CORS Configuration Report

juice-shop.herokuapp.com_#_cors_2025-09-08_00-16-10

juice-shop.herokuapp.com_#_cors_2025-09-08_00-16-10

Executive Summary

This report summarizes the findings of a CORS (Cross-Origin Resource Sharing) configuration audit performed on https://juice-shop.herokuapp.com/# on 2025-09-08_00-16-10. A total of 3 potential CORS-related vulnerabilities were identified. These vulnerabilities, all classified as Medium risk, could potentially be exploited by malicious actors to perform actions on behalf of legitimate users.

Immediate action is recommended to address these issues to ensure the security and integrity of the application.

Report Details

Target: https://juice-shop.herokuapp.com/#

Generated: 2025-09-08_00-16-10

Total findings: 3 (vulnerabilities: 3) | Elapsed: 4.38s

Detailed Recommendations

1. Wildcard Access-Control-Allow-Origin

Description: The server is configured to allow requests from any origin by using a wildcard ("*") in the Access-Control-Allow-Origin header. This effectively disables CORS protection.

Impact: A malicious website can make requests to the target domain as if it were a legitimate user, potentially stealing sensitive data or performing unauthorized actions.

Recommendation: Replace the wildcard with a specific list of trusted origins. Regularly review and update this list as needed. Only allow origins that genuinely require access to the resource.

2. Overly Permissive Methods

Description: The Access-Control-Allow-Methods header includes a wide range of HTTP methods, potentially exposing the server to unexpected operations.

Impact: Allowing unnecessary methods increases the attack surface. For example, if a resource only requires GET requests, allowing PUT, POST, or DELETE could lead to unintended data modification or deletion.

Recommendation: Restrict the allowed methods to only those that are strictly required by the resource. Avoid including potentially dangerous methods if they are not needed.

3. Sensitive Headers Exposed

Description: The Access-Control-Allow-Headers header includes sensitive headers such as "Authorization" and "X-Api-Key".

Impact: Exposing these headers could allow a malicious origin to access or manipulate user authentication tokens or API keys, leading to unauthorized access and data breaches.

Recommendation: Remove sensitive headers from the Access-Control-Allow-Headers list. Only allow headers that are necessary for legitimate cross-origin requests.