Report

Website: https://vsr-janata-mart.business.site

Open ports: https://www.shodan.io/host/142.251.42.78

Vulnerability name: Missing Content-Type Header

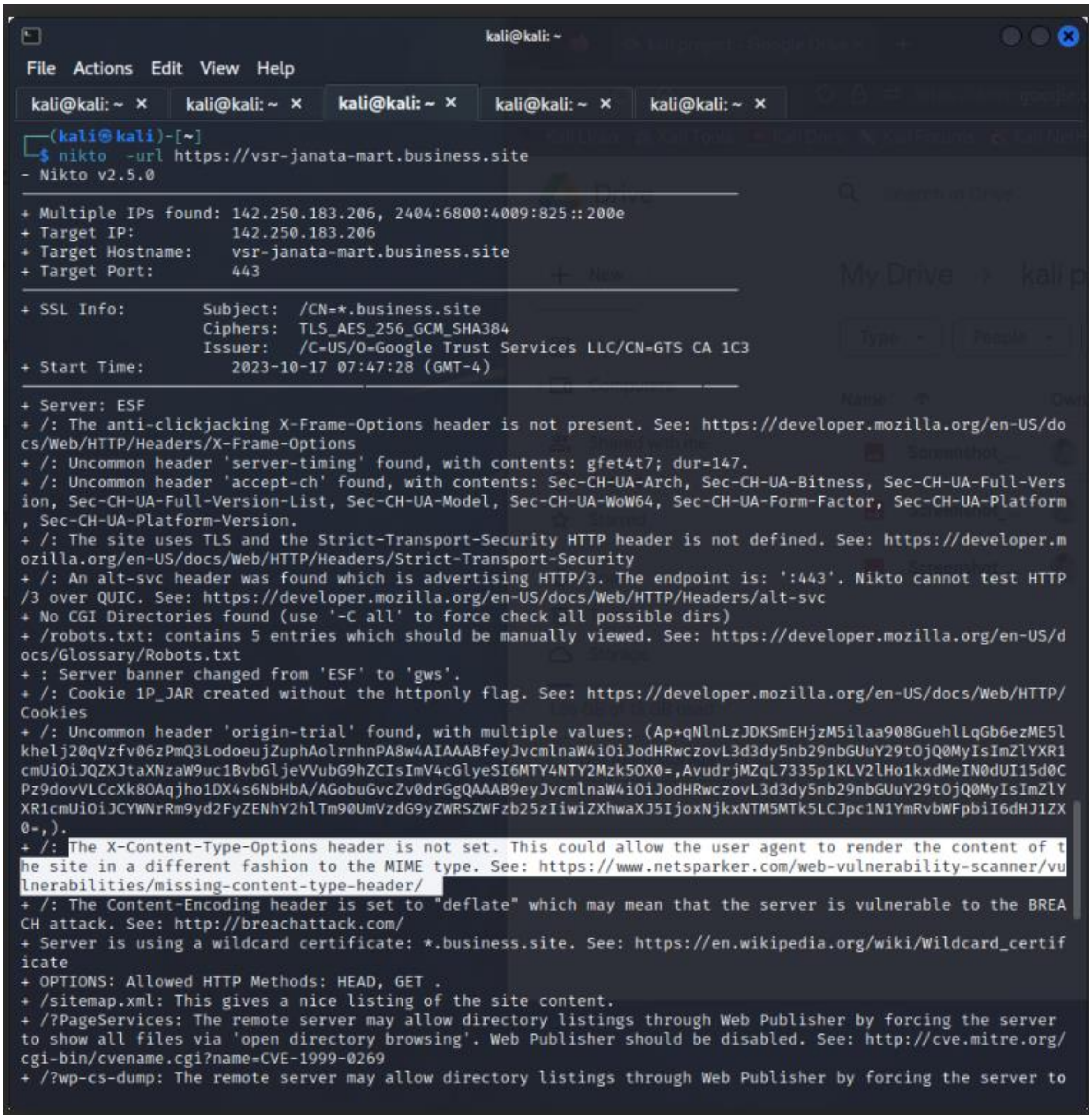
CWE: 16 (Low)

OSWAP: 2017-A6, 2013-A5

Description: Invicti detected a missing `Content-Type` header which means that this website could be at risk of a MIME-sniffing attacks.

Business impact: MIME type sniffing is a standard functionality in browsers to find an appropriate way to render data where the HTTP headers sent by the server are either inconclusive or missing.
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 intended content type.
The problem arises once a website allows users to upload content which is then published on the web server. If an attacker can carry out XSS (Cross-site Scripting) attack by manipulating the content in a way to be accepted by the web application and rendered as HTML by the browser, it is possible to inject code in e.g. an image file and make the victim execute it by viewing the image.

POC (Proof of Concept):



```
┌──(kali㉿kali)-[~]
└─$ nikto  -url https://vsr-janata-mart.business.site
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Multiple IPs found: 142.250.183.206, 2404:6800:4009:825::200e
+ Target IP:          142.250.183.206
+ Target Hostname:    vsr-janata-mart.business.site
+ Target Port:        443
---------------------------------------------------------------------------
+ SSL Info:        Subject:  /CN=*.business.site
                   Ciphers:  TLS_AES_256_GCM_SHA384
                   Issuer:   /C=US/O=Google Trust Services LLC/CN=GTS CA 1C3
+ Start Time:         2023-10-17 07:47:28 (GMT-4)
---------------------------------------------------------------------------
+ Server: ESF
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/do
cs/Web/HTTP/Headers/X-Frame-Options
+ /: Uncommon header 'server-timing' found, with contents: gfet4t7; dur=147.
+ /: Uncommon header 'accept-ch' found, with contents: Sec-CH-UA-Arch, Sec-CH-UA-Bitness, Sec-CH-UA-Full-Vers
ion, Sec-CH-UA-Full-Version-List, Sec-CH-UA-Model, Sec-CH-UA-WoW64, Sec-CH-UA-Form-Factor, Sec-CH-UA-Platform
, Sec-CH-UA-Platform-Version.
+ /: The site uses TLS and the Strict-Transport-Security HTTP header is not defined. See: https://developer.m
ozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP
/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /robots.txt: contains 5 entries which should be manually viewed. See: https://developer.mozilla.org/en-US/d
ocs/Glossary/Robots.txt
+ : Server banner changed from 'ESF' to 'gws'.
+ /: Cookie 1P_JAR created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/
Cookies
+ /: Uncommon header 'origin-trial' found, with multiple values: (Ap+qNlnLzJDKSmEHjzM5ilaa908GuehlLqGb6ezME5l
khelj20qVzfv06zPmQ3LodoeujZuphAolrnhnPA8w4AIAAABfeyJvcmlnaW4iOiJodHRwczovL3d3dy5nb29nbGUuY29tOjQ0MyIsImZlYXR1
cmUiOiJQZXJtaXNzaW9uc1BvbGljeVVubG9hZCIsImV4cGlyeSI6MTY4NTY2Mzk5OX0=,AvudrjMZqL7335p1KLV2lHo1kxdMeIN0dUI15d0C
Pz9dovVLCcXk8OAqjho1DX4s6NbHbA/AGobuGvcZv0drGgQAAAB9eyJvcmlnaW4iOiJodHRwczovL3d3dy5nb29nbGUuY29tOjQ0MyIsImZlY
XR1cmUiOiJCYWNrRm9yd2FyZENhY2hlTm90UmVzdG9yZWRSZWFzb25zIiwiZXhwaXJ5IjoxNjkxNTM5MTk5LCJpc1N1YmRvbWFpbiI6dHJ1ZX
0=,).
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of t
he site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vu
lnerabilities/missing-content-type-header/
+ /: The Content-Encoding header is set to "deflate" which may mean that the server is vulnerable to the BREA
CH attack. See: http://breachattack.com/
+ Server is using a wildcard certificate: *.business.site. See: https://en.wikipedia.org/wiki/Wildcard_certif
icate
+ OPTIONS: Allowed HTTP Methods: HEAD, GET .
+ /sitemap.xml: This gives a nice listing of the site content.
+ /?PageServices: The remote server may allow directory listings through Web Publisher by forcing the server
to show all files via 'open directory browsing'. Web Publisher should be disabled. See: http://cve.mitre.org/
cgi-bin/cvename.cgi?name=CVE-1999-0269
+ /?wp-cs-dump: The remote server may allow directory listings through Web Publisher by forcing the server to
```

Remediation:

1. When serving resources, make sure you send the content-type header to appropriately match the type of the resource being served. For example, if you are serving an HTML page, you should send the HTTP header:

   ```
   Content-Type: text/html
   ```

2. Add the X-Content-Type-Options header with a value of "nosniff" to inform the browser to trust what the site has sent is the appropriate content-type, and to not attempt "sniffing" the real content-type.

   ```
   X-Content-Type-Options: nosniff
   ```

Vulnerability name: Missing X Frame Option Header

CWE: 693 (Low)

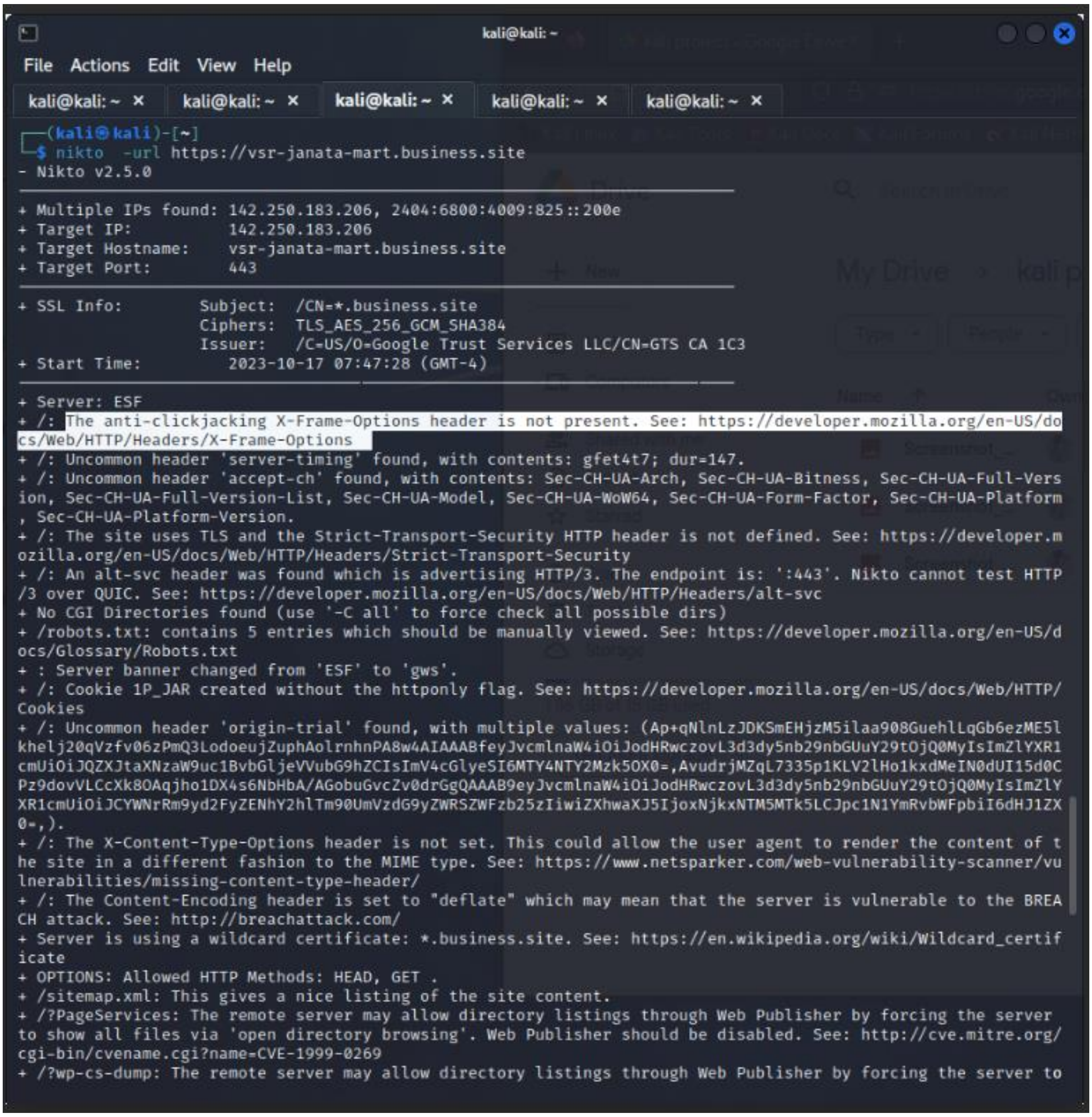
OSWAP: 2017-A6, 2013-A5

Description: a missing `X-Frame-Options` header which tells us that this website could be at risk of a clickjacking attack.
The `X-Frame-Options` HTTP header field indicates a policy that specifies whether the browser should render the transmitted resource within a `frame` or an `iframe`. Servers can declare this policy in the header of their HTTP responses to prevent clickjacking attacks, which ensures that their content is not embedded into other pages or frames.

Business impact:

Clickjacking is when an attacker uses multiple transparent or opaque layers to trick a user into clicking on a button or link on a framed page when they were intending to click on the top level page. Thus, the attacker is "hijacking" clicks meant for their page and routing them to other another page, most likely owned by another application, domain, or both.
Using a similar technique, keystrokes can also be hijacked. With a carefully crafted combination of stylesheets, iframes, and text boxes, a user can be led to believe they are typing in the password to their email or bank account, but are instead typing into an invisible frame controlled by the attacker.

POC (Proof of Concept):



```
┌──(kali㉿kali)-[~]
└─$ nikto -url https://vsr-janata-mart.business.site
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Multiple IPs found: 142.250.183.206, 2404:6800:4009:825::200e
+ Target IP:          142.250.183.206
+ Target Hostname:    vsr-janata-mart.business.site
+ Target Port:        443
---------------------------------------------------------------------------
+ SSL Info:        Subject:  /CN=*.business.site
                   Ciphers:  TLS_AES_256_GCM_SHA384
                   Issuer:   /C=US/O=Google Trust Services LLC/CN=GTS CA 1C3
+ Start Time:         2023-10-17 07:47:28 (GMT-4)
---------------------------------------------------------------------------
+ Server: ESF
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/do
cs/Web/HTTP/Headers/X-Frame-Options
+ /: Uncommon header 'server-timing' found, with contents: gfet4t7; dur=147.
+ /: Uncommon header 'accept-ch' found, with contents: Sec-CH-UA-Arch, Sec-CH-UA-Bitness, Sec-CH-UA-Full-Vers
ion, Sec-CH-UA-Full-Version-List, Sec-CH-UA-Model, Sec-CH-UA-WoW64, Sec-CH-UA-Form-Factor, Sec-CH-UA-Platform
, Sec-CH-UA-Platform-Version.
+ /: The site uses TLS and the Strict-Transport-Security HTTP header is not defined. See: https://developer.m
ozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP
/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /robots.txt: contains 5 entries which should be manually viewed. See: https://developer.mozilla.org/en-US/d
ocs/Glossary/Robots.txt
+ : Server banner changed from 'ESF' to 'gws'.
+ /: Cookie 1P_JAR created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/
Cookies
+ /: Uncommon header 'origin-trial' found, with multiple values: (Ap+qNlnLzJDKSmEHjzM5ilaa908GuehlLqGb6ezME5l
khelj20qVzfv06zPmQ3LodoeujZuphAolrnhnPA8w4AIAAABfeyJvcmlnaW4iOiJodHRwczovL3d3dy5nb29nbGUuY29tOjQ0MyIsImZlYXR1
cmUiOiJQZXJtaXNzaW9uc1BvbGljeVVubG9hZCIsImV4cGlyeSI6MTY4NTY2Mzk5OX0=,AvudrjMZqL7335p1KLV2lHo1kxdMeIN0dUI15d0C
Pz9dovVLCcXk8OAqjho1DX4s6NbHbA/AGobuGvcZv0drGgQAAAB9eyJvcmlnaW4iOiJodHRwczovL3d3dy5nb29nbGUuY29tOjQ0MyIsImZlY
XR1cmUiOiJCYWNrRm9yd2FyZENhY2hlTm90UmVzdG9yZWRSZWFzb25zIiwiZXhwaXJ5IjoxNjkxNTM5MTk5LCJpc1N1YmRvbWFpbiI6dHJ1ZX
0=,).
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of t
he site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vu
lnerabilities/missing-content-type-header/
+ /: The Content-Encoding header is set to "deflate" which may mean that the server is vulnerable to the BREA
CH attack. See: http://breachattack.com/
+ Server is using a wildcard certificate: *.business.site. See: https://en.wikipedia.org/wiki/Wildcard_certif
icate
+ OPTIONS: Allowed HTTP Methods: HEAD, GET .
+ /sitemap.xml: This gives a nice listing of the site content.
+ /?PageServices: The remote server may allow directory listings through Web Publisher by forcing the server
to show all files via 'open directory browsing'. Web Publisher should be disabled. See: http://cve.mitre.org/
cgi-bin/cvename.cgi?name=CVE-1999-0269
+ /?wp-cs-dump: The remote server may allow directory listings through Web Publisher by forcing the server to
```

Remediation:

- Sending the proper X-Frame-Options in HTTP response headers that instruct the browser to not allow framing from other domains.
  - `X-Frame-Options: DENY` It completely denies to be loaded in frame/iframe.
  - `X-Frame-Options: SAMEORIGIN` It allows only if the site which wants to load has a same origin.
  - `X-Frame-Options: ALLOW-FROM` *`URL`* It grants a specific URL to load itself in a iframe. However please pay attention to that, not all browsers support this.
- Employing defensive code in the UI to ensure that the current frame is the most top level window.

Vulnerability name: Breach Attack (Low)

CWE: 310

OSWAP: 2017-A3, 2021-A2

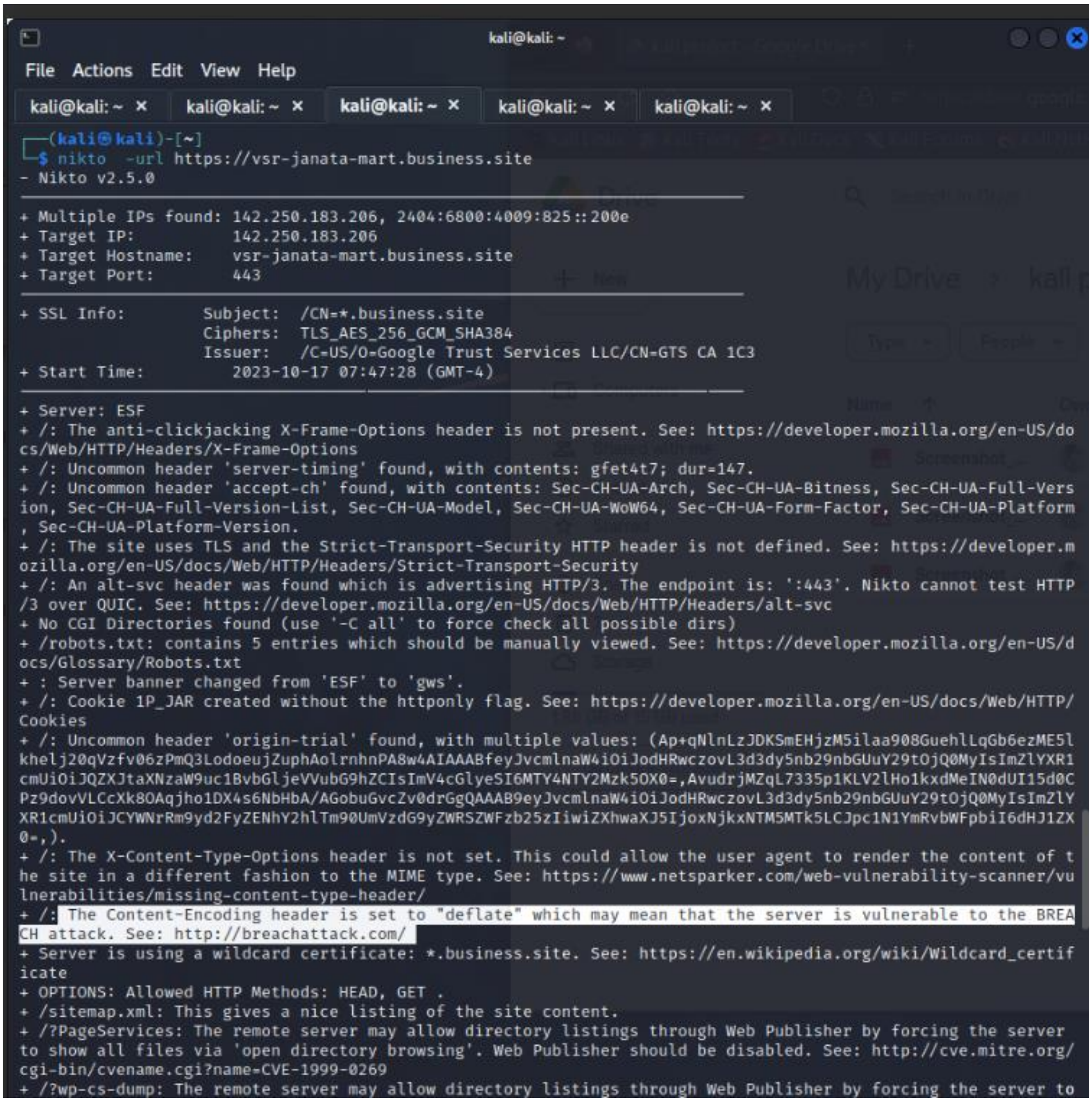
Description: BREACH is an instance of the CRIME attack against HTTP compression—the use of gzip or DEFLATE data compression algorithms via the content-encoding option within HTTP by many web browsers and servers. Given this compression oracle, the rest of the BREACH attack follows the same general lines as the CRIME exploit, by performing an initial blind brute-force search to guess a few bytes, followed by divide-and-conquer search to expand a correct guess to an arbitrarily large amount of content.

Business impact:

Even if you use an SSL/TLS protected connection, an attacker can still view the victim’s encrypted traffic and cause the victim to send HTTP requests to the vulnerable web server (by using invisible frames). Following these steps, an attacker could steal information from the website and do the following:

- Inject partial plaintext they have uncovered into a victim's requests
- Measure the size of encrypted traffic

POC (Proof of Concept):



```
┌──(kali㉿kali)-[~]
└─$ nikto  -url https://vsr-janata-mart.business.site
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Multiple IPs found: 142.250.183.206, 2404:6800:4009:825::200e
+ Target IP:          142.250.183.206
+ Target Hostname:    vsr-janata-mart.business.site
+ Target Port:        443
---------------------------------------------------------------------------
+ SSL Info:        Subject:  /CN=*.business.site
                   Ciphers:  TLS_AES_256_GCM_SHA384
                   Issuer:   /C=US/O=Google Trust Services LLC/CN=GTS CA 1C3
+ Start Time:         2023-10-17 07:47:28 (GMT-4)
---------------------------------------------------------------------------
+ Server: ESF
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/do
cs/Web/HTTP/Headers/X-Frame-Options
+ /: Uncommon header 'server-timing' found, with contents: gfet4t7; dur=147.
+ /: Uncommon header 'accept-ch' found, with contents: Sec-CH-UA-Arch, Sec-CH-UA-Bitness, Sec-CH-UA-Full-Vers
ion, Sec-CH-UA-Full-Version-List, Sec-CH-UA-Model, Sec-CH-UA-WoW64, Sec-CH-UA-Form-Factor, Sec-CH-UA-Platform
, Sec-CH-UA-Platform-Version.
+ /: The site uses TLS and the Strict-Transport-Security HTTP header is not defined. See: https://developer.m
ozilla.org/en-US/docs/Web/HTTP/Headers/Strict-Transport-Security
+ /: An alt-svc header was found which is advertising HTTP/3. The endpoint is: ':443'. Nikto cannot test HTTP
/3 over QUIC. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/alt-svc
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /robots.txt: contains 5 entries which should be manually viewed. See: https://developer.mozilla.org/en-US/d
ocs/Glossary/Robots.txt
+ : Server banner changed from 'ESF' to 'gws'.
+ /: Cookie 1P_JAR created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/
Cookies
+ /: Uncommon header 'origin-trial' found, with multiple values: (Ap+qNlnLzJDKSmEHjzM5ilaa908GuehlLqGb6ezME5l
khelj20qVzfv06zPmQ3LodoeujZuphAolrnhnPA8w4AIAAABfeyJvcmlnaW4iOiJodHRwczovL3d3dy5nb29nbGUuY29tOjQ0MyIsImZlYXR1
cmUiOiJQZXJtaXNzaW9uc1BvbGljeVVubG9hZCIsImV4cGlyeSI6MTY4NTY2Mzk5OX0=,AvudrjMZqL7335p1KLV2lHo1kxdMeIN0dUI15d0C
Pz9dovVLCcXk8OAqjho1DX4s6NbHbA/AGobuGvcZv0drGgQAAAB9eyJvcmlnaW4iOiJodHRwczovL3d3dy5nb29nbGUuY29tOjQ0MyIsImZlY
XR1cmUiOiJCYWNrRm9yd2FyZENhY2hlTm90UmVzdG9yZWRSZWFzb25zIiwiZXhwaXJ5IjoxNjkxNTM5MTk5LCJpc1N1YmRvbWFpbiI6dHJ1ZX
0=,).
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of t
he site in a different fashion to the MIME type. See: https://www.netsparker.com/web-vulnerability-scanner/vu
lnerabilities/missing-content-type-header/
+ /: The Content-Encoding header is set to "deflate" which may mean that the server is vulnerable to the BREA
CH attack. See: http://breachattack.com/
+ Server is using a wildcard certificate: *.business.site. See: https://en.wikipedia.org/wiki/Wildcard_certif
icate
+ OPTIONS: Allowed HTTP Methods: HEAD, GET .
+ /sitemap.xml: This gives a nice listing of the site content.
+ /?PageServices: The remote server may allow directory listings through Web Publisher by forcing the server
to show all files via 'open directory browsing'. Web Publisher should be disabled. See: http://cve.mitre.org/
cgi-bin/cvename.cgi?name=CVE-1999-0269
+ /?wp-cs-dump: The remote server may allow directory listings through Web Publisher by forcing the server to
```

Remediation:

Invicti reported a Possible BREACH Attack issue because the target web page meets the following conditions that facilitate it:

- Served from a server that uses HTTP-level compression (ie. gzip)
- Reflects user-input in the HTTP response bodies
- Contains sensitive information (such as a CSRF token) in HTTP response bodies

To mitigate the issue, we recommend the following solutions:

1. If possible, disable HTTP level compression
2. Separate sensitive information from user input

3. Protect vulnerable pages with CSRF token. The SameSite Cookie attribute will mitigate this issue, because to exploit this issue an attacker forces the victim to visit a target website using invisible frames. With the SameSite cookie attribute added, cookies that belong to the target won't be sent with a request that does not include top level navigation.
4. Hide the length of the traffic by adding a random number of bytes to the responses.
5. Add in a rate limit, so that the page maximum is reached five times per minute.

Vulnerability name: Missing security header: Strict-Transport-Security

CWE: 693

OSWAP: A5,A6

Description:

The vulnerability is named “Missing security header: Strict-Transport-Security.” This indicates that a web application or service does not have the HTTP Strict Transport Security (HSTS) header configured. HSTS is an important security mechanism that instructs web browsers to only interact with a website over secure, encrypted connections (HTTPS).

Business Impact:

The absence of the HSTS header can have several negative consequences for a business. Without HSTS, users might be exposed to man-in-the-middle attacks or session hijacking when accessing the application over insecure HTTP connections. This vulnerability can lead to data breaches, loss of user trust, and potential legal issues.

Remediation:

To remediate this vulnerability, the following steps should be taken:

1. Configure HSTS Header: Implement the HTTP Strict Transport Security (HSTS) header in your web server or application to ensure that all communication is conducted over secure HTTPS connections.
2. Set Appropriate HSTS Policy: Define the HSTS policy, including the max-age directive, which specifies how long the browser should enforce HTTPS, and the

includeSubDomains directive if necessary. These values depend on your specific requirements.

3. Ensure Proper Implementation: Ensure that the HSTS header is correctly implemented and tested. It should be included in the server's response headers for all web pages.
4. Monitoring and Reporting: Regularly monitor the application to ensure that the HSTS header is in place and functioning as expected. Set up a process to report and handle any violations.

By implementing the HSTS header with the appropriate policy, you can protect your users and the integrity of your web application, preventing the risks associated with unencrypted HTTP connections.

Vulnerability name: Unsafe security header: Content-Security-Policy

CWE:693

OSWAP:A5,A6

Description:

The vulnerability is named "Unsafe security header: Content-Security-Policy." This indicates that a web application or service has a Content Security Policy (CSP) header that is not configured correctly. CSP is a security feature that defines which resources can be loaded and executed on a web page, mitigating various types of attacks like Cross-Site Scripting (XSS).

Business Impact:

The absence or misconfiguration of a CSP header can have significant security implications for a business. Without a proper CSP, the application becomes more susceptible to XSS attacks, data theft, and other malicious activities. This can lead to data breaches, reputation damage, and potential legal consequences.

Remediation:

To remediate this vulnerability, the following steps should be taken:

1. Configure Content Security Policy (CSP) Header: Implement a properly configured CSP header in your web application to control which scripts and resources can be executed on a web page.

2. Define and Enforce a Secure CSP: Define and enforce a strict CSP policy that only allows trusted sources and domains to load content on your web page. Utilize directives like `default-src`, `script-src`, `style-src`, and others to specify the sources of content.

3. Test and Monitor: Test your CSP to ensure that it doesn't break the functionality of your application, and monitor for violations. Adjust your policy as necessary.

4. Educate Development Team: Ensure that your development team is aware of the importance of CSP and follows best practices for secure web development.

By implementing a properly configured CSP header, you can significantly reduce the risk of XSS attacks and enhance the security of your web application, protecting both your business and your users.