

DR. VISHWANATH KARAD MIT WORLD PEACE
UNIVERSITY, PUNE

Vulnerability Identification and Penetration Testing
Third Year B. Tech, Semester 6

VULNERABILITY AND PENETRATION TESTING ON
APPOINTMENT ASSISTANT SYSTEM

VIPT MINI PROJECT REPORT

Under the Guidance of
Dr. Dhanashri Wategaonkar

Prepared By

Krishnaraj Thadesar, PA10, 1032210888

Parth Zarekar, PA07, 1032210846

Sourab Karad, PA25, 10332211150

Saubhagya Singh, PA24, 1032211144

Department of School of Computer Engineering and
Technology
Maharashtra, India.

2023-2024

April 22, 2024

Contents

1 Document Authorities	4
2 Executive Summary	5
2.1 Scope of Work	5
2.2 Project Objectives	6
2.3 Assumptions	6
2.4 Timeline	6
2.5 summary of Findings	7
2.5.1 Cloud Metadata Potentially Exposed	7
2.5.2 Content Security Policy (CSP) Header Not Set	8
2.5.3 Missing Anti-clickjacking Header	9
2.5.4 Cross-Domain Misconfiguration	10
2.5.5 Strict-Transport-Security Header Not Set	12
2.5.6 X-Content-Type-Options Header Missing	12
2.5.7 Burp Suit Report	13
2.6 summary of Recommendations	14
3 Methodology	16
3.1 Planning	16
3.2 Exploitation	16
3.3 Reporting	16
4 Detail Findings	18
4.1 Windows Server Information	19
Bibliography	20

List of Figures

2.1	Header and Body Script	7
2.2	Cloud Metadata Potentially Exposed	8
2.3	Header and Body Script	9
2.4	Content Security Policy (CSP) Header Not Set	9
2.5	Header and Body Script	10
2.6	Missing Anti-clickjacking Header	10
2.7	Header and Body Script	11
2.8	Cross-Domain Misconfiguration	11
2.9	Header and Body Script	12
2.10	Strict-Transport-Security Header Not Set	12
2.11	Header and Body Script	13
2.12	X-Content-Type-Options Header Missing	13
2.13	Header and Body Script	14
4.1	Windows Server Information	19

List of Tables

1.1 Document Authorities	4
------------------------------------	---

Chapter 1

Document Authorities

Company	Clarity Boys
Document Title	Appointment Assistant Vulnerability Assessment Penetration Testing
Date	17/04/2024
Prepared By	Krishnaraj Thadesar, Parth Zarekar, Sourab Karad, Saubhagya Singh
Scope	Application Security Assessment

Table 1.1: Document Authorities

Chapter 2

Executive Summary

2.1 Scope of Work

The vulnerability assessment was conducted using OWASP Zed Attack Proxy (ZAP) to test various components of the appointment assistant website. The scope of the assessment included but was not limited to the following areas:

- **1. Login Functionality**
Authentication mechanisms, including username/password validation and session management.
Potential vulnerabilities such as brute force attacks, session fixation, and authentication bypass.
- **2. Complete Site**
All functionalities accessible to authenticated users, including booking appointments with teachers, administrators, and other personnel.
Input validation for user-submitted data in forms and interactions with the database.
Client-side and server-side processing of user inputs.
- **3. Password Management**
Password storage mechanisms, such as encryption and hashing.
Password reset functionality and security controls to prevent unauthorized access to user accounts.
- **Sensitive Data Handling**
Protection of sensitive data, such as personal information, payment details, and communication logs.
Encryption in transit (HTTPS) and at rest (database encryption).
- **5. Security Headers**
API Endpoints Security of API endpoints used for data exchange with external systems or mobile applications.
Authorization checks and input validation for API requests.
- **6. Security Headers** Presence and effectiveness of security headers, such as Content Security Policy (CSP), Strict-Transport-Security (HSTS), and X-Frame-Options.
- **7. Error Handling** Robustness of error handling mechanisms to prevent information disclosure and potential exploitation.
- **8. Third-party Integrations** Security considerations for third-party libraries, plugins, and integrations used within the website.
- **9. Session Management** Secure handling of user sessions, including session expiration, cookie security, and protection against session hijacking.
- **10. Access Controls** Role-based access controls (RBAC) and permissions management to ensure appropriate access levels for different user roles.

2.2 Project Objectives

The primary goals of the vulnerability assessment are as follows:

- **1. Identify Security Weaknesses:** Conduct a comprehensive evaluation of the appointment assistant website to identify potential vulnerabilities and security weaknesses that could be exploited by malicious actors.
- **2. Assess Risks:** Evaluate the severity and potential impact of identified vulnerabilities on the confidentiality, integrity, and availability of the website and its data. Classify vulnerabilities based on their risk levels.
- **3. Mitigate Security Risks:** Provide actionable recommendations and remediation strategies to mitigate identified security risks and vulnerabilities effectively.
- **4. Enhance Security Posture:** Improve the overall security posture of the appointment assistant website by addressing vulnerabilities, implementing security best practices, and enhancing security controls.
- **5. Compliance:** Ensure compliance with relevant security standards, regulations, and industry best practices, such as OWASP Top 10, PCI DSS, and GDPR, where applicable.

2.3 Assumptions

- **System Patching:** It is assumed that the appointment assistant website's underlying operating systems, web servers, database servers, and third-party software components are kept up-to-date with the latest security patches and updates. Vulnerability assessment results may vary if systems are not patched regularly.
- **Secure Configuration:** The assessment assumes that the configuration of the web server, application server, database server, and related components follows industry best practices for security, such as disabling unnecessary services, implementing access controls, and enabling security features like firewalls and intrusion detection systems.
- **Secure Development Practices:** It is assumed that the appointment assistant website has been developed using secure coding practices, including input validation, output encoding, parameterized queries to prevent SQL injection, and secure authentication mechanisms.
- **Access Controls:** The assessment assumes that appropriate access controls and permissions are implemented within the website to restrict unauthorized access to sensitive data and functionalities based on user roles and responsibilities.
- **Network Security:** It is assumed that the network infrastructure supporting the appointment assistant website, including routers, firewalls, and network segmentation, is configured securely to protect against external threats, such as denial-of-service (DoS) attacks and network intrusions.

2.4 Timeline

The vulnerability assessment project for the appointment assistant website commenced on the 17th of the month and concluded on the 20th of the same month. The timeline was structured as follows:

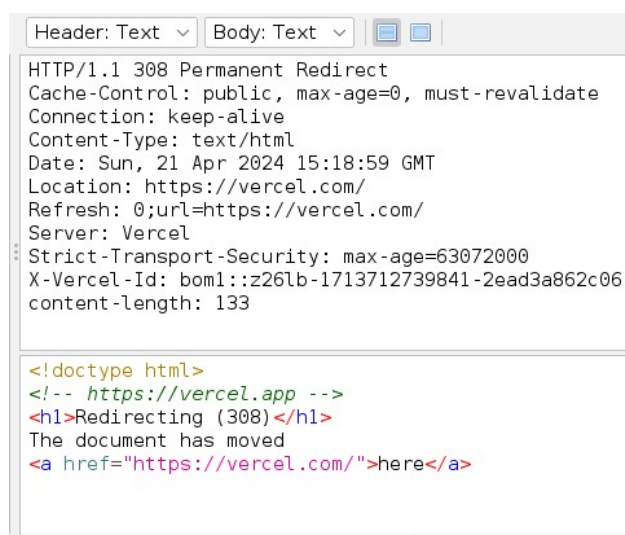
- **1. Project Kickoff (Day 1: 17th)**
Initial project briefing and goal setting.
Allocation of resources and assignment of responsibilities.
Setup of testing environments, including the configuration of testing tools (e.g., ZAP, Burp Suite).

- 2. Vulnerability Assessment (Days 2-3: 18th-19th)
Conducted automated scans using ZAP and Burp Suite to identify common vulnerabilities, such as SQL injection, Cross-Site Scripting (XSS), and authentication bypass.
Executed manual testing procedures to validate automated findings, explore potential attack vectors, and identify complex security issues.
Documented and categorized vulnerabilities based on severity levels (e.g., critical, high, medium, low).

2.5 summary of Findings

2.5.1 Cloud Metadata Potentially Exposed

1. **Url** : <https://appointment-assistant.vercel.app/latest/meta-data/>
2. **Risk** : High
3. **Confidence** : Low
4. **Attack** : 169.254.169.254
5. **CWE ID** : 0
6. **WASC ID** : 0
7. **Source** : Active(90034-Cloud Metadata Potentially Exposed)
8. **description** : The Cloud Metadata Attack attempts to abuse a misconfigured NGINX server in order to access the instance metadata maintained by cloud service providers such as AWS, GCP and Azure. All of these providers provide metadata via an internal unroutable IP address '169.254.169.254'-this can be exposed by incorrectly configured NGINX servers and accessed by using this IP address in the Host header field.
9. **More Info** : Based on the successful response status code cloud metadata may have been returned in the response. Check the response data to see if any cloud metadata has been returned. The meta data returned can include information that would allow an attacker to completely compromise the system.
10. **solution** : Do not trust any user data in NGINX configs. In this case it is probably the use of the host variable which is set from the 'Host' header and can be controlled by an attacker.



```
Header: Text  Body: Text  [icon] [icon]
HTTP/1.1 308 Permanent Redirect
Cache-Control: public, max-age=0, must-revalidate
Connection: keep-alive
Content-Type: text/html
Date: Sun, 21 Apr 2024 15:18:59 GMT
Location: https://vercel.com/
Refresh: 0;url=https://vercel.com/
Server: Vercel
Strict-Transport-Security: max-age=63072000
X-Vercel-Id: bom1::z26lb-1713712739841-2ead3a862c06
content-length: 133

<!doctype html>
<!-- https://vercel.app -->
<h1>Redirecting (308)</h1>
The document has moved
<a href="https://vercel.com/">here</a>
```

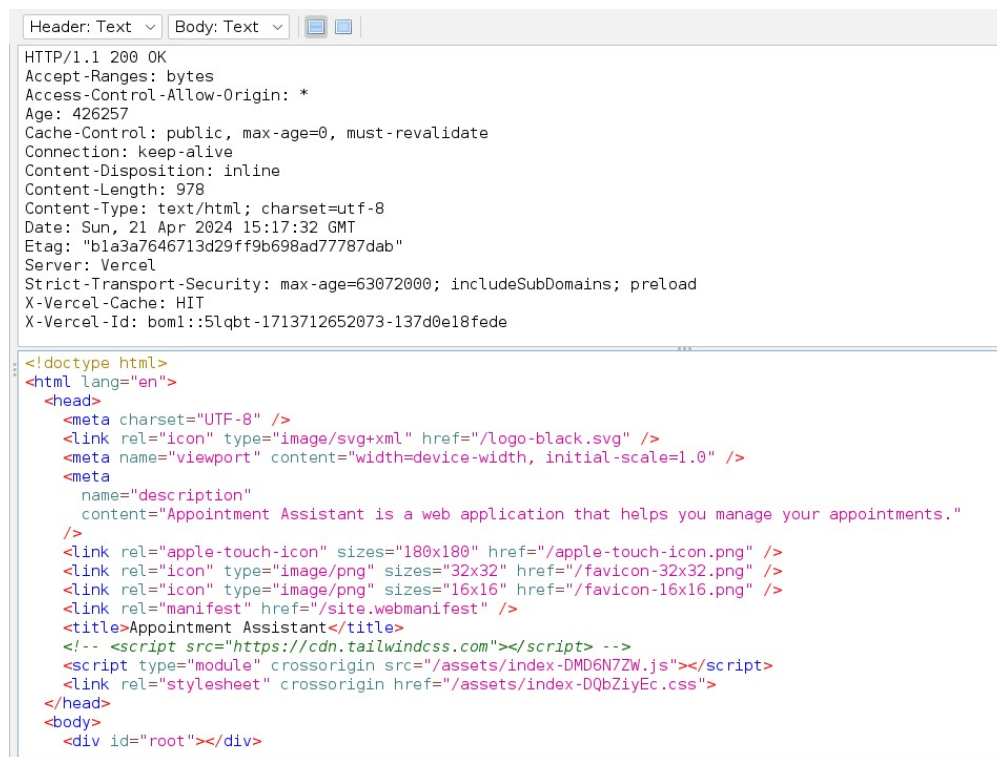
Figure 2.1: Header and Body Script

Cloud Metadata Potentially Exposed	
URL:	https://appointment-assistant.vercel.app/latest/meta-data/
Risk:	High
Confidence:	Low
Parameter:	
Attack:	169.254.169.254
Evidence:	
CWE ID:	0
WASC ID:	0
Source:	Active (90034 - Cloud Metadata Potentially Exposed)
Input Vector:	
Description:	The Cloud Metadata Attack attempts to abuse a misconfigured NGINX server in order to access the instance metadata maintained by cloud service providers such as AWS, GCP and Azure. All of these providers provide metadata via an internal unrouteable IP address '169.254.169.254' - this can be exposed by incorrectly configured NGINX servers and accessed by using this IP address in the Host header field.
Other Info:	Based on the successful response status code cloud metadata may have been returned in the response. Check the response data to see if any cloud metadata has been returned. The meta data returned can include information that would allow an attacker to completely compromise the system.
Solution:	Do not trust any user data in NGINX configs. In this case it is probably the use of the \$host variable which is set from the 'Host' header and can be controlled by an attacker.
Reference:	https://www.nginx.com/blog/trust-no-one-perils-of-trusting-user-input/

Figure 2.2: Cloud Metadata Potentially Exposed

2.5.2 Content Security Policy (CSP) Header Not Set

1. **Url** : <https://appointment-assistant.vercel.app/>
2. **Risk** : Medium
3. **Confidence** : High
4. **CWE ID** : 693
5. **WASC ID** : 15
6. **Source** : Passive(10038-Content Security Policy (CSP) Header Not Set)
7. **description** : Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.
8. **solution** :Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.



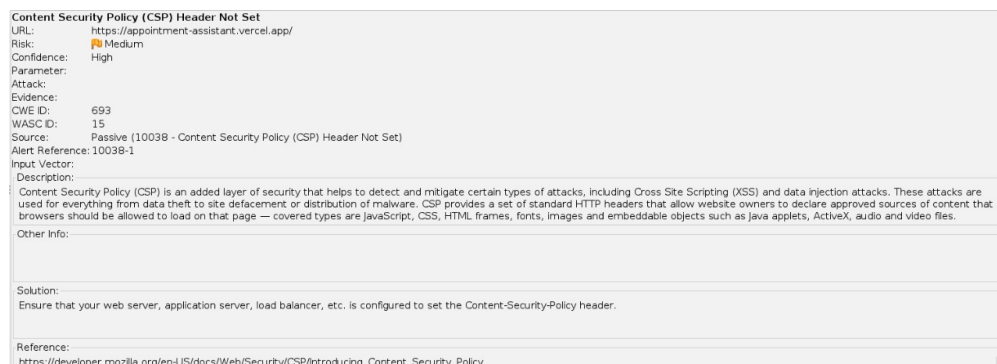
```

Header: Text Body: Text
HTTP/1.1 200 OK
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Age: 426257
Cache-Control: public, max-age=0, must-revalidate
Connection: keep-alive
Content-Disposition: inline
Content-Length: 978
Content-Type: text/html; charset=utf-8
Date: Sun, 21 Apr 2024 15:17:32 GMT
Etag: "bla3a7646713d29ff9b698ad77787dab"
Server: Vercel
Strict-Transport-Security: max-age=63072000; includeSubDomains; preload
X-Verce: Cache: HIT
X-Verce-Id: boml::5lqbt-1713712652073-137d0e18fed

<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link rel="icon" type="image/svg+xml" href="/logo-black.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta
      name="description"
      content="Appointment Assistant is a web application that helps you manage your appointments."
    />
    <link rel="apple-touch-icon" sizes="180x180" href="/apple-touch-icon.png" />
    <link rel="icon" type="image/png" sizes="32x32" href="/favicon-32x32.png" />
    <link rel="icon" type="image/png" sizes="16x16" href="/favicon-16x16.png" />
    <link rel="manifest" href="/site.webmanifest" />
    <title>Appointment Assistant</title>
    <!-- <script src="https://cdn.tailwindcss.com"></script> -->
    <script type="module" crossorigin src="/assets/index-DMD6N7ZW.js"></script>
    <link rel="stylesheet" crossorigin href="/assets/index-DQbZiyEc.css">
  </head>
  <body>
    <div id="root"></div>
  </body>
</html>

```

Figure 2.3: Header and Body Script



Content Security Policy (CSP) Header Not Set

URL: <https://appointment-assistant.vercel.app/>

Risk: Medium

Confidence: High

Parameter:

Attack:

Evidence:

CWE ID: 693

WASC ID: 15

Source: Passive (10038 - Content Security Policy (CSP) Header Not Set)

Alert Reference: 10038-1

Input Vector:

Description:

Content Security Policy (CSP) is an added layer of security that helps to detect and mitigate certain types of attacks, including Cross Site Scripting (XSS) and data injection attacks. These attacks are used for everything from data theft to site defacement or distribution of malware. CSP provides a set of standard HTTP headers that allow website owners to declare approved sources of content that browsers should be allowed to load on that page — covered types are JavaScript, CSS, HTML frames, fonts, images and embeddable objects such as Java applets, ActiveX, audio and video files.

Other Info:

Solution:

Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.

Reference:

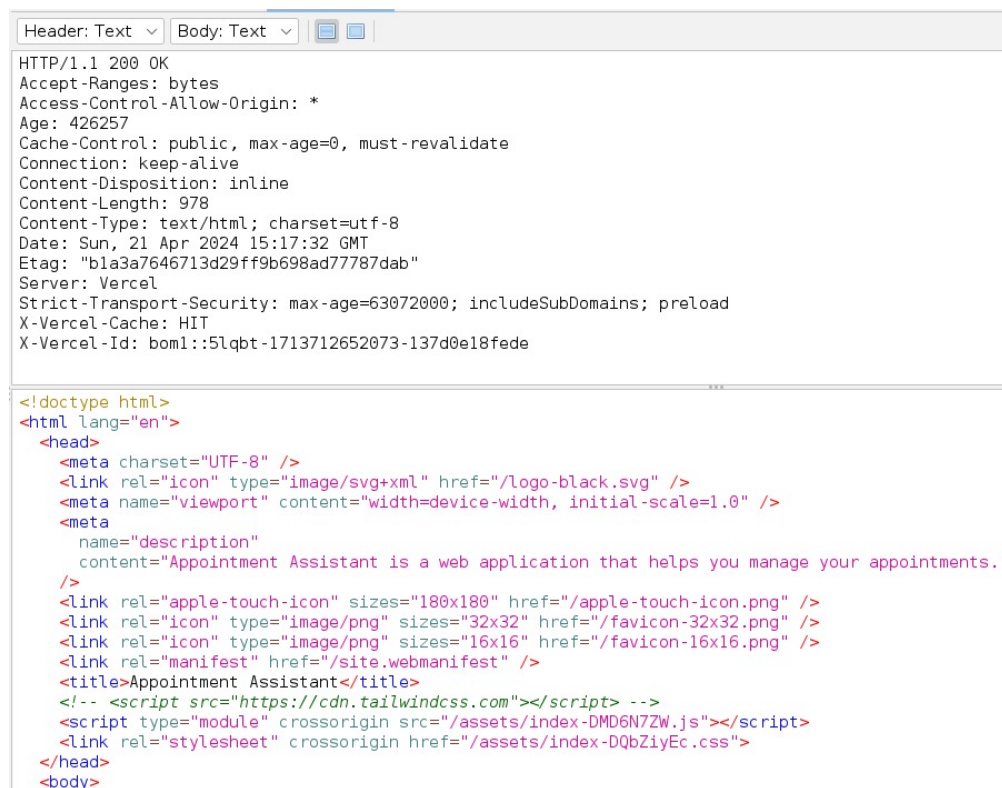
https://developer.mozilla.org/en-US/docs/Web/Security/CSP/introducing_Content_Security_Policy

Figure 2.4: Content Security Policy (CSP) Header Not Set

2.5.3 Missing Anti-clickjacking Header

1. **Url** : <https://appointment-assistant.vercel.app/>
2. **Risk** : Medium
3. **Confidence** : Medium
4. **CWE ID** : 1021
5. **WASC ID** : 15
6. **Source** : Passive(10020-Content Security Policy (CSP) Header Not Set)

7. **description** : The response does not include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options to protect against 'Clickjacking' attacks.
8. **solution** : Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app. If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.



```

Header: Text  Body: Text
HTTP/1.1 200 OK
Accept-Ranges: bytes
Access-Control-Allow-Origin: *
Age: 426257
Cache-Control: public, max-age=0, must-revalidate
Connection: keep-alive
Content-Disposition: inline
Content-Length: 978
Content-Type: text/html; charset=utf-8
Date: Sun, 21 Apr 2024 15:17:32 GMT
Etag: "bla3a7646713d29ff9b698ad77787dab"
Server: Vercel
Strict-Transport-Security: max-age=63072000; includeSubDomains; preload
X-Vercel-Cache: HIT
X-Vercel-Id: boml::5lqbt-1713712652073-137d0e18fede

<!doctype html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <link rel="icon" type="image/svg+xml" href="/logo-black.svg" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <meta
      name="description"
      content="Appointment Assistant is a web application that helps you manage your appointments."
    />
    <link rel="apple-touch-icon" sizes="180x180" href="/apple-touch-icon.png" />
    <link rel="icon" type="image/png" sizes="32x32" href="/favicon-32x32.png" />
    <link rel="icon" type="image/png" sizes="16x16" href="/favicon-16x16.png" />
    <link rel="manifest" href="/site.webmanifest" />
    <title>Appointment Assistant</title>
    <!-- <script src="https://cdn.tailwindcss.com"></script> -->
    <script type="module" crossorigin src="/assets/index-DMD6N7ZW.js"></script>
    <link rel="stylesheet" crossorigin href="/assets/index-DQbZiyEc.css">
  </head>
  <body>

```

Figure 2.5: Header and Body Script



```

Missing Anti-clickjacking Header
URL: https://appointment-assistant.vercel.app/
Risk: Medium
Confidence: Medium
Parameter: x-frame-options
Attack:
Evidence:
CWE ID: 1021
WASC ID: 15
Source: Passive (10020 - Anti-clickjacking Header)
Alert Reference: 10020-1
Input Vector:
Description:
The response does not include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options to protect against 'Clickjacking' attacks.

Other Info:

Solution:
Modern Web browsers support the Content-Security-Policy and X-Frame-Options HTTP headers. Ensure one of them is set on all web pages returned by your site/app.
If you expect the page to be framed only by pages on your server (e.g. it's part of a FRAMESET) then you'll want to use SAMEORIGIN, otherwise if you never expect the page to be framed, you should use DENY. Alternatively consider implementing Content Security Policy's "frame-ancestors" directive.

```

Figure 2.6: Missing Anti-clickjacking Header

2.5.4 Cross-Domain Misconfiguration

1. **Url** : <https://appointment-assistant.vercel.app/>

2. **Risk** : Medium
3. **Confidence** : Medium
4. **CWE ID** : 264
5. **WASC ID** : 14
6. **Source** : Passive(10098-Cross-Domain Misconfiguration)
7. **description** : Web browser data loading may be possible, due to a Cross Origin Resource Sharing (CORS) misconfiguration on the web server
8. **Other info** : The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing
9. **solution** : Ensure that sensitive data is not available in an unauthenticated manner (using IP address white-listing, for instance). Configure the "Access-Control-Allow-Origin" HTTP header to a more restrictive set of domains, or remove all CORS headers entirely, to allow the web browser to enforce the Same Origin Policy (SOP) in a more restrictive manner.

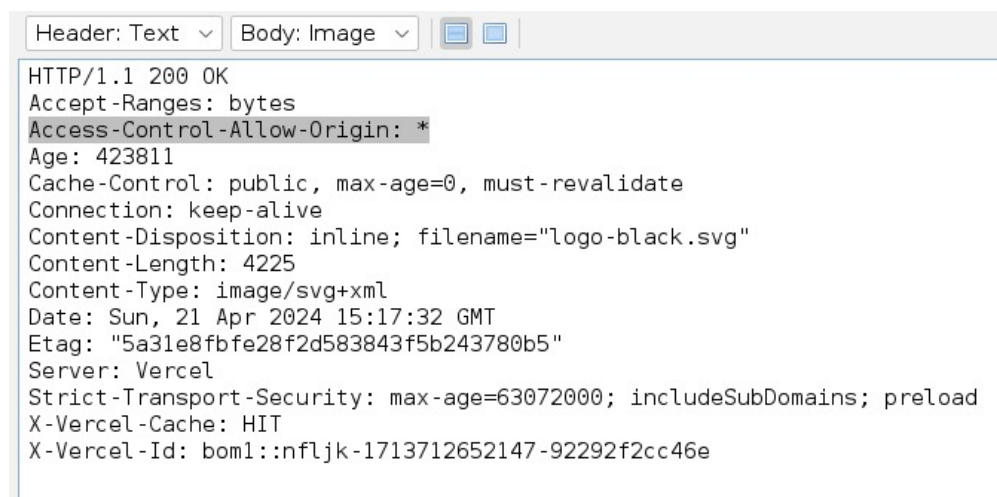


Figure 2.7: Header and Body Script



Figure 2.8: Cross-Domain Misconfiguration

2.5.5 Strict-Transport-Security Header Not Set

1. **Url** : <https://appointment-assistant.vercel.app/>
2. **Risk** : Low
3. **Confidence** : High
4. **CWE ID** :
5. **WASC ID** : 15
6. **Source** : Passive(10035-Strict-Transport-Security Header Not Set)
7. **description** : HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standards track protocol and is specified in RFC 6797.
8. **solution** :Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.

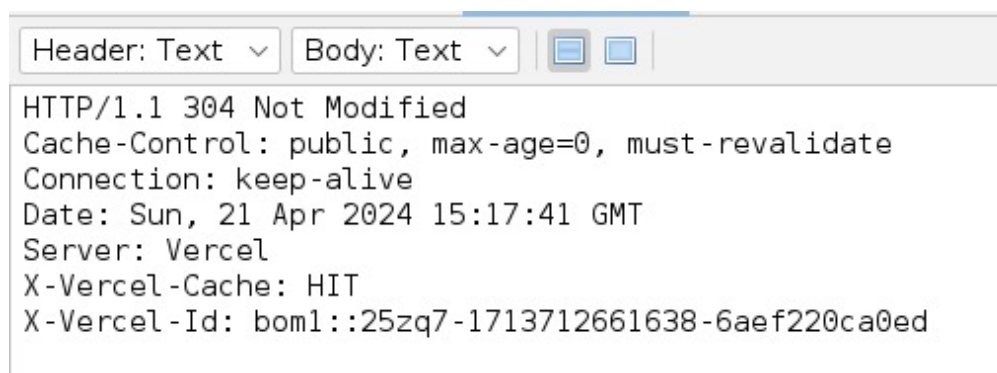


Figure 2.9: Header and Body Script



Figure 2.10: Strict-Transport-Security Header Not Set

2.5.6 X-Content-Type-Options Header Missing

1. **Url** : <https://appointment-assistant.vercel.app/>
2. **Risk** : Low

3. **Confidence** : Medium
4. **CWE ID** : 693
5. **WASC ID** : 15
6. **Source** : Passive(10021-X-Content-Type-Options Header Missing)
7. **Description** : The Anti-MIME-Sniffing header X-Content-Type-Options was not set to 'nosniff'. This allows older versions of Internet Explorer and Chrome to perform MIME-sniffing on the response body, potentially causing the response body to be interpreted and displayed as a content type other than the declared content type. Current (early 2014) and legacy versions of Firefox will use the declared content type (if one is set), rather than performing MIME-sniffing.
8. **Solution** Ensure that the application/web server sets the Content-Type header appropriately, and that it sets the X-Content-Type-Options header to 'nosniff' for all web pages. If possible, ensure that the end user uses a standards-compliant and modern web browser that does not perform MIME-sniffing at all, or that can be directed by the web application/web server to not perform MIME-sniffing.

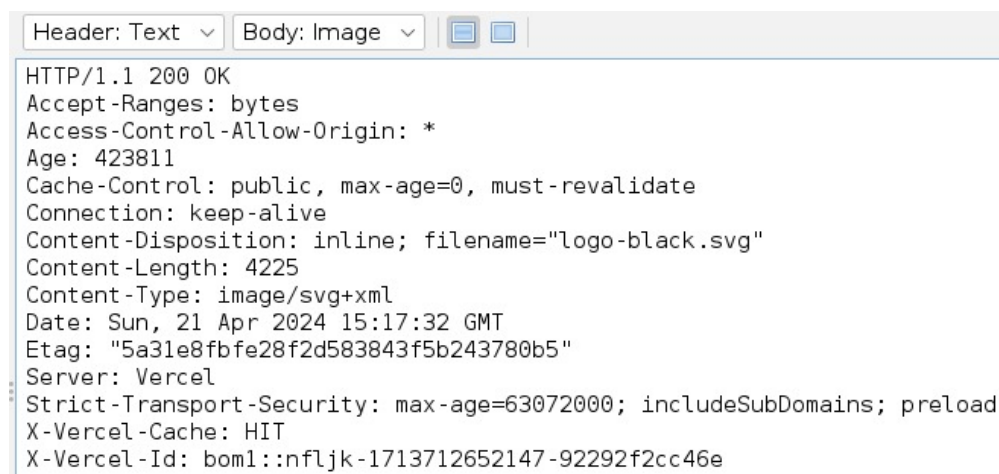


Figure 2.11: Header and Body Script

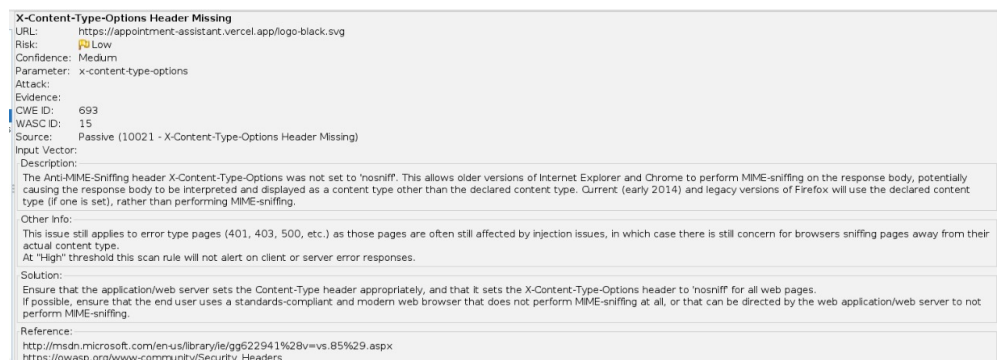


Figure 2.12: X-Content-Type-Options Header Missing

2.5.7 Burp Suit Report

1. Request Method: OPTIONS

2. Request URL: https://identitytoolkit.googleapis.com/v1/accounts:signInWithPassword?key=AIzaSyC3n_giJsoJsyVBWB6Bp4kWAjibEofEALo
3. Host: identitytoolkit.googleapis.com
4. Accept: /
5. Access-Control-Request-Method: POST
6. Access-Control-Request-Headers: content-type, x-client-version, x-firebase-client
7. Origin: <https://appointment-assistant.vercel.app/>
8. User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/122.0.6261.112 Safari/537.36
9. Sec-Fetch-Mode: cors
10. Sec-Fetch-Site: cross-site
11. Sec-Fetch-Dest: empty
12. Accept-Encoding: gzip, deflate, br
13. Accept-Language: en-US, en; q=0.9
14. Priority: u=1, 1
15. Connection: close
16. This information provides details about the intercepted HTTP OPTIONS request, including the target URL, headers, user agent, and other metadata related to the request.

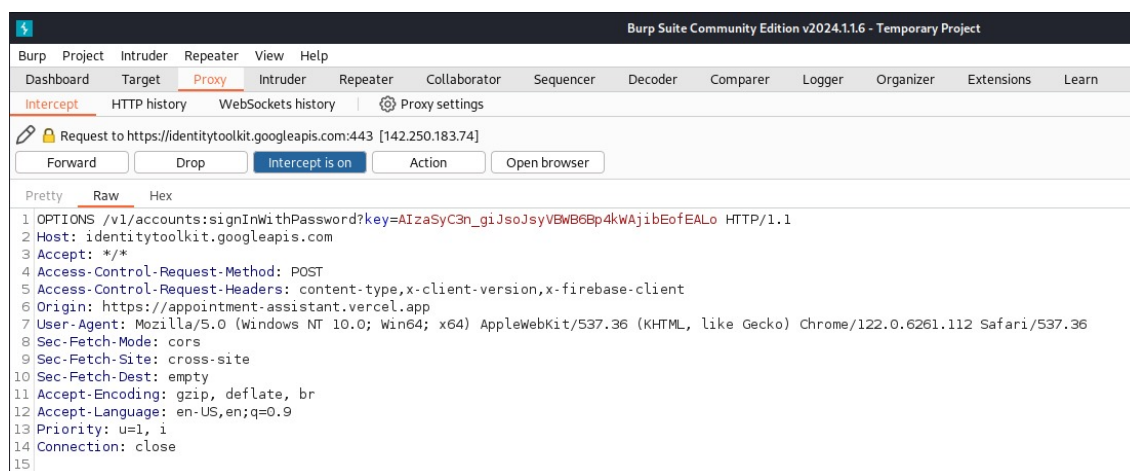


Figure 2.13: Header and Body Script

2.6 summary of Recommendations

- 1. **Access-Control-Allow-Origin Header:** Ensure that the server's response includes appropriate Access-Control-Allow-Origin headers to control cross-origin resource sharing (CORS) and prevent unauthorized access from different origins. Implement strict CORS policies based on the application's requirements, limiting access to trusted domains only.

- **2. Sec-Fetch Headers:** Review how the application handles Fetch Metadata headers (Sec-Fetch-Mode, Sec-Fetch-Site, Sec-Fetch-Dest) to prevent potential security bypass or spoofing attacks. Implement server-side validation and verification of Fetch Metadata headers to ensure they are consistent with expected behavior.
- **3. User-Agent Header:** Be cautious of relying solely on user-agent information for access control or security decisions, as user-agent headers can be manipulated by attackers. Implement additional security measures such as session tokens, IP address logging, and behavior analysis to enhance user identification and authentication.
- **4. Authentication Endpoint:** Conduct a thorough security assessment of the authentication endpoint (/v1/accounts:signInWithPassword) to identify and remediate vulnerabilities related to authentication and session management. Implement secure authentication practices, including strong password policies, multi-factor authentication (MFA), account lockout mechanisms, and secure session handling.
- **5. Input Validation and Sanitization:** Implement rigorous input validation and sanitization mechanisms across all input fields, parameters, and payloads to prevent injection vulnerabilities (e.g., SQL injection, XSS). Utilize parameterized queries for database interactions, encode user inputs properly, and sanitize data before rendering it in HTML to mitigate XSS risks.

Chapter 3

Methodology

3.1 Planning

The vulnerability assessment was meticulously planned to ensure comprehensive coverage of the appointment assistant website's security posture. The planning phase involved the following key steps

- **1. Scope Definition:** Clearly defined the scope of the assessment, including specific functionalities, components, and areas of the website to be tested (e.g., login functionality, data handling, API endpoints).
- **2. Tool Selection:** Selected appropriate tools for vulnerability assessment, including OWASP Zed Attack Proxy (ZAP) and Burp Suite, considering their capabilities for automated scanning, manual testing, and in-depth analysis.
- **3. Resource Allocation:** Assigned skilled security professionals to perform the assessment, including penetration testers, security analysts, and developers with expertise in web application security.
- **4. Environment Setup :** Set up testing environments, such as staging servers or virtualized environments, to simulate real-world scenarios without impacting the production environment.
- **5. Documentation:** Documented the assessment plan, including objectives, methodologies, testing protocols, timelines, and reporting guidelines, to ensure clarity and alignment with project goals.

3.2 Exploitation

The exploitation phase involved testing and exploiting identified vulnerabilities to assess their severity and potential impact on the appointment assistant website. The following techniques and approaches were used:

- **Automated Scanning:** Utilized automated scanning tools like ZAP and Burp Suite to conduct vulnerability scans across the defined scope, identifying common vulnerabilities such as SQL injection, Cross-Site Scripting (XSS), and insecure configurations.
- **Manual Testing:** Performed manual testing procedures to validate automated findings, explore complex attack vectors, and identify nuanced security weaknesses that automated scans might miss.
- **Attack Simulation:** Simulated real-world attack scenarios, including but not limited to authentication bypass, privilege escalation, data exfiltration, and business logic vulnerabilities, to assess the application's resilience against advanced threats.

3.3 Reporting

The reporting phase involved documenting and communicating assessment findings, vulnerabilities, and recommended actions to stakeholders. The reporting process followed these steps:

- **Vulnerability Classification:** Classified identified vulnerabilities based on severity levels (e.g., critical, high, medium, low) and potential impact on the security, functionality, and data integrity of the appointment assistant website.
- **Detailed Documentation:** Documented detailed findings for each identified vulnerability, including descriptions, attack vectors, exploitation scenarios, risk assessments, and recommendations for mitigation.

Chapter 4

Detail Findings

- **Cloud Metadata Potentially Exposed**
- **Content Security Policy (CSP) Header Not Set**
- **Missing Anti-clickjacking Header**
- **Cross-Domain Misconfiguration**
- **Strict-Transport-Security Header Not Set**
- **X-Content-Type-Options Header Missing**
- **Burp Suit Report**

4.1 Windows Server Information

Item	Value
OS Name	Microsoft Windows 11 Pro
Version	10.0.22631 Build 22631
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	PARTH
System Manufacturer	Micro-Star International Co., Ltd.
System Model	MS-7D91
System Type	x64-based PC
System SKU	Default string
Processor	Intel(R) Core(TM) i7-14700K, 3400 Mhz, 20 Core(s), 28 Logical Processor(s)
BIOS Version/Date	American Megatrends International, LLC. H.70, 10-07-2023
SMBIOS Version	3.6
Embedded Controller Version	255.255
BIOS Mode	UEFI
BaseBoard Manufacturer	Micro-Star International Co., Ltd.
BaseBoard Product	MAG Z790 TOMAHAWK WIFI (MS-7D91)
BaseBoard Version	4.0
Platform Role	Desktop
Secure Boot State	On
PCR7 Configuration	Elevation Required to View
Windows Directory	C:\WINDOWS
System Directory	C:\WINDOWS\system32
Boot Device	\Device\HarddiskVolume1
Locale	United States
Hardware Abstraction Layer	Version = "10.0.22621.2506"
User Name	PARTH\pc
Time Zone	India Standard Time
Installed Physical Memory (RAM)	32.0 GB
Total Physical Memory	31.8 GB
Available Physical Memory	21.9 GB
Total Virtual Memory	36.5 GB
Available Virtual Memory	18.7 GB
Page File Space	4.75 GB
Page File	C:\pagefile.sys
Kernel DMA Protection	On
Virtualization-based security	Running

Figure 4.1: Windows Server Information

Bibliography

- [1] <https://www.zaproxy.org/getting-started/>
- [2] <https://portswigger.net/burp/documentation>
- [3] <https://demo.infopercept.com/assets/download/VAPT-Sample-Report.pdf>