ZAP Scanning Report raja

Site: http://127.0.0.1:5000

Generated on Wed, 27 Nov 2024 13:57:06

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	3
Low	5
Informational	4

Alerts

Name	Risk Level	Number of Instances
Absence of Anti-CSRF Tokens	Medium	9
Content Security Policy (CSP) Header Not Set	Medium	16
Missing Anti-clickjacking Header	Medium	11
Application Error Disclosure	Low	2
Cross-Domain JavaScript Source File Inclusion	Low	11
Information Disclosure - Debug Error Messages	Low	2
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	24
X-Content-Type-Options Header Missing	Low	19
Authentication Request Identified	Informational	2
Information Disclosure - Suspicious Comments	Informational	14
Session Management Response Identified	Informational	8
User Controllable HTML Element Attribute (Potential XSS)	Informational	2

Alert Detail

Medium	Absence of Anti-CSRF Tokens
	No Anti-CSRF tokens were found in a HTML submission form. A cross-site request forgery is an attack that involves forcing a victim to send an HTTP request to a target destination without their knowledge or intent in order to perform an action as the victim. The underlying cause is application functionality using predictable URL /form actions in a repeatable way. The nature of the attack is that CSRF exploits the trust that a web site has for a user. By contrast, cross-site scripting (XSS) exploits the trust that a

Description	user has for a web site. Like XSS, CSRF attacks are not necessarily cross-site, but they can be. Cross-site request forgery is also known as CSRF, XSRF, one-click attack, session riding, confused deputy, and sea surf. CSRF attacks are effective in a number of situations, including: * The victim has an active session on the target site. * The victim is authenticated via HTTP auth on the target site. * The victim is on the same local network as the target site. CSRF has primarily been used to perform an action against a target site using the victim's privileges, but recent techniques have been discovered to disclose information by gaining access to the response. The risk of information disclosure is dramatically increased when the target site is vulnerable to XSS, because XSS can be used as a platform for CSRF, allowing the attack to operate within the bounds of the same-origin policy.
URL	http://127.0.0.1:5000/products?item=1
Method	GET
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "quantity"].
URL	http://127.0.0.1:5000/products?item=2
Method	GET
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "quantity"].
URL	http://127.0.0.1:5000/products?item=3
Method	GET
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "quantity"].
URL	http://127.0.0.1:5000/user/create
Method	GET
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "email" "password" "password2" "terms"].
URL	http://127.0.0.1:5000/user/login
Method	GET
Attack	

Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "email" "password" "prev"].
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "quantity"].
URL	http://127.0.0.1:5000/products?item=2
Method	POST
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "quantity"].
URL	http://127.0.0.1:5000/products?item=3
Method	POST
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "quantity"].
URL	http://127.0.0.1:5000/user/login
Method	POST
Attack	
Evidence	<form method="POST">
Other Info	No known Anti-CSRF token [anticsrf, CSRFToken, __RequestVerificationToken, csrfmiddlewaretoken, authenticity_token, OWASP_CSRFTOKEN, anoncsrf, csrf_token, _csrf, _csrfSecret, __csrf_magic, CSRF, _token, _csrf_token] was found in the following HTML form: [Form 1: "email" "password" "prev"].
Instances	9
	Phase: Architecture and Design Use a vetted library or framework that does not allow this weakness to occur or provides constructs that make this weakness easier to avoid. For example, use anti-CSRF packages such as the OWASP CSRFGuard. Phase: Implementation Ensure that your application is free of cross-site scripting issues, because most CSRF defenses can be bypassed using attacker-controlled script. Phase: Architecture and Design

Solution	Generate a unique nonce for each form, place the nonce into the form, and verify the nonce upon receipt of the form. Be sure that the nonce is not predictable (CWE-330). Note that this can be bypassed using XSS. Identify especially dangerous operations. When the user performs a dangerous operation, send a separate confirmation request to ensure that the user intended to perform that operation. Note that this can be bypassed using XSS. Use the ESAPI Session Management control. This control includes a component for CSRF. Do not use the GET method for any request that triggers a state change. Phase: Implementation Check the HTTP Referer header to see if the request originated from an expected page. This could break legitimate functionality, because users or proxies may have disabled sending the Referer for privacy reasons.
Reference	https://cheatsheetseries.owasp.org/cheatsheets/Cross-Site_Request_Forgery_Prevention_Cheat_Sheet.html https://cwe.mitre.org/data/definitions/352.html
CWE Id	352
WASC Id	9
Plugin Id	10202

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://127.0.0.1:5000/
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/basket
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products
Method	GET
Attack	
Evidence	
Other Info	

URL	http://127.0.0.1:5000/products?item=1
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/robots.txt
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/sitemap.xml
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/login
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/terms.html

Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	POST
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	POST
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	POST
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/login
Method	POST
Attack	
Evidence	
Other Info	
Instances	16
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	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	http://127.0.0.1:5000/
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=1
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/login

Method	GET
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	POST
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	POST
Attack	
Evidence	
Other Info	
URL	http://127.0.0.1:5000/user/login
Method	POST
Attack	
Evidence	
Other Info	
Instances	11
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

Low	**Application Error Disclosure**
Description	This page contains an error/warning message that may disclose sensitive information like the location of the file that produced the unhandled exception. This information can be used to launch further attacks against the web application. The alert could be a false positive if the error message is found inside a documentation page.
URL	http://127.0.0.1:5000/basket

Method	GET
Attack	
Evidence	HTTP/1.1 500 INTERNAL SERVER ERROR
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	POST
Attack	
Evidence	HTTP/1.1 500 INTERNAL SERVER ERROR
Other Info	
Instances	2
Solution	Review the source code of this page. Implement custom error pages. Consider implementing a mechanism to provide a unique error reference/identifier to the client (browser) while logging the details on the server side and not exposing them to the user.
Reference	
CWE Id	200
WASC Id	13
Plugin Id	90022

Low	**Cross-Domain JavaScript Source File Inclusion**
Description	The page includes one or more script files from a third-party domain.
URL	http://127.0.0.1:5000/
Method	GET
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/products
Method	GET
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/products?item=1
Method	GET
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	GET
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>

Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	GET
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	GET
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/user/login
Method	GET
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	POST
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	POST
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	
URL	http://127.0.0.1:5000/user/login
Method	POST
Attack	
Evidence	<script src="https://cdn.jsdelivr.net/simplemde/latest/simplemde.min.js"></script>
Other Info	

Instances	11
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	**Information Disclosure - Debug Error Messages**
Description	The response appeared to contain common error messages returned by platforms such as ASP.NET, and Web-servers such as IIS and Apache. You can configure the list of common debug messages.
URL	http://127.0.0.1:5000/basket
Method	GET
Attack	
Evidence	Internal Server Error
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	POST
Attack	
Evidence	Internal Server Error
Other Info	
Instances	2
Solution	Disable debugging messages before pushing to production.
Reference	
CWE Id	200
WASC Id	13
Plugin Id	10023

Low	**Server Leaks Version Information via "Server" HTTP Response Header Field**
Description	The web/application server is leaking version information via the "Server" HTTP response header. Access to such information may facilitate attackers identifying other vulnerabilities your web/application server is subject to.
URL	http://127.0.0.1:5000/
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/basket
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	

URL	http://127.0.0.1:5000/products
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/products?item=1
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/robots.txt
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/sitemap.xml
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/static/custom.css
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/static/markdownEditor/dist/simplemde.min.css

Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/static/markdownEditor/dist/simplemde.min.js
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/static/materialize/css/materialize.min.css
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/static/materialize/js/materialize.min.js
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/uploads/GrayHat.jpg
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/uploads/LinuxBasics.jpg
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/uploads/WebAppHackers.jpg
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	GET
Attack	

Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/user/login
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/user/terms.html
Method	GET
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/products?item=2
Method	POST
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/products?item=3
Method	POST
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/user/create
Method	POST
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6
Other Info	
URL	http://127.0.0.1:5000/user/login
Method	POST
Attack	
Evidence	Werkzeug/3.0.4 Python/3.12.6

Other Info	
Instances	24
Solution	Ensure that your web server, application server, load balancer, etc. is configured to suppress the "Server" header or provide generic details.
Reference	https://httpd.apache.org/docs/current/mod/core.html#servertokens https://learn.microsoft.com/en-us/previous-versions/msp-n-p/ff648552(v=pandp.10) https://www.troyhunt.com/shhh-dont-let-your-response-headers/
CWE Id	200
WASC Id	13
Plugin Id	10036

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://127.0.0.1:5000/
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://127.0.0.1:5000/products
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://127.0.0.1:5000/products?item=1
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://127.0.0.1:5000/products?item=2
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://127.0.0.1:5000/products?item=3
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://127.0.0.1:5000/static/custom.css
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://127.0.0.1:5000/static/markdownEditor/dist/simplemde.min.css
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://127.0.0.1:5000/static/markdownEditor/dist/simplemde.min.js
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://127.0.0.1:5000/static/materialize/css/materialize.min.css
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://127.0.0.1:5000/static/materialize/js/materialize.min.js
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://127.0.0.1:5000/uploads/GrayHat.jpg
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://127.0.0.1:5000/uploads/LinuxBasics.jpg
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://127.0.0.1:5000/uploads/WebAppHackers.jpg
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://127.0.0.1:5000/user/create
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://127.0.0.1:5000/user/login
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://127.0.0.1:5000/products?item=1
Method	POST
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://127.0.0.1:5000/products?item=2

Method	POST
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://127.0.0.1:5000/products?item=3
Method	POST
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://127.0.0.1:5000/user/login
Method	POST
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	19
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	**Authentication Request Identified**
Description	The given request has been identified as an authentication request. The 'Other Info' field contains a set of key=value lines which identify any relevant fields. If the request is in a context which has an Authentication Method set to "Auto-Detect" then this rule will change the authentication to match the request identified.
URL	http://127.0.0.1:5000/user/create
Method	POST
Attack	
Evidence	password
Other Info	userParam=email userValue=zaproxy@example.com passwordParam=password referer=http://127.0.0.1:5000/user/create
URL	http://127.0.0.1:5000/user/login
Method	POST

Attack	
Evidence	password
Other Info	userParam=email userValue=zaproxy@example.com passwordParam=password referer=http://127.0.0.1:5000/user/login
Instances	2
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/auth-req-id/
CWE Id	
WASC Id	
Plugin Id	10111

Informational	**Information Disclosure - Suspicious Comments**
Description	The response appears to contain suspicious comments which may help an attacker. Note: Matches made within script blocks or files are against the entire content not only comments.
URL	http://127.0.0.1:5000/static/markdownEditor/dist/simplemde.min.js
Method	GET
Attack	
Evidence	from
Other Info	The following pattern was used: \bFROM\b and was detected 5 times, the first in the element starting with: "!function(e){if("object"==typeof exports&&"undefined"!=typeof module)module.exports=e();else if("function"==typeof define&&defin", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/static/markdownEditor/dist/simplemde.min.js
Method	GET
Attack	
Evidence	query
Other Info	The following pattern was used: \bQUERY\b and was detected 2 times, the first in the element starting with: "if(a)return a}}},da=e.isModifierKey=function(e){var t="string"==typeof e?e:ol[e.keyCode];return"Ctrl"==t||"Alt"==t||"Shift"==t||", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/static/materialize/js/materialize.min.js
Method	GET
Attack	
Evidence	select
Other Info	The following pattern was used: \bSELECT\b and was detected in the element starting with: "var _get=function t(e,i,n){null===e&&(e=Function.prototype);var s=Object. getOwnPropertyDescriptor(e,i);if(void 0===s){var o=Obje", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/
Method	GET
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/products
Method	GET
Attack	

Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/products?item=1
Method	GET
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/products?item=2
Method	GET
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/products?item=3
Method	GET
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/user/create
Method	GET
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/user/login
Method	GET
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/products?item=2
Method	POST
Attack	
Evidence	admin
Other	The following pattern was used: \bADMIN\b and was detected in the element starting with:

Info	"<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/products?item=3
Method	POST
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
URL	http://127.0.0.1:5000/user/login
Method	POST
Attack	
Evidence	admin
Other Info	The following pattern was used: \bADMIN\b and was detected in the element starting with: "<!-- Dropdown for user admin -->", see evidence field for the suspicious comment/snippet.
Instances	14
Solution	Remove all comments that return information that may help an attacker and fix any underlying problems they refer to.
Reference	
CWE Id	200
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 can be used in the Header Based Session Management Method. If the request is in a context w "Auto-Detect" then this rule will change the session management to use the tokens identified.
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	eyJiYXNrZXQiOnsiMSI6M319.Z0cS6g.XW_v6x0SZHpqQr534WD8uwZM7tc
Other Info	cookie:session
URL	http://127.0.0.1:5000/products?item=2
Method	POST
Attack	
Evidence	eyJiYXNrZXQiOnsiMSI6MywiMiI6M319.Z0cS6w.PjnhipA7H7GEDZdEDfJHhysQJDc
Other Info	cookie:session
URL	http://127.0.0.1:5000/products?item=3
Method	POST
Attack	
Evidence	eyJiYXNrZXQiOnsiMSI6MywiMyI6M319.Z0cS6w.mqPnxm1bjwKwQZo8-lQQJVb0-MU
Other Info	cookie:session
URL	http://127.0.0.1:5000/user/create
Method	POST

Attack	
Evidence	eyJfZmxhc2hlcyI6W3siIHQiOlsibWVzc2FnZSIsIkEgVXNlciB3aXRoIHRoYXQgRW1haWwgRXh Z0cS6w.oAnjko3F-1rMLB79qMop_CxUKbc
Other Info	cookie:session
URL	http://127.0.0.1:5000/user/login
Method	POST
Attack	
Evidence	eyJiYXNrZXQiOnsiMSI6M319.Z0cS6w.wxbq3CpvhBh7NC_Jq6BXOcdyWGY
Other Info	cookie:session
URL	http://127.0.0.1:5000/basket
Method	GET
Attack	
Evidence	eyJiYXNrZXQiOnsiMSI6MywiMyI6M319.Z0cS6w.mqPnxm1bjwKwQZo8-lQQJVb0-MU
Other Info	cookie:session
URL	http://127.0.0.1:5000/user/terms.html
Method	GET
Attack	
Evidence	eyJiYXNrZXQiOnsiMSI6M319.Z0cS6w.wxbq3CpvhBh7NC_Jq6BXOcdyWGY
Other Info	cookie:session
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	eyJiYXNrZXQiOnsiMSI6M319.Z0cS6g.XW_v6x0SZHpqQr534WD8uwZM7tc
Other Info	cookie:session
Instances	8
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

Informational	**User Controllable HTML Element Attribute (Potential XSS)**
Description	This check looks at user-supplied input in query string parameters and POST data to identify where certain HTML attribute values might be controlled. This provides hot-spot detection for XSS (cross-site scripting) that will require further review by a security analyst to determine exploitability.
URL	http://127.0.0.1:5000/products?item=1
Method	GET
Attack	
Evidence	
	User-controlled HTML attribute values were found. Try injecting special characters to see if

Other Info	XSS might be possible. The page at the following URL: http://127.0.0.1:5000/products?item=1 appears to include user input in: a(n) [meta] tag [content] attribute The user input found was: item=1 The user-controlled value was: width=device-width, initial-scale=1
URL	http://127.0.0.1:5000/products?item=1
Method	POST
Attack	
Evidence	
Other Info	User-controlled HTML attribute values were found. Try injecting special characters to see if XSS might be possible. The page at the following URL: http://127.0.0.1:5000/products?item=1 appears to include user input in: a(n) [meta] tag [content] attribute The user input found was: item=1 The user-controlled value was: width=device-width, initial-scale=1
Instances	2
Solution	Validate all input and sanitize output it before writing to any HTML attributes.
Reference	https://cheatsheetseries.owasp.org/cheatsheets/Input_Validation_Cheat_Sheet.html
CWE Id	20
WASC Id	20
Plugin Id	10031