primeiro-relatorio

Site: https://cyqar4usf1.execute-api.us-east-1.amazonaws.com

Generated on Mon, 22 Jul 2024 22:13:41

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	1
Low	3
Informational	1

Alerts

Name	Risk Level	Number of Instances
Cross-Domain Misconfiguration	Medium	6
Server Leaks Information via "X-Powered-By" HTTP Response Header Field(s)	Low	6
Strict-Transport-Security Header Not Set	Low	7
X-Content-Type-Options Header Missing	Low	4
Re-examine Cache-control Directives	Informational	4

Alert Detail

Medium	Cross-Domain Misconfiguration
Description	Web browser data loading may be possible, due to a Cross Origin Resource Sharing (CORS) misconfiguration on the web server
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae
Method	GET
Attack	
Evidence	access-control-allow-origin: *
Other Info	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8
Method	GET

Attack	
Evidence	access-control-allow-origin: *
Other Info	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order
Method	POST
Attack	
Evidence	access-control-allow-origin: *
Other Info	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae/pay
Method	POST
Attack	
Evidence	access-control-allow-origin: *
Other Info	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8/pay
Method	POST
Attack	
Evidence	access-control-allow-origin: *
Other Info	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/callback
Method	POST
Attack	
Evidence	access-control-allow-origin: *
Other Info	The CORS misconfiguration on the web server permits cross-domain read requests from arbitrary third party domains, using unauthenticated APIs on this domain. Web browser implementations do not permit arbitrary third parties to read the response from authenticated APIs, however. This reduces the risk somewhat. This misconfiguration could be used by an attacker to access data that is available in an unauthenticated manner, but which uses some other form of security, such as IP address white-listing.
Instances	6
	Ensure that sensitive data is not available in an unauthenticated manner (using IP address white-listing, for instance).

Solution	Configure the "Access-Control-Allow-Origin" HTTP header to a more restrictive set of domains, or remove all CORS headers entirely, to allow the web browser to enforce the Same Origin Policy (SOP) in a more restrictive manner.
Reference	https://vulncat.fortify.com/en/detail?id=desc.config.dotnet.html5_overly_permissive_cors_policy
CWE Id	264
WASC Id	14
Plugin Id	10098

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	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8
Method	GET
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order
Method	POST
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae/pay
Method	POST
Attack	
Evidence	X-Powered-By: Express
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8/pay
Method	POST
Attack	
Evidence	X-Powered-By: Express
Other Info	

URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/callback
Method	POST
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/2012/02/shhh-dont-let-your-response-headers.html
CWE Id	200
WASC Id	13
Plugin Id	10037

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://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae
Method	GET
Attack	
Evidence	
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8
Method	GET
Attack	
Evidence	
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/products
Method	GET
Attack	
Evidence	
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order
Method	POST
Attack	
Evidence	
Other	

Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae/pay
Method	POST
Attack	
Evidence	
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8/pay
Method	POST
Attack	
Evidence	
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/callback
Method	POST
Attack	
Evidence	
Other Info	
Instances	7
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://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae
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://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8
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://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/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	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order
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	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	**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://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/92ee36f4-1dba-4533-a02a-1e3c61f0bdae
Method	GET
Attack	
Evidence	
Other Info	

URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order/c37ae30d-0181-4f20-946e-2d4a32f898b8
Method	GET
Attack	
Evidence	
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/products
Method	GET
Attack	
Evidence	
Other Info	
URL	https://cyqar4usf1.execute-api.us-east-1.amazonaws.com/api/order
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