ZAP Scanning Report

Site: https://10.103.97.207

Generated on Mon, 3 Jun 2024 15:04:36

ZAP Version: 2.15.0

ZAP is supported by the Crash Override Open Source Fellowship

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	4

Alerts

Name	Risk Level	Number of Instances
Content Security Policy (CSP) Header Not Set	Medium	7
Directory Browsing	Medium	2
Hidden File Found	Medium	2
Missing Anti-clickjacking Header	Medium	5

Alert Detail

Medium	Content Security Policy (CSP) Header Not Set
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.
URL	https://10.103.97.207/E-cars4U/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/E-cars4U/index.php?success=registration_complete
Method	GET
Attack	
Evidence	
Other Info	

URL	https://10.103.97.207/E-cars4U/login.php
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/E-cars4U/login.php?error=user_not_found
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/E-cars4U/register.php
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/robots.txt
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/sitemap.xml
Method	GET
Attack	
Evidence	
Other Info	
Instances	7
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 https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html https://www.w3.org/TR/CSP/ https://w3c.github.io/webappsec-csp/ https://web.dev/articles/csp https://caniuse.com/#feat=contentsecuritypolicy https://content-security-policy.com/
CWE Id	693
WASC Id	15
Plugin Id	10038

Medium	Directory Browsing

Description	It is possible to view the directory listing. Directory listing may reveal hidden scripts, include files, backup source files, etc. which can be accessed to read sensitive information.
URL	https://10.103.97.207/E-cars4U/database/
Method	GET
Attack	https://10.103.97.207/E-cars4U/database/
Evidence	Parent Directory
Other Info	
URL	https://10.103.97.207/E-cars4U/style/
Method	GET
Attack	https://10.103.97.207/E-cars4U/style/
Evidence	Parent Directory
Other Info	
Instances	2
Solution	Disable directory browsing. If this is required, make sure the listed files does not induce risks.
Reference	https://httpd.apache.org/docs/mod/core.html#options
CWE Id	548
WASC Id	48
Plugin Id	0

Medium	**Hidden File Found**
Description	A sensitive file was identified as accessible or available. This may leak administrative, configuration, or credential information which can be leveraged by a malicious individual to further attack the system or conduct social engineering efforts.
URL	https://10.103.97.207/server-info
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	apache_server_info
URL	https://10.103.97.207/server-status
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	apache_server_status
Instances	2
Solution	Consider whether or not the component is actually required in production, if it isn't then disable it. If it is then ensure access to it requires appropriate authentication and authorization, or limit exposure to internal systems or specific source IPs, etc.
Reference	https://blog.hboeck.de/archives/892-Introducing-Snallygaster-a-Tool-to-Scan-for-Secrets-on-Web-Servers.html https://httpd.apache.org/docs/current/mod/mod_status.html
CWE Id	538
WASC Id	13

Plugin Id	40035

Medium	Missing Anti-clickjacking Header
Description	The response does not include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options to protect against 'ClickJacking' attacks.
URL	https://10.103.97.207/E-cars4U/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/E-cars4U/index.php?success=registration_complete
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/E-cars4U/login.php
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/E-cars4U/login.php?error=user_not_found
Method	GET
Attack	
Evidence	
Other Info	
URL	https://10.103.97.207/E-cars4U/register.php
Method	GET
Attack	
Evidence	
Other Info	
Instances	5
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.
Reference	https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
CWE Id	1021
WASC Id	15
Plugin Id	10020