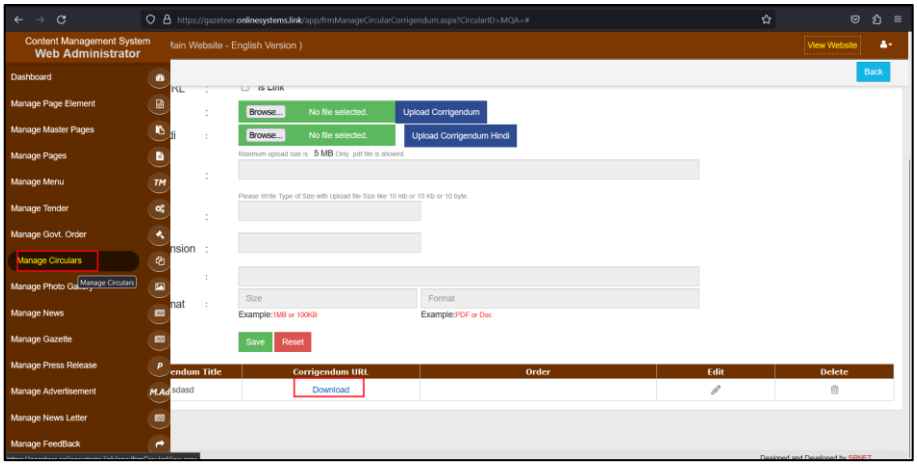


Reference (R2) :

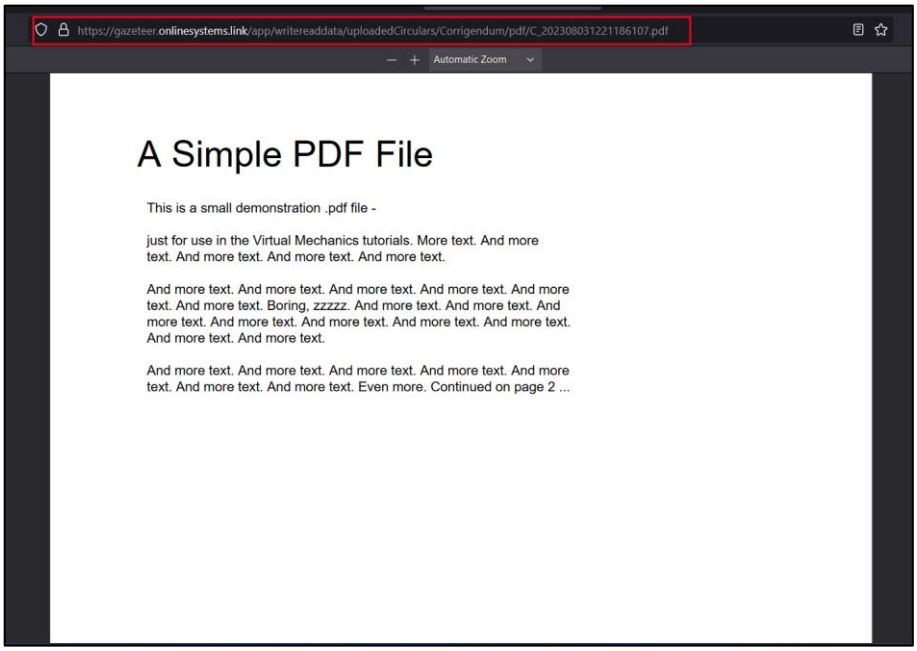


Click on highlighted url, the file will open in new tab. Copy Url and paste it on any browser.

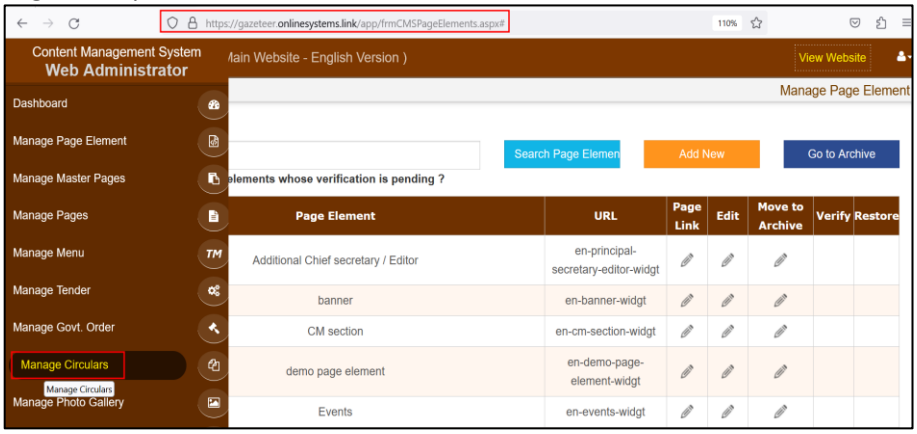


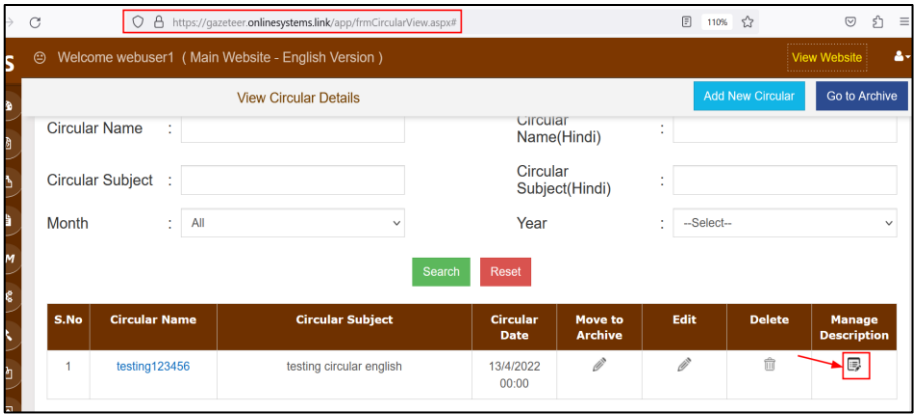


Click on Download, the file will open in new tab. Copy Url and paste it on any browser.

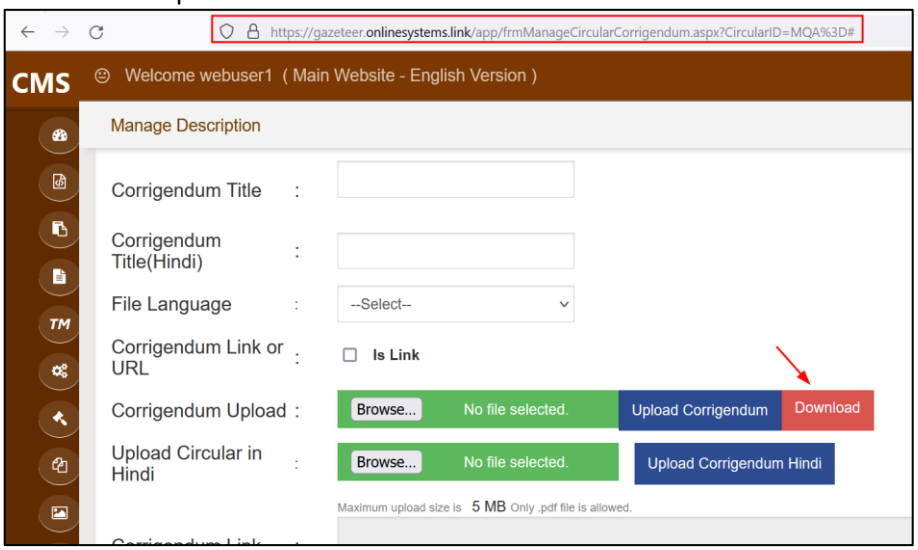


Reference(R3) : Login with provided credentials.

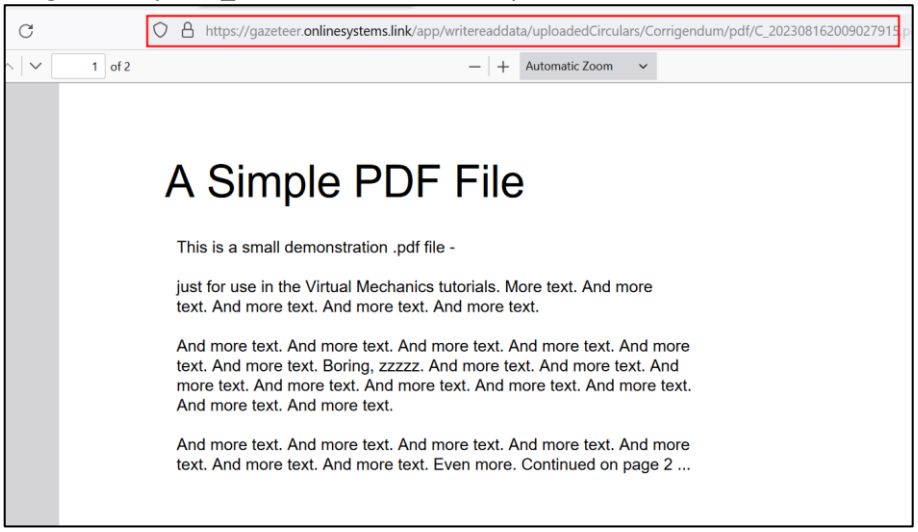




First browse a pdf file then click on download.

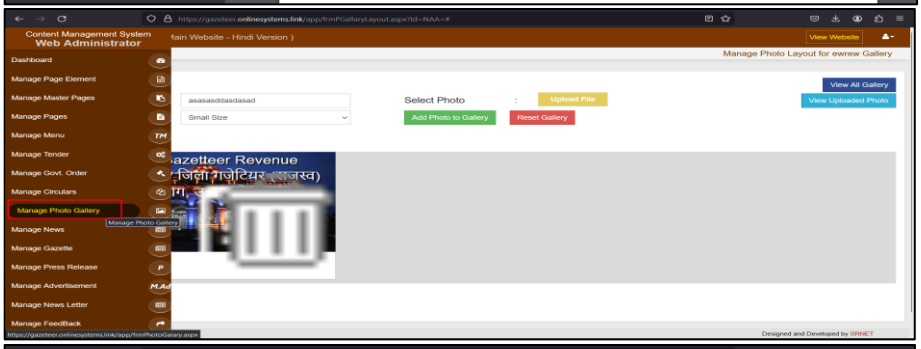


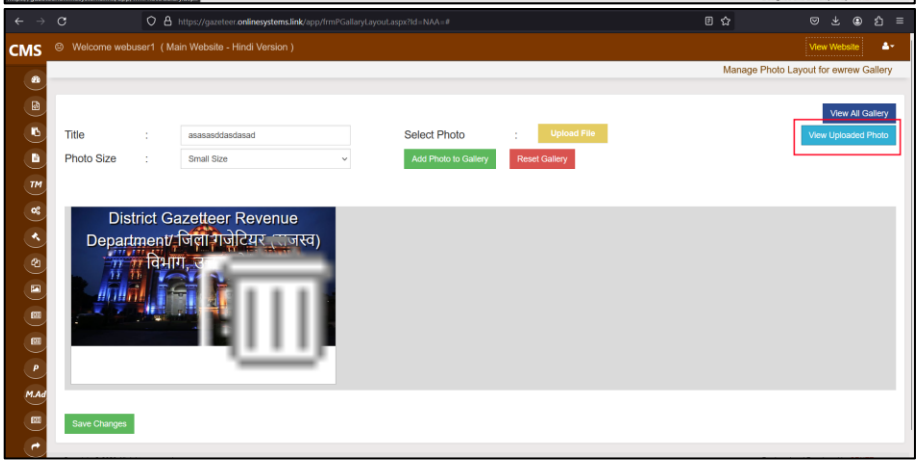
It will open the uploaded pdf file in new tab.Copy this URL https://gazeteer.onlinesystems.link/app/writereaddata/uploadedCirculars/Corrigendum/pdf/C_202308162009027915.pdf



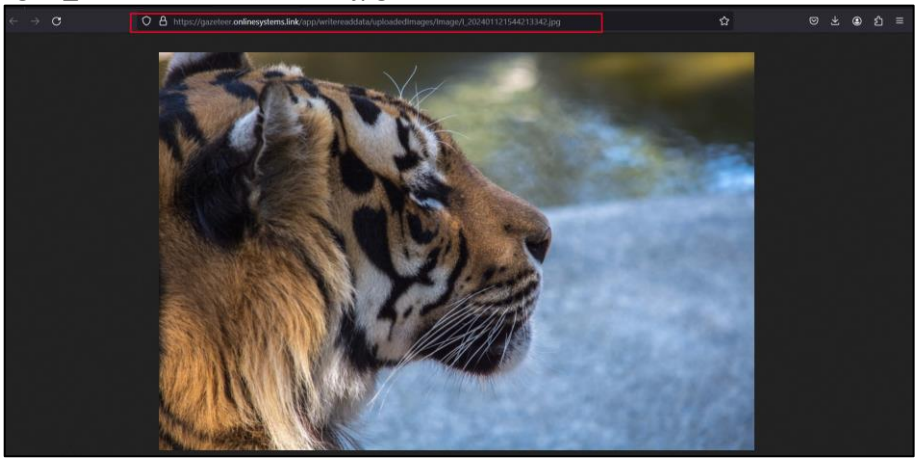
Now paste the copied URL in any browser and we can see the uploaded file directly without any authentication.

Reference(R4) :





https://gazeteer.onlinesystems.link/app/writereaddata/uploadedImages/Image/I_202401121544213342.jpg



Copy URL and paste it on any browser. Able to access image without authentication.

7.2 Anonymous/Weak Ciphers Supported

Affected URL : https://gazeteer.onlinesystems.link

Severity : **Medium**

CVE/CWE ID : CVE-2007-1858

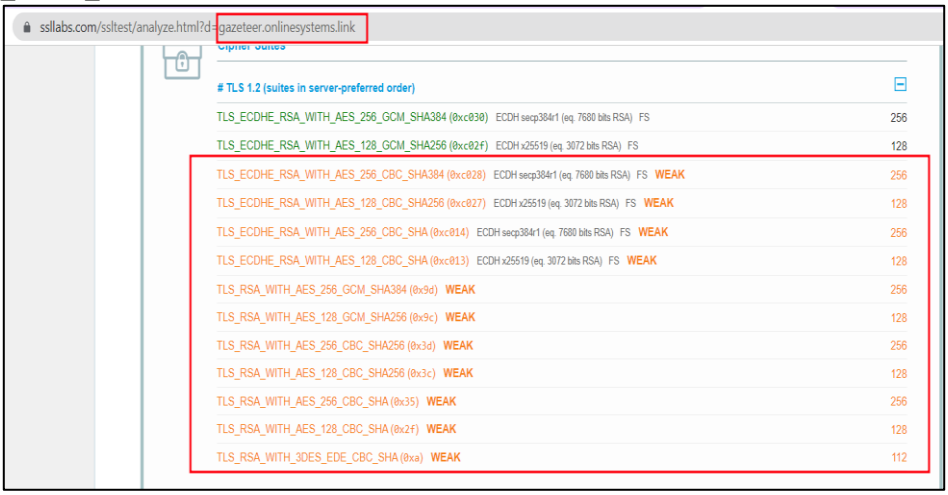
Description : Identified that anonymous ciphers is supported during secure communication (SSL). You should allow only strong ciphers on your web server to protect your secure communication with your visitors.

Impact : Attackers can perform man-in-the middle attacks and observe the encrypted traffic between your website and your visitors.

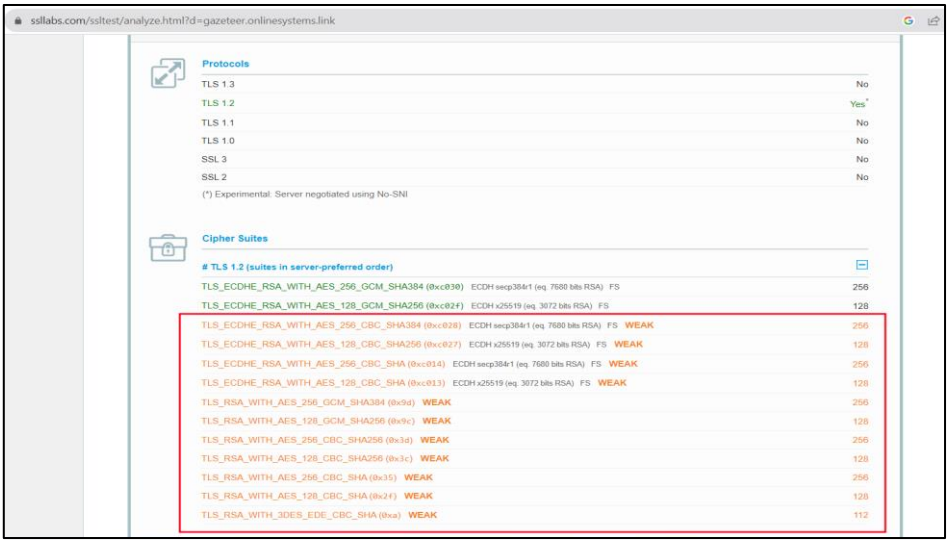
Recommendation : Configure your web server to disallow using anonymous ciphers.
For Microsoft IIS, you should make some changes to the system registry. Incorrectly editing the registry may severely damage your system. Before making changes to the registry, you should back up any valued data on your computer.
a. Click Start, click Run, type regedt32 or type regedit, and then click OK.
b. In Registry Editor, locate the following registry key: HKLM\SYSTEM\CurrentControlSet\Control\SecurityProviders
c. Set "Enabled" DWORD to "0x0" for the following registry keys:
SCHANNEL\Ciphers\DES 56/56
SCHANNEL\Ciphers\RC4 64/128
SCHANNEL\Ciphers\RC4 40/128
SCHANNEL\Ciphers\RC2 56/128
SCHANNEL\Ciphers\RC2 40/128
SCHANNEL\Ciphers\NULL
SCHANNEL\Hashes\MD5

https://cheatsheetseries.owasp.org/cheatsheets/Transport_Layer_Protection_Cheat_Sheet.html

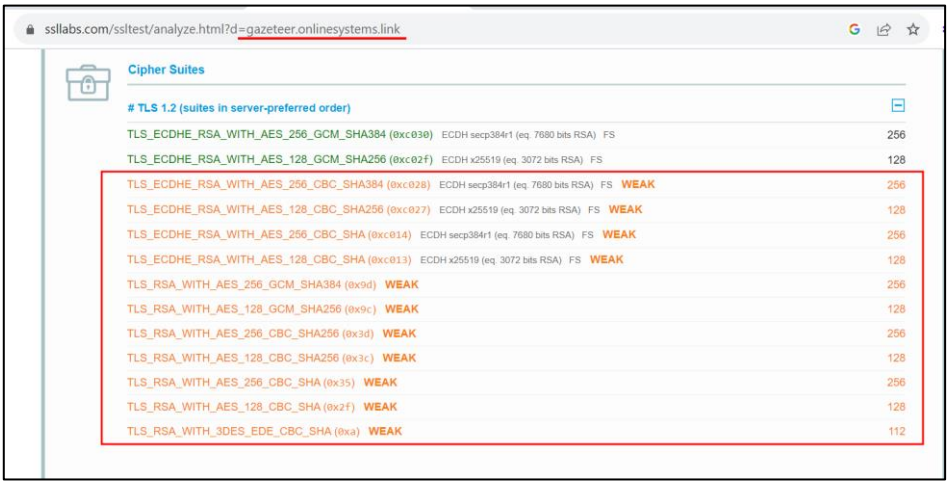
Reference :



Reference (R2) :



Reference(R3) :



Reference(R4) : Closed*

7.3 Unsupported Version (Bootstrap)

Affected URL : https://gazeteer.onlinesystems.link/js/bootstrap.min.js

Severity : **Medium**

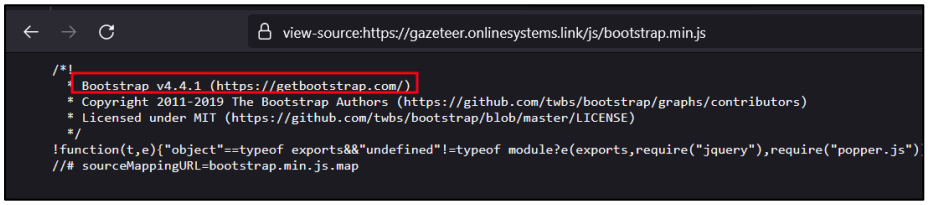
CVE/CWE ID : CVE-2021-23472

Description : Bootstrap is a free and open-source CSS framework directed at responsive, front-end web development. It contains CSS and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

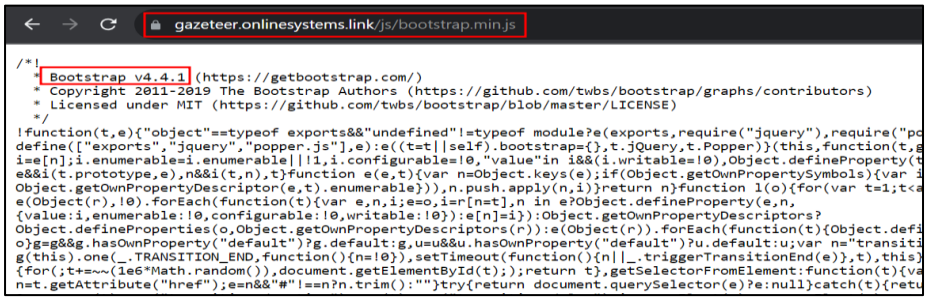
Impact : Impact may be varying according to the vulnerability present on currently installed Bootstrap version.

Recommendation : Upgrade to the latest version of the Bootstrap.

Reference (R2) :



Reference(R3) :



Reference(R4) : Closed*

7.4 No Rate Limiting

Affected URL : https://gazeteer.onlinesystems.link/en/feedback?cd=NgAzAA%3D%3D
https://gazeteer.onlinesystems.link/app/frmForgetPwd.aspx

Severity : **Medium**

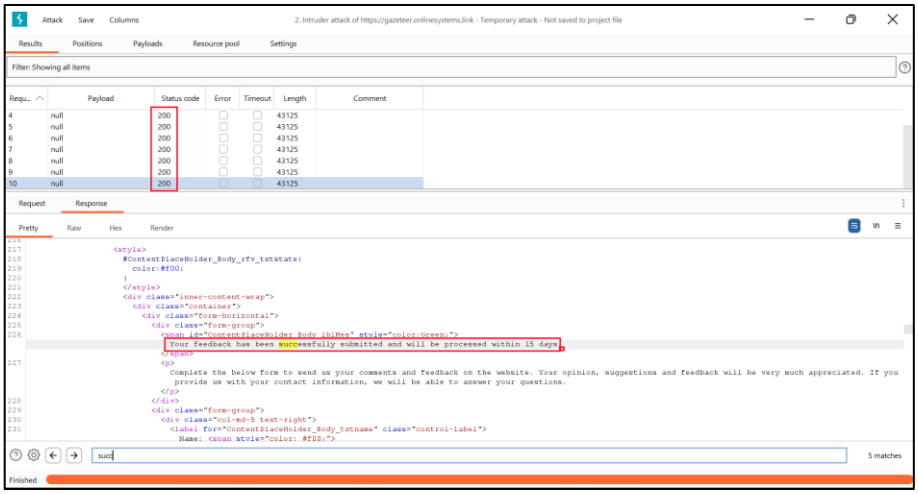
CVE/CWE ID : CWE-770

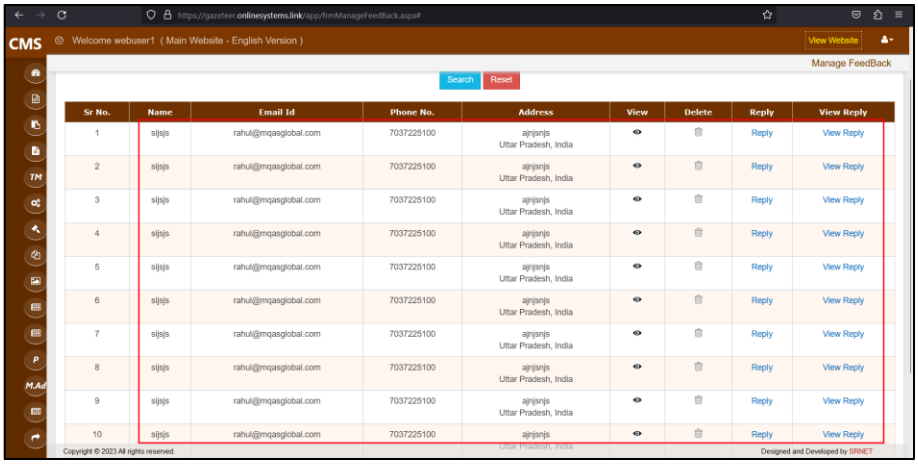
Description : Rate limiting is used to control the rate of traffic sent or received by a network interface controller and is used to prevent DoS attacks.

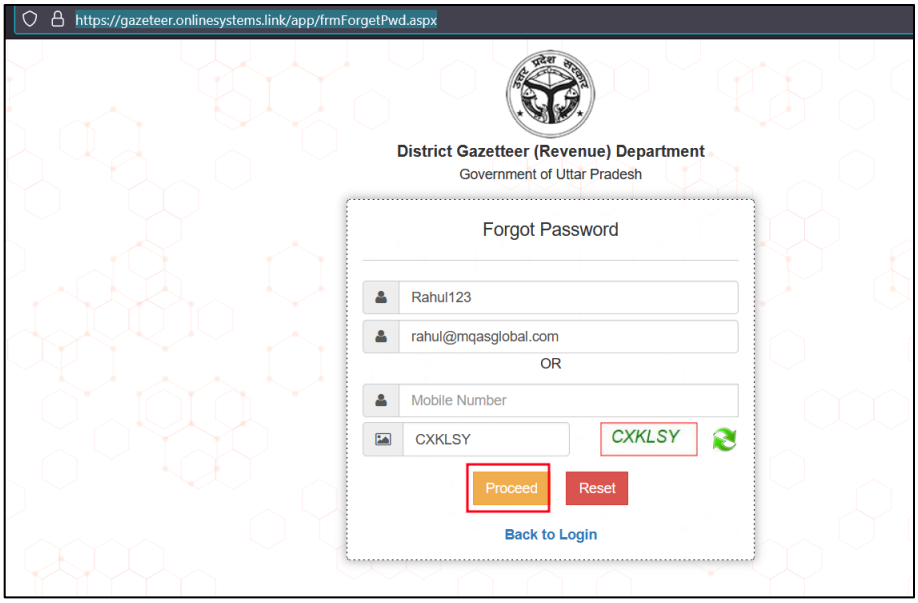
Impact : No rate limit means there is no mechanism to protect against the requests you made in a short frame of time.

Recommendation : Limit the rate for reset links/opt to avoid such kind of issues.
Note: Fix this issue throughout the application.

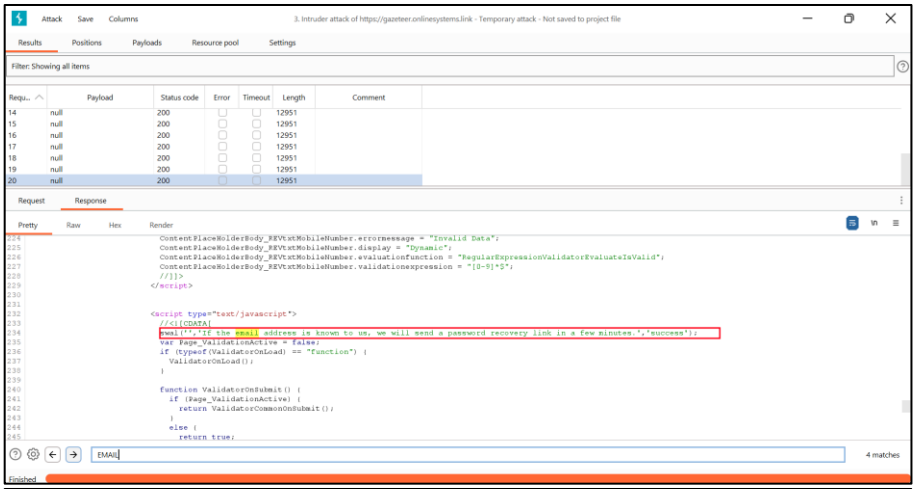
Reference (R2) :

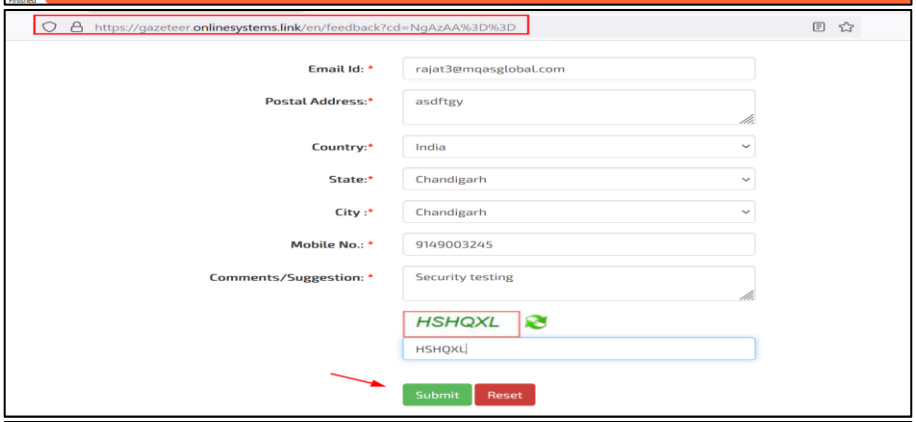


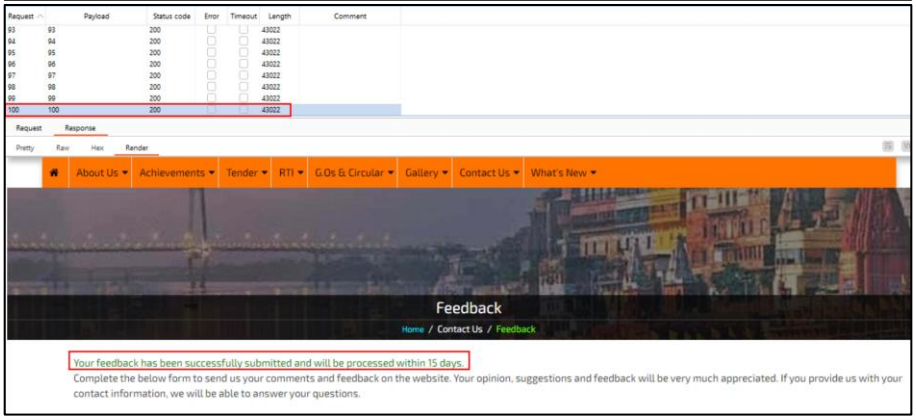


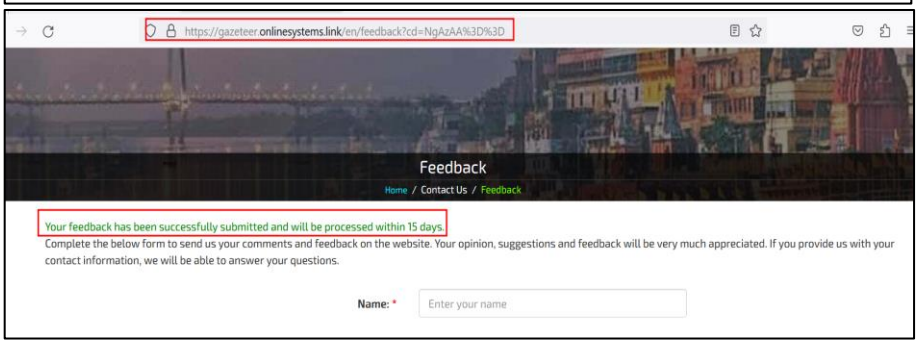


Reference(R3) :









Reference(R4) : Feedback form is not working properly. Not able to verify bugs.
Open

7.5 Out-of-date Version (jQuery)

Affected URL : https://gazeteer.onlinesystems.link/

Severity : **Low**

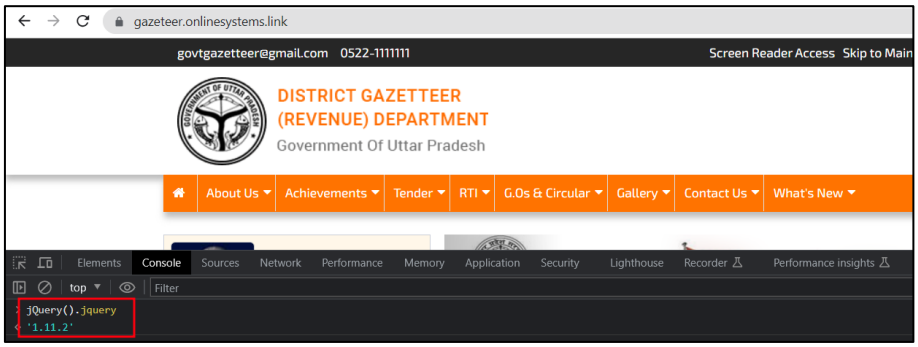
CVE/CWE ID : CWE-672

Description : jQuery is a package that makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.

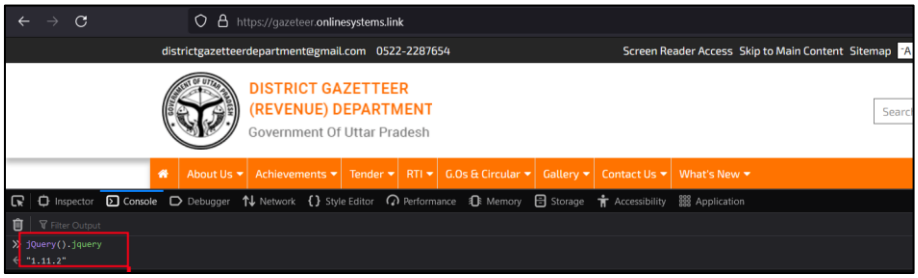
Impact : Affected version is vulnerable to Cross-site Scripting (XSS) attacks when a cross-domain Ajax request is performed without the datatype option, causing text/JavaScript responses to be executed.

Recommendation : Upgrade jQuery to latest stable version.

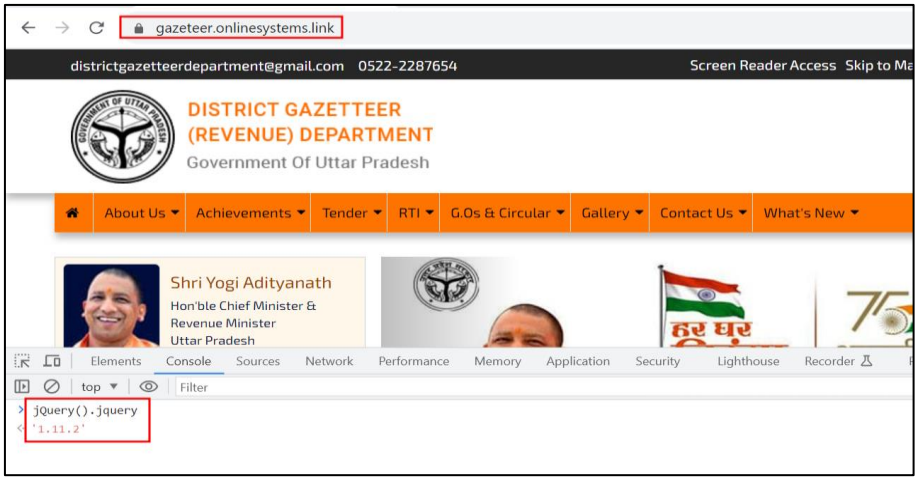
Reference :



Reference (R2) :



Reference(R3) :



Reference(R4) : Closed*

7.6 Out-of-date Version (Bootstrap)

Affected URL : https://gazeteer.onlinesystems.link/js/bootstrap.min.js

Severity : **Low**

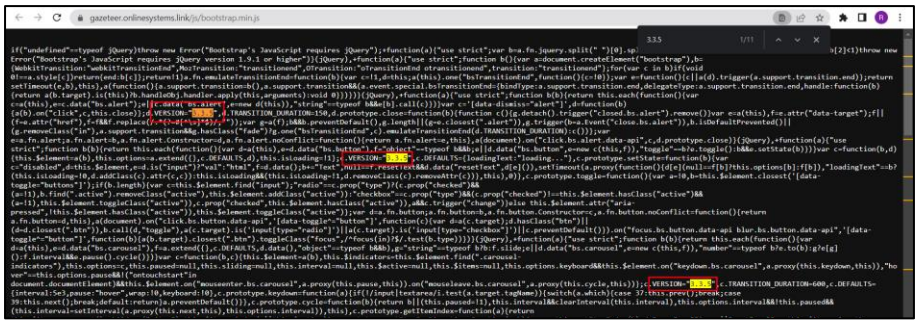
CVE/CWE ID : CWE-672, CWE-937

Description : Bootstrap is a free and open-source CSS framework directed at responsive, front-end web development. It contains CSS and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.

Impact : Impact may be varying according to the vulnerability present on currently installed Bootstrap version.

Recommendation : Upgrade to the latest version of the Bootstrap.

Reference :



7.7 Server Banner Grabbing

Affected URL : https://gazeteer.onlinesystems.link

Severity : **Low**

CVE/CWE ID : CWE-200

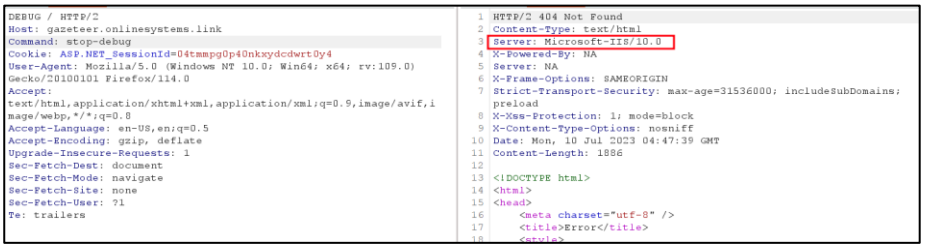
Description : Banner grabbing is a technique used to remotely glean information about the web server (version, underlying OS, vendor etc.).

Impact : By identifying the HTTP server, the attacker can determine known vulnerabilities related to the web server and the appropriate exploits for the same.

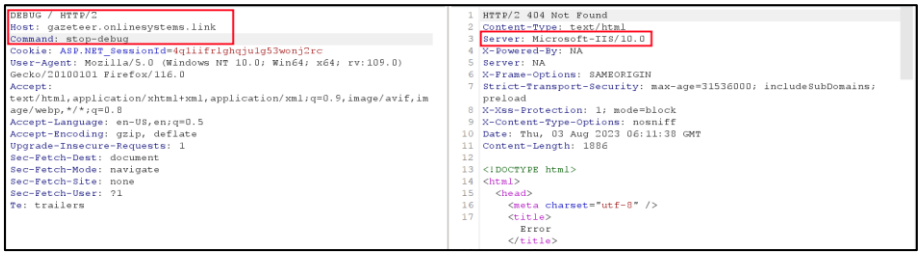
Recommendation : It is recommended to disable the banner and upgrade the latest version. The web server should be hardened according to industry best practices and configured appropriately

Reference :

1. Intercept the response from the server using intercepting proxy tool like Burp.
2. Notice the Server name is observed in the Response.



Reference (R2) :



Reference(R3) :

Reference(R4) :

7.8 Server Leaks Information via "X-Powered-By"

Affected URL : https://gazeteer.onlinesystems.link

Severity : **Low**

CVE/CWE ID : CWE-200

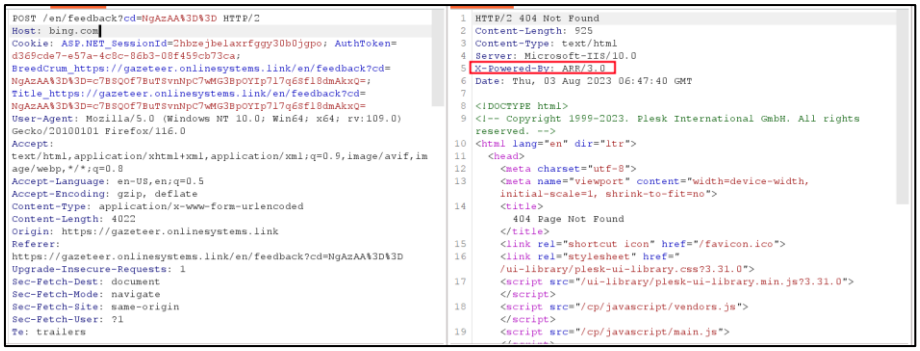
Description : The most basic form of identifying a web framework is to look at the X-Powered-By field in the HTTP response header. Many tools can be used to fingerprint a target.

Impact : The web/application server is leaking information via one or more "X-Powered-By" HTTP response headers. Access to such information may facilitate attackers identifying other frameworks/components your web application is reliant upon and the vulnerabilities such components may be subject to.

Recommendation : Ensure that your web server, application server, load balancer, etc. is configured to suppress "X-Powered-By" headers.

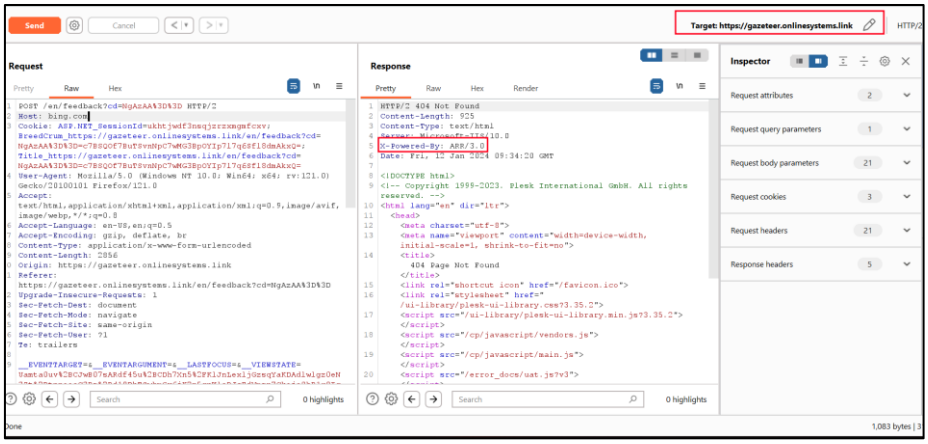
Reference :

Reference (R2) :



Reference(R3) :

Reference(R4) :



7.9 Cookies Without SameSite Flag Detected

Affected URL : https://gazeteer.onlinesystems.link/app/frmDashboardDetails.aspx#

Severity : **Low**

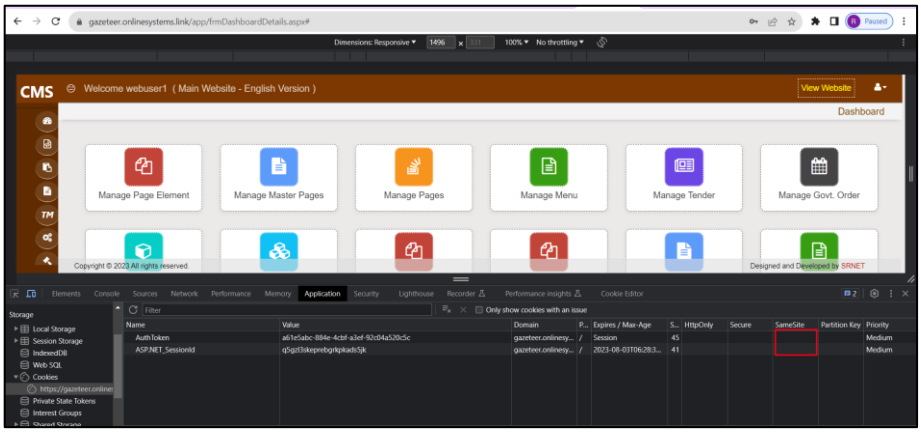
CVE/CWE ID : CWE-614

Description : The SameSite flag is not set on a cookie.

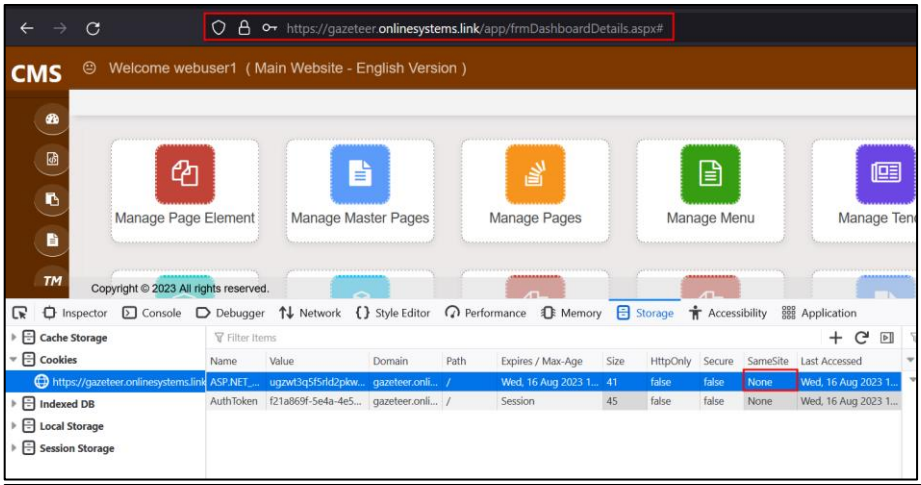
Impact : The browser will prevent it from being sent along with cross-site requests. This can help prevent Cross-Site Request Forgery (CSRF) attacks.

Recommendation : If the cookie contains sensitive information, then the server should ensure that the cookie has the SameSite flag set. This flag can have two values: strict or lax. With the strict value the cookie will only be sent if the request originates from the same website. With the lax value the cookie will only be sent for GET requests.

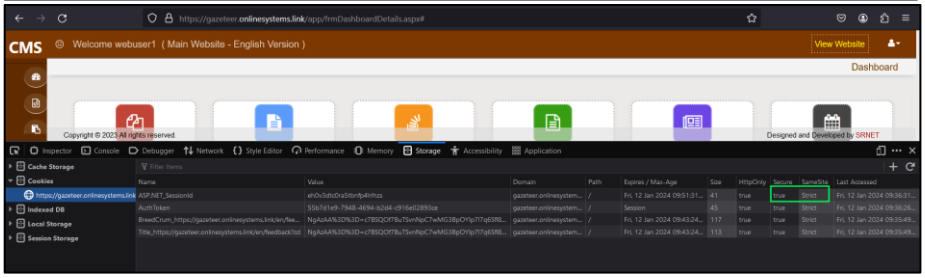
Reference (R2) :



Reference(R3) :



Reference(R4) :



7.10 Cookie Not Flagged 'Secure'

Affected URL : https://gazeteer.onlinesystems.link/app/frmDashboardDetails.aspx#

Severity : **Low**

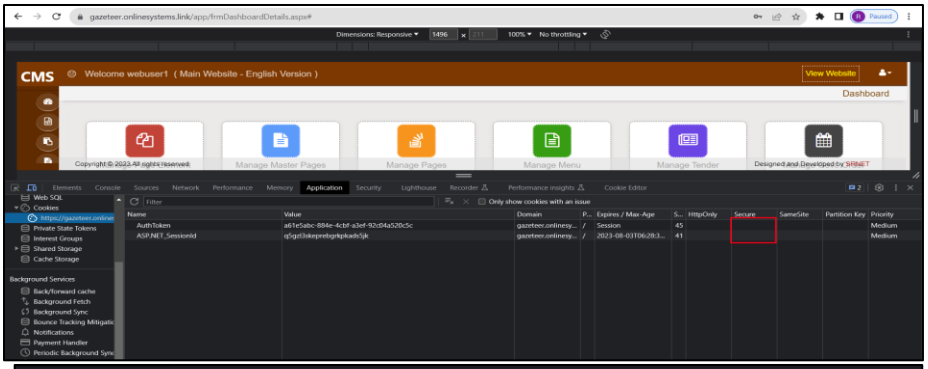
CVE/CWE ID : CWE-614

Description : It is found that cookie which passes sensitive information is not flagged as 'Secure'.
This cookie does not have the Secure flag set. When a cookie is set with the Secure flag, it instructs the browser that the cookie can only be accessed over secure SSL channels. This is an important security protection for session cookies.

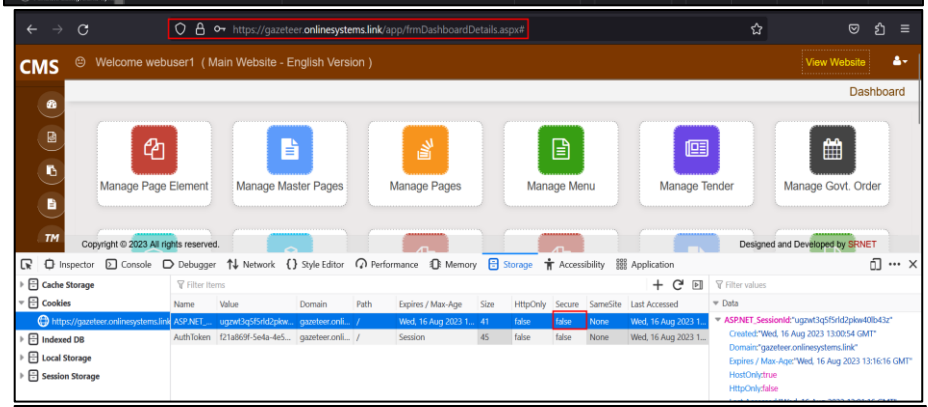
Impact : The hacker can observe the cookie if it is not set to 'Secure'.

Recommendation : It is strongly recommended to mark the cookie as 'Secure'.

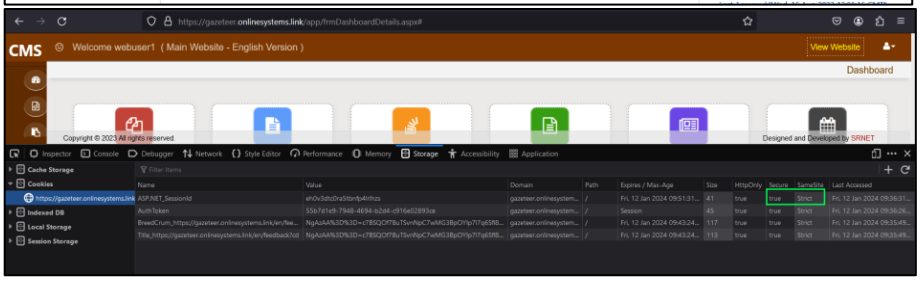
Reference (R2) :
1. Access the URL and capture the request via proxy or analyze cookie using browser tools.
2. Secure flag attribute is missing in cookie.



Reference(R3) :



Reference(R4) :



7.11 Cookie Without HttpOnly Flag Set

Affected URL : https://gazeteer.onlinesystems.link/app/frmDashboardDetails.aspx#

Severity : **Low**

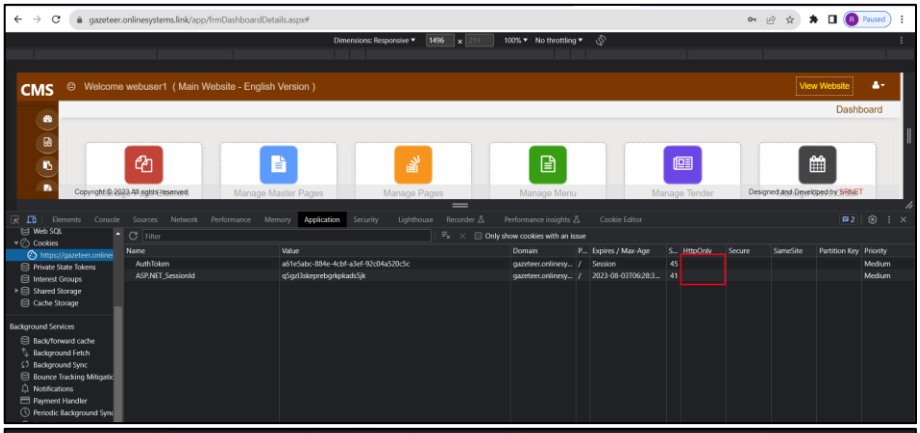
CVE/CWE ID : CWE-614

Description : This cookie does not have the HTTPOnly flag set. The HttpOnly flag directs compatible browsers to prevent client-side script from accessing cookies. Including the HttpOnly flag in the Set-Cookie HTTP response header helps mitigate the risk associated with Cross-Site Scripting (XSS) where an attacker's script code might attempt to read the contents of a cookie and exfiltrate information obtained. When set, browsers that support the flag will not reveal the contents of the cookie to a third party via client-side script executed via XSS.

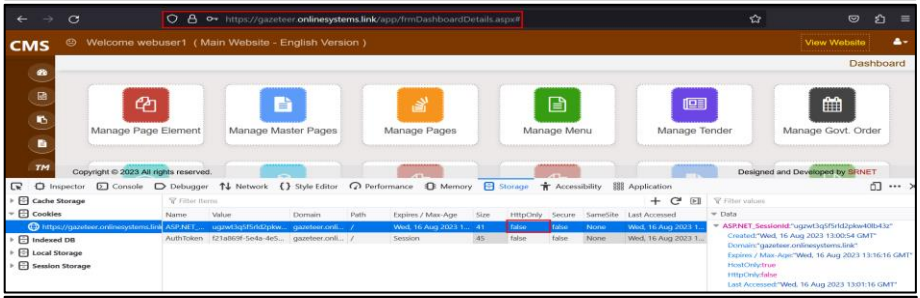
Impact :
- Technical Impact Confidentiality: Read Application Data If the HttpOnly flag is not set, then sensitive information stored in the cookie may be exposed to unintended parties.
- Technical Impact Integrity: Gain Privileges or Assume Identity. If the cookie in question is an authentication cookie, then not setting the HttpOnly flag may allow an adversary to steal authentication data (e.g., a session ID) and assume the identity of the user.

Recommendation : There is usually no good reason not to set the HttpOnly flag on all cookies. Unless you specifically require legitimate client-side scripts within your application to read or set a cookie's value, you should set the HttpOnly flag by including this attribute within the relevant Set-cookie directive.
Numerous other serious attacks can be delivered by client-side script injection, aside from simple cookie stealing.

Reference (R2) :
1. Access the URL and capture the request via proxy or analyze cookie using browser tools.
2. HttpOnly flag attribute is missing in cookie.



Reference(R3) :



Reference (R4) :