



ZAP by Checkmarx Scanning Report

Site: <http://localhost:5000>

Generated on Thu, 11 Dec 2025 23:15:06

ZAP Version: 2.16.1

ZAP by [Checkmarx](#)

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	3
Low	4
Informational	2

Alerts

Name	Risk Level	Number of Instances
CSP: Failure to Define Directive with No Fallback	Medium	3
Content Security Policy (CSP) Header Not Set	Medium	2
Missing Anti-clickjacking Header	Medium	2
Cookie No HttpOnly Flag	Low	3
Cross-Domain JavaScript Source File Inclusion	Low	2
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	6
X-Content-Type-Options Header Missing	Low	3
Information Disclosure - Suspicious Comments	Informational	2
Session Management Response Identified	Informational	4

Alert Detail

Medium	CSP: Failure to Define Directive with No Fallback
Description	The Content Security Policy fails to define one of the directives that has no fallback. Missing /excluding them is the same as allowing anything.
URL	http://localhost:5000/index.html
Method	GET
Attack	
Evidence	default-src 'none'

Other Info	The directive(s): frame-ancestors, form-action is/are among the directives that do not fallback to default-src.
URL	http://localhost:5000/robots.txt
Method	GET
Attack	
Evidence	default-src 'none'
Other Info	The directive(s): frame-ancestors, form-action is/are among the directives that do not fallback to default-src.
URL	http://localhost:5000/sitemap.xml
Method	GET
Attack	
Evidence	default-src 'none'
Other Info	The directive(s): frame-ancestors, form-action is/are among the directives that do not fallback to default-src.
Instances	3
Solution	Ensure that your web server, application server, load balancer, etc. is properly configured to set the Content-Security-Policy header.
Reference	https://www.w3.org/TR/CSP/ https://caniuse.com/#search=content+security+policy https://content-security-policy.com/ https://github.com/HtmlUnit/htmlunit-csp https://web.dev/articles/csp#resource-options
CWE Id	693
WASC Id	15
Plugin Id	10055

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	http://localhost:5000/
Method	GET
Attack	
Evidence	
Other Info	
URL	http://localhost:5000/register
Method	GET
Attack	
Evidence	
Other Info	
Instances	2
Solution	Ensure that your web server, application server, load balancer, etc. is configured to set the Content-Security-Policy header.

	https://developer.mozilla.org/en-US/docs/Web/HTTP/Guides/CSP https://cheatsheetseries.owasp.org/cheatsheets/Content_Security_Policy_Cheat_Sheet.html
Reference	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	Missing Anti-clickjacking Header
Description	The response does not protect against 'ClickJacking' attacks. It should include either Content-Security-Policy with 'frame-ancestors' directive or X-Frame-Options.
URL	http://localhost:5000/
Method	GET
Attack	
Evidence	
Other Info	
URL	http://localhost:5000/register
Method	GET
Attack	
Evidence	
Other Info	
Instances	2
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/Reference/Headers/X-Frame-Options
CWE Id	1021
WASC Id	15
Plugin Id	10020

Low	Cookie No HttpOnly Flag
Description	A cookie has been set without the HttpOnly flag, which means that the cookie can be accessed by JavaScript. If a malicious script can be run on this page then the cookie will be accessible and can be transmitted to another site. If this is a session cookie then session hijacking may be possible.
URL	http://localhost:5000/
Method	GET
Attack	
Evidence	Set-Cookie: sessionID
Other Info	

URL	http://localhost:5000/robots.txt
Method	GET
Attack	
Evidence	Set-Cookie: sessionID
Other Info	
URL	http://localhost:5000/sitemap.xml
Method	GET
Attack	
Evidence	Set-Cookie: sessionID
Other Info	
Instances	3
Solution	Ensure that the HttpOnly flag is set for all cookies.
Reference	https://owasp.org/www-community/HttpOnly
CWE Id	1004
WASC Id	13
Plugin Id	10010

Low	Cross-Domain JavaScript Source File Inclusion
Description	The page includes one or more script files from a third-party domain.
URL	http://localhost:5000/
Method	GET
Attack	
Evidence	<script async defer src="https://buttons.github.io/buttons.js"></script>
Other Info	
URL	http://localhost:5000/register
Method	GET
Attack	
Evidence	<script async defer src="https://buttons.github.io/buttons.js"></script>
Other Info	
Instances	2
Solution	Ensure JavaScript source files are loaded from only trusted sources, and the sources can't be controlled by end users of the application.
Reference	
CWE Id	829
WASC Id	15
Plugin Id	10017

Low	Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)
Description	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.

URL	http://localhost:5000/
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:5000/img/favicon/favicon.ico
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:5000/index.html
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:5000/register
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:5000/robots.txt
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:5000/sitemap.xml
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
Instances	6
Solution	Ensure that your web server, application server, load balancer, etc. is configured to suppress "X-Powered-By" headers.
Reference	https://owasp.org/www-project-web-security-testing-guide/v42/4-Web_Application_Security_Testing/01-Information_Gathering/08-Fingerprint_Web_Application_Framework https://www.troyhunt.com/shhh-dont-let-your-response-headers/
CWE Id	497
WASC Id	13

Plugin Id	10037
Low	X-Content-Type-Options Header Missing
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.
URL	http://localhost:5000/
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	http://localhost:5000/img/favicon/favicon.ico
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	http://localhost:5000/register
Method	GET
Attack	
Evidence	
Other Info	This issue still applies to error type pages (401, 403, 500, etc.) as those pages are often still affected by injection issues, in which case there is still concern for browsers sniffing pages away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
Instances	3
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.
Reference	https://learn.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/compatibility/gg622941(v=vs.85) https://owasp.org/www-community/Security_Headers
CWE Id	693
WASC Id	15
Plugin Id	10021
Informational	Information Disclosure - Suspicious Comments
Description	The response appears to contain suspicious comments which may help an attacker.
URL	http://localhost:5000/
Method	GET

Attack	
Evidence	Admin
Other Info	The following pattern was used: \bADMIN\b and was detected in likely comment: "<!-- ===== * Sneat - Bootstrap 5 HTML Admin Template - Pro v1.0.0 =====", see evidence field for the suspicious comment/snippet.
URL	http://localhost:5000/register
Method	GET
Attack	
Evidence	Admin
Other Info	The following pattern was used: \bADMIN\b and was detected in likely comment: "<!-- ===== * Sneat - Bootstrap 5 HTML Admin Template - Pro v1.0.0 =====", see evidence field for the suspicious comment/snippet.
Instances	2
Solution	Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.
Reference	
CWE Id	615
WASC Id	13
Plugin Id	10027

Informational	Session Management Response Identified
Description	The given response has been identified as containing a session management token. The 'Other Info' field contains a set of header tokens that can be used in the Header Based Session Management Method. If the request is in a context which has a Session Management Method set to "Auto-Detect" then this rule will change the session management to use the tokens identified.
URL	http://localhost:5000/
Method	GET
Attack	
Evidence	sessionID
Other Info	cookie:sessionID
URL	http://localhost:5000/robots.txt
Method	GET
Attack	
Evidence	sessionID
Other Info	cookie:sessionID
URL	http://localhost:5000/sitemap.xml
Method	GET
Attack	
Evidence	sessionID
Other Info	cookie:sessionID
URL	http://localhost:5000/
Method	GET

Attack	
Evidence	sessionID
Other Info	cookie:sessionID
Instances	4
Solution	This is an informational alert rather than a vulnerability and so there is nothing to fix.
Reference	https://www.zaproxy.org/docs/desktop/addons/authentication-helper/session-mgmt-id/
CWE Id	
WASC Id	
Plugin Id	10112