

ZAP Scanning Report

Sites: <https://grupo23-lanchonente-pagamento.lemonmoss-42edc919.southcentralus.azurecontainerapps.io> <https://grupo23-lanchonente-pedidos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io> <https://grupo23-lanchonente-produtos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io>

Generated on Thu, 28 Mar 2024 23:33:22

ZAP Version: 2.14.0

ZAP is supported by the [Crash Override Open Source Fellowship](#)

Summary of Alerts

Risk Level	Number of Alerts
High	0
Medium	0
Low	3
Informational	1

Alerts

Name	Risk Level	Number of Instances
Server Leaks Version Information via "Server" HTTP Response Header Field	Low	1
Strict-Transport-Security Header Not Set	Low	4
X-Content-Type-Options Header Missing	Low	4
Re-examine Cache-control Directives	Informational	4

Alert Detail

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	https://grupo23-lanchonente-produtos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/produtos/
Method	GET
Attack	
Evidence	Werkzeug/3.0.1 Python/3.11.8
Other Info	
Instances	1

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	Strict-Transport-Security Header Not Set
Description	HTTP Strict Transport Security (HSTS) is a web security policy mechanism whereby a web server declares that complying user agents (such as a web browser) are to interact with it using only secure HTTPS connections (i.e. HTTP layered over TLS/SSL). HSTS is an IETF standards track protocol and is specified in RFC 6797.
URL	https://grupo23-lanchonente-pagamento.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/pagamentos/pedido/c3e84d9d-58dc-48dc-82ef-3fd85376b742
Method	GET
Attack	
Evidence	
Other Info	
URL	https://grupo23-lanchonente-produtos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/produtos/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://grupo23-lanchonente-pagamento.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/pagamentos/status-atualizacao
Method	POST
Attack	
Evidence	
Other Info	
URL	https://grupo23-lanchonente-pedidos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/api/v1/Pedidos
Method	POST
Attack	
Evidence	
Other Info	
Instances	4
Solution	Ensure that your web server, application server, load balancer, etc. is configured to enforce Strict-Transport-Security.
Reference	https://cheatsheetseries.owasp.org/cheatsheets/HTTP_Strict_Transport_Security_Cheat_Sheet.html https://owasp.org/www-community/Security-Headers

	https://en.wikipedia.org/wiki/HTTP_Strict_Transport_Security https://caniuse.com/stricttransportsecurity https://datatracker.ietf.org/doc/html/rfc6797
CWE Id	319
WASC Id	15
Plugin Id	10035

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	https://grupo23-lanchonente-pagamento.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/pagamentos/pedido/c3e84d9d-58dc-48dc-82ef-3fd85376b742
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	https://grupo23-lanchonente-produtos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/produtos/
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	https://grupo23-lanchonente-pagamento.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/pagamentos/status-atualizacao
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	https://grupo23-lanchonente-pedidos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/api/v1/Pedidos
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	4

Solution	<p>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.</p> <p>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.</p>
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	Re-examine Cache-control Directives
Description	The cache-control header has not been set properly or is missing, allowing the browser and proxies to cache content. For static assets like css, js, or image files this might be intended, however, the resources should be reviewed to ensure that no sensitive content will be cached.
URL	https://grupo23-lanchonente-pagamento.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/pagamentos/pedido/c3e84d9d-58dc-48dc-82ef-3fd85376b742
Method	GET
Attack	
Evidence	
Other Info	
URL	https://grupo23-lanchonente-produtos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/produtos/
Method	GET
Attack	
Evidence	
Other Info	
URL	https://grupo23-lanchonente-pagamento.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/pagamentos/status-atualizacao
Method	POST
Attack	
Evidence	
Other Info	
URL	https://grupo23-lanchonente-pedidos.lemonmoss-42edc919.southcentralus.azurecontainerapps.io/api/v1/Pedidos
Method	POST
Attack	
Evidence	
Other Info	
Instances	4
Solution	For secure content, ensure the cache-control HTTP header is set with "no-cache, no-store, must-revalidate". If an asset should be cached consider setting the directives "public, max-age, immutable".

Reference	https://cheatsheetseries.owasp.org/cheatsheets/Session_Management_Cheat_Sheet.html#web-content-caching https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Cache-Control https://grayduck.mn/2021/09/13/cache-control-recommendations/
CWE Id	525
WASC Id	13
Plugin Id	10015