User Alert Report

Sites: http://localhost:8883 http://localhost:8881 http://localhost:8882 https://spocs.getpocket.com http://localhost:8884

Generated on Sat, 30 Mar 2024 11:18:14

ZAP Version: 2.14.0

ZAP is supported by the Crash Override Open Source Fellowship

Summary of Alerts

Risk Level	Number of Alerts
High	1
Medium	1
Low	2
Informational	1
False Positives:	0

Alerts

Name	Risk Level	Number of Instances
SQL Injection	High	5
Hidden File Found	Medium	4
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	10
X-Content-Type-Options Header Missing	Low	10
User Agent Fuzzer	Informational	12

Alert Detail

High	SQL Injection
Description	SQL injection may be possible.
URL	http://localhost:8881/login
Method	POST
Attack	*/*" AND "1"="1" --
Evidence	
Other Info	The page results were successfully manipulated using the boolean conditions [*/*" AND "1"="1" --] and [*/*" AND "1"="2" --] The parameter value being modified was NOT stripped from the HTML output for the purposes of the comparison Data was returned for the original parameter. The vulnerability was detected by successfully restricting the data originally returned, by manipulating the parameter
URL	http://localhost:8881/login
Method	POST

Attack	localhost:8881" AND "1"="1" --
Evidence	
Other Info	The page results were successfully manipulated using the boolean conditions [localhost: 8881" AND "1"="1" --] and [localhost:8881" AND "1"="2" --] The parameter value being modified was NOT stripped from the HTML output for the purposes of the comparison Data was returned for the original parameter. The vulnerability was detected by successfully restricting the data originally returned, by manipulating the parameter
URL	http://localhost:8881/login
Method	POST
Attack	1b4da2dc-c2dc-445f-b578-d1be4b9f0232" OR "1"="1" --
Evidence	
Other Info	The page results were successfully manipulated using the boolean conditions [1b4da2dc-c2dc-445f-b578-d1be4b9f0232" AND "1"="1" --] and [1b4da2dc-c2dc-445f-b578-d1be4b9f0232" OR "1"="1" --] The parameter value being modified was NOT stripped from the HTML output for the purposes of the comparison Data was NOT returned for the original parameter. The vulnerability was detected by successfully retrieving more data than originally returned, by manipulating the parameter
URL	http://localhost:8881/login
Method	POST
Attack	http://localhost:8881/login" OR "1"="1" --
Evidence	
Other Info	The page results were successfully manipulated using the boolean conditions [http://localhost:8881/login" AND "1"="1" --] and [http://localhost:8881/login" OR "1"="1" --] The parameter value being modified was NOT stripped from the HTML output for the purposes of the comparison Data was NOT returned for the original parameter. The vulnerability was detected by successfully retrieving more data than originally returned, by manipulating the parameter
URL	http://localhost:8881/login
Method	POST
Attack	PostmanRuntime/7.37.0 AND 1=1 --
Evidence	
Other Info	The page results were successfully manipulated using the boolean conditions [PostmanRuntime/7.37.0 AND 1=1 --] and [PostmanRuntime/7.37.0 AND 1=2 --] The parameter value being modified was stripped from the HTML output for the purposes of the comparison Data was returned for the original parameter. The vulnerability was detected by successfully restricting the data originally returned, by manipulating the parameter
Instances	5
Solution	Do not trust client side input, even if there is client side validation in place. In general, type check all data on the server side. If the application uses JDBC, use PreparedStatement or CallableStatement, with parameters passed by '?' If the application uses ASP, use ADO Command Objects with strong type checking and parameterized queries. If database Stored Procedures can be used, use them. Do *not* concatenate strings into queries in the stored procedure, or use 'exec', 'exec immediate', or equivalent functionality! Do not create dynamic SQL queries using simple string concatenation. Escape all data received from the client.

	Apply an 'allow list' of allowed characters, or a 'deny list' of disallowed characters in user input. Apply the principle of least privilege by using the least privileged database user possible. In particular, avoid using the 'sa' or 'db-owner' database users. This does not eliminate SQL injection, but minimizes its impact. Grant the minimum database access that is necessary for the application.
Reference	https://cheatsheetseries.owasp.org/cheatsheets/SQL_Injection_Prevention_Cheat_Sheet.html
CWE Id	89
WASC Id	19
Plugin Id	40018

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	http://localhost:8881/._darcs
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	http://localhost:8881/.bzr
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	http://localhost:8881/.hg
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
URL	http://localhost:8881/BitKeeper
Method	GET
Attack	
Evidence	HTTP/1.1 200 OK
Other Info	
Instances	4
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

CWE Id	538
WASC Id	13
Plugin Id	40035

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:8881
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/._darcs
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/.bzr
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/.hg
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/BitKeeper
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	

URL	http://localhost:8881/favicon.ico
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/login
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/robots.txt
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	http://localhost:8881/sitemap.xml
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
Instances	10
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/2012/02/shhh-dont-let-your-response-headers.html
CWE Id	200
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:8881
Method	GET
Attack	
Evidence	
Other	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

Info	away from their actual content type. At "High" threshold this scan rule will not alert on client or server error responses.
URL	http://localhost:8881/
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:8881/._darcs
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:8881/.bzr
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:8881/.hg
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:8881/BitKeeper
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:8881/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:8881/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://localhost:8881/robots.txt
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:8881/sitemap.xml
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	10
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	**User Agent Fuzzer**
Description	Check for differences in response based on fuzzed User Agent (eg. mobile sites, access as a Search Engine Crawler). Compares the response statuscode and the hashcode of the response body with the original response.
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1)
Evidence	
Other Info	
URL	http://localhost:8881/login

Method	POST
Attack	Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.0)
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/4.0 (compatible; MSIE 8.0; Windows NT 6.1)
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Trident/7.0; rv:11.0) like Gecko
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/75.0.3739.0 Safari/537.36 Edg/75.0.109.0
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.124 Safari/537.36
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:93.0) Gecko/20100101 Firefox/91.0
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST

Attack	Mozilla/5.0 (compatible; Yahoo! Slurp; http://help.yahoo.com/help/us/ysearch/slurp)
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/5.0 (iPhone; CPU iPhone OS 8_0_2 like Mac OS X) AppleWebKit/600.1.4 (KHTML, like Gecko) Version/8.0 Mobile/12A366 Safari/600.1.4
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	Mozilla/5.0 (iPhone; U; CPU iPhone OS 3_0 like Mac OS X; en-us) AppleWebKit/528.18 (KHTML, like Gecko) Version/4.0 Mobile/7A341 Safari/528.16
Evidence	
Other Info	
URL	http://localhost:8881/login
Method	POST
Attack	msnbot/1.1 (+http://search.msn.com/msnbot.htm)
Evidence	
Other Info	
Instances	12
Solution	
Reference	https://owasp.org/wstg
CWE Id	
WASC Id	
Plugin Id	10104