Security Audit: https://juice-shop-vsq1.onrender.com

ID: 16b16ae3-5283-45e3-8b64-a4dbf2c13cf2 | **Date**: 2026-02-21 | **Findings**: 63

Sev	Crit	High	Med	Low	Info
Amt	0	7	19	37	0

Findings Summary

#	Sev	Type	Endpoint
1	High	Broken Access Control	`https://juice-shop-vsq1.onrender.com/admin`
2	High	Broken Access Control	`https://juice-shop-vsq1.onrender.com/administrator`
3	High	Broken Access Control	`https://juice-shop-vsq1.onrender.com/admin/dashboa...`
4	High	Broken Access Control	`https://juice-shop-vsq1.onrender.com/panel`
5	High	Broken Access Control	`https://juice-shop-vsq1.onrender.com/manage`
6	High	Broken Access Control	`https://juice-shop-vsq1.onrender.com/backend`
7	High	Cryptographic Failure	`https://juice-shop-vsq1.onrender.com`
8	Med	Information Disclosure	`https://juice-shop-vsq1.onrender.com/.env`
9	Med	Information Disclosure	`https://juice-shop-vsq1.onrender.com/admin`
10	Med	Information Disclosure	`https://juice-shop-vsq1.onrender.com/.git/HEAD`
11	Med	Information Disclosure	`https://juice-shop-vsq1.onrender.com`
12	Med	CORS Misconfiguration	`https://juice-shop-vsq1.onrender.com`
13	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/debug`
14	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/trace`
15	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/actuator`
16	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/phpinfo.php`
17	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/server-status`
18	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/console`
19	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/shell`
20	Med	Debug Endpoint	`https://juice-shop-vsq1.onrender.com/admin/config`
21	Med	CORS Misconfiguration	`https://juice-shop-vsq1.onrender.com`
22	Med	Vulnerable Component	`https://juice-shop-vsq1.onrender.com/.env`
23	Med	Vulnerable Component	`https://juice-shop-vsq1.onrender.com/admin`
24	Med	Vulnerable Component	`https://juice-shop-vsq1.onrender.com/login`
25	Med	Vulnerable Component	`https://juice-shop-vsq1.onrender.com/.git/HEAD`
26	Med	Vulnerable Component	`https://juice-shop-vsq1.onrender.com/swagger.json`
27	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
28	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
29	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
30	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
31	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
32	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
33	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
34	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
35	Low	Missing Security Header	`https://juice-shop-vsq1.onrender.com`
36	Low	Missing SRI	`https://juice-shop-vsq1.onrender.com/.env`
37	Low	Missing SRI	`https://juice-shop-vsq1.onrender.com/.env`
38	Low	Missing SRI	`https://juice-shop-vsq1.onrender.com/.env`
39	Low	Missing SRI	`https://juice-shop-vsq1.onrender.com/admin`

#	Sev	Type	Endpoint
40	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/admin`
41	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/admin`
42	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/login`
43	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/login`
44	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/login`
45	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/.git/HEAD`
46	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/.git/HEAD`
47	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/.git/HEAD`
48	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/swagger.json`
49	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/swagger.json`
50	🔵	Missing SRI	`https://juice-shop-vsq1.onrender.com/swagger.json`
51	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/logs`
52	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/log`
53	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/error.log`
54	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/debug.log`
55	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/app.log`
56	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/access.log`
57	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/storage/logs/...`
58	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/logs/error.lo...`
59	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/npm-debug.log`
60	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/yarn-error.lo...`
61	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/.logs`
62	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com/.log`
63	🔵	Logging Failure	`https://juice-shop-vsq1.onrender.com`

🔍 Technical Evidence & Remediation

1. Broken Access Control [🟠 High]

Description: Admin path "/admin" accessible without authentication

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/admin`	`HTTP 200 response on /admin`
`https://juice-shop-vsq1.onrender.com/administrator`	`HTTP 200 response on /administrator`
`https://juice-shop-vsq1.onrender.com/admin/dashboard`	`HTTP 200 response on /admin/dashboard`
`https://juice-shop-vsq1.onrender.com/panel`	`HTTP 200 response on /panel`
`https://juice-shop-vsq1.onrender.com/manage`	`HTTP 200 response on /manage`
`https://juice-shop-vsq1.onrender.com/backend`	`HTTP 200 response on /backend`

Fix & Simulation: Remediation: Restrict access to `/admin` endpoint by implementing role-based access control (RBAC) checks.

How it was confirmed: Unauthorized access to `/admin` endpoint returned HTTP 200 response.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com/admin
```

Code Fix (Node.js/Express):

```
app.get('/admin', ensureAuthenticated, ensureAdminRole, (req, res) => {
  // Admin-only content
});
```

Middleware:

```
function ensureAdminRole(req, res, next) {
  if (req.user && req.user.role === 'admin') {
    return next();
  }
  res.status(403).send('Forbidden');
}
```

---

2. Cryptographic Failure (HSTS) [🟠 High]

Description: Missing HTTP Strict Transport Security (HSTS) header

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`No Strict-Transport-Security header in response`

Fix & Simulation: Remediation: Add HSTS header with max-age and includeSubDomains directives.

How it was confirmed: `curl -I https://juice-shop-vsq1.onrender.com` showed no `Strict-Transport-Security` header.

How to simulate: `curl -I https://juice-shop-vsq1.onrender.com` should return `Strict-Transport-Security: max-age=31536000; includeSubDomains`.

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader("Strict-Transport-Security", "max-age=31536000; includeSubDomains");
  next();
});
```

---

3. Information Disclosure [🟡 Medium]

Description: Sensitive path /.env is accessible (HTTP 200)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/.env`	`Status: 200`
`https://juice-shop-vsq1.onrender.com/admin`	`Status: 200`
`https://juice-shop-vsq1.onrender.com/.git/HEAD`	`Status: 200`

Fix & Simulation: Remediation: Restrict access to `.env` file by adding a rule to your web server configuration to deny access to this file.

How it was confirmed: Accessing `https://juice-shop-vsq1.onrender.com/.env` returned a 200 status code, indicating the file was served.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com/.env
```

Code Fix (for Node.js/Express):

```
app.get('/\.env', (req, res) => {
  res.status(403).send('Access forbidden');
});
```

---

4. Information Disclosure (server) [🟡 Medium]

Description: Server information disclosure via server header

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`server: cloudflare`

Fix & Simulation: Remediation: Implement server-side request validation and disable server info headers.

How it was confirmed: Server info was disclosed in the response headers.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com
```

Code fix (Node.js/Express):

```
app.disable('x-powered-by');
app.use((req, res, next) => {
  res.removeHeader('Server');
  next();
});
```

---

5. CORS Misconfiguration (CORS) [🟡 Medium]

Description: Wildcard CORS origin

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`*`

Fix & Simulation: Remediation: Set `Access-Control-Allow-Origin` to `null` or specific domains, not `*`.

How it was confirmed: Curl request to the endpoint returned `Access-Control-Allow-Origin: *` in the response headers.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com
```

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader('Access-Control-Allow-Origin', 'https://yourdomain.com');
  next();
});
```

---

6. Debug Endpoint [🟡 Medium]

Description: Debug/admin endpoint accessible: /debug

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/debug`	`HTTP 200 on /debug`
`https://juice-shop-vsq1.onrender.com/trace`	`HTTP 200 on /trace`
`https://juice-shop-vsq1.onrender.com/actuator`	`HTTP 200 on /actuator`
`https://juice-shop-vsq1.onrender.com/phpinfo.php`	`HTTP 200 on /phpinfo.php`
`https://juice-shop-vsq1.onrender.com/server-status`	`HTTP 200 on /server-status`
`https://juice-shop-vsq1.onrender.com/console`	`HTTP 200 on /console`
...	*+ 2 more*

Fix & Simulation: Remediation: Remove or restrict access to the `/debug` endpoint.

How it was confirmed: `curl -I https://juice-shop-vsq1.onrender.com/debug` returned HTTP 200.

How to simulate: `curl -I https://juice-shop-vsq1.onrender.com/debug`

Code fix (Node.js/Express):

```
// Remove or comment out the following line in your route configuration:
app.get('/debug', debugEndpoint); // Remove this line or restrict access
```

---

7. CORS Misconfiguration (Access-Control-Allow-Origin) [🟡 Medium]

Description: CORS policy reflects arbitrary origin or uses wildcard

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Access-Control-Allow-Origin: * for Origin: https://evil-attacker.com`

Fix & Simulation: Remediation: Restrict `Access-Control-Allow-Origin` to specific trusted domains.

How it was confirmed: Curl request to the endpoint with `Origin: https://evil-attacker.com` header returned `Access-Control-Allow-Origin: *`.

How to simulate:

```
curl -I -H "Origin: https://evil-attacker.com" https://juice-shop-vsq1.onrender.com
```

Code fix (Node.js/Express):

```
app.use((req, res, next) => {
  const allowedOrigins = ['https://trusted-domain.com'];
  const origin = req.headers.origin;
  if (allowedOrigins.includes(origin)) {
    res.setHeader('Access-Control-Allow-Origin', origin);
  }
  next();
});
```

---

8. Vulnerable Component (jQuery) [🟡 Medium]

Description: jQuery v2.2.4 detected in page - jQuery < 3.0 has XSS vulnerabilities (CVE-2020-11022)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/.env`	`Pattern matched in response body`
`https://juice-shop-vsq1.onrender.com/admin`	`Pattern matched in response body`
`https://juice-shop-vsq1.onrender.com/login`	`Pattern matched in response body`
`https://juice-shop-vsq1.onrender.com/.git/HEAD`	`Pattern matched in response body`
`https://juice-shop-vsq1.onrender.com/swagger.json`	`Pattern matched in response body`

Fix & Simulation: Remediation: Upgrade jQuery to version 3.5.0 or later.

How it was confirmed: Pattern "jquery" matched in response body with version < 3.5.0.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com/.env | grep -i "jquery"
```

Code Fix (JavaScript):

```
// Update jQuery reference in your HTML or JS files
<script src="https://code.jquery.com/jquery-3.5.1.min.js"></script>
```

---

9. Missing Security Header (strict-transport-security) [🔵 Low]

Description: Missing security header: HSTS (HTTP Strict Transport Security)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "strict-transport-security" not present in response`

Fix & Simulation: Remediation: Add `Strict-Transport-Security: max-age=31536000; includeSubDomains; preload` header.

How it was confirmed: Checked response headers using `curl -I https://juice-shop-vsq1.onrender.com`.

How to simulate: `curl -I https://juice-shop-vsq1.onrender.com | grep "Strict-Transport-Security"`

Code Fix: For Node.js/Express:

```
app.use((req, res, next) => {
  res.setHeader("Strict-Transport-Security", "max-age=31536000; includeSubDomains; preload");
  next();
});
```

---

10. Missing Security Header (content-security-policy) [🔵 Low]

Description: Missing security header: CSP (Content Security Policy)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "content-security-policy" not present in response`

Fix & Simulation: Remediation: Add Content-Security-Policy header to server responses.

How it was confirmed: `curl -I https://juice-shop-vsq1.onrender.com | grep -i "content-security-policy"`

How to simulate: `curl -H "Content-Security-Policy: default-src 'self'" https://juice-shop-vsq1.onrender.com`

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader("Content-Security-Policy", "default-src 'self'");
  next();
});
```

---

11. Missing Security Header (x-xss-protection) [🔵 Low]

Description: Missing security header: X-XSS-Protection (XSS filter)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "x-xss-protection" not present in response`

Fix & Simulation: Remediation: Add `X-XSS-Protection: 1; mode=block` header.

How it was confirmed: `curl -I https://juice-shop-vsq1.onrender.com` did not return `x-xss-protection` header.

How to simulate: `curl -H "X-XSS-Protection: 1; mode=block" -I https://juice-shop-vsq1.onrender.com`

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader("X-XSS-Protection", "1; mode=block");
  next();
});
```

---

12. Missing Security Header (referrer-policy) [🔵 Low]

Description: Missing security header: Referrer-Policy (Controls referrer information)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "referrer-policy" not present in response`

Fix & Simulation: Remediation: Add `Referrer-Policy` header to server responses.

How it was confirmed: Checked response headers using `curl -I https://juice-shop-vsq1.onrender.com`.

How to simulate:

`curl -I -H "Referer: https://example.com" https://juice-shop-vsq1.onrender.com`

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader('Referrer-Policy', 'strict-origin-when-cross-origin');
  next();
});
```

---

13. Missing Security Header (permissions-policy) [🔵 Low]

Description: Missing security header: Permissions-Policy (Controls browser features)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "permissions-policy" not present in response`

Fix & Simulation: Remediation: Add the `Permissions-Policy` header to restrict browser features.

How it was confirmed: `curl -I https://juice-shop-vsq1.onrender.com` did not return a `Permissions-Policy` header.

How to simulate: `curl -H "Permissions-Policy: geolocation=(), camera=(), microphone=()" https://juice-shop-vsq1.onrender.com`

Code Fix: For Node.js/Express:

```
app.use((req, res, next) => {
  res.setHeader("Permissions-Policy", "geolocation=(), camera=(), microphone=()");
  next();
});
```

---

14. Missing Security Header (cross-origin-embedder-policy) [🔵 Low]

Description: Missing security header: COEP (Cross-Origin Embedder Policy)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "cross-origin-embedder-policy" not present in response`

Fix & Simulation: Remediation: Add `Cross-Origin-Embedder-Policy: require-corp` header.

How it was confirmed: Checked response headers with `curl -I https://juice-shop-vsq1.onrender.com`.

How to simulate: `curl -I -H "Origin: https://evil.com" https://juice-shop-vsq1.onrender.com`.

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader("Cross-Origin-Embedder-Policy", "require-corp");
  next();
});
```

---

15. Missing Security Header (cross-origin-opener-policy) [🔵 Low]

Description: Missing security header: COOP (Cross-Origin Opener Policy)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "cross-origin-opener-policy" not present in response`

Fix & Simulation: Remediation: Add `cross-origin-opener-policy: same-origin` header.

How it was confirmed: Checked response headers using `curl -I https://juice-shop-vsq1.onrender.com`.

How to simulate: `curl -I -H "Origin: https://evil.com" https://juice-shop-vsq1.onrender.com`.

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader("cross-origin-opener-policy", "same-origin");
  next();
});
```

---

16. Missing Security Header (cross-origin-resource-policy) [🔵 Low]

Description: Missing security header: CORP (Cross-Origin Resource Policy)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "cross-origin-resource-policy" not present in response`

Fix & Simulation: Remediation: Add the `Cross-Origin-Resource-Policy` header to server responses.

How it was confirmed: `curl -I https://juice-shop-vsq1.onrender.com` did not return a `Cross-Origin-Resource-Policy` header.

How to simulate: `curl -I -H "Origin: https://example.com" https://juice-shop-vsq1.onrender.com`

Code Fix: For a Node.js/Express application, add the following middleware:

```
app.use((req, res, next) => {
  res.setHeader('Cross-Origin-Resource-Policy', 'same-site');
  next();
});
```

---

17. Missing Security Header (x-permitted-cross-domain-policies) [🔵 Low]

Description: Missing security header: X-Permitted-Cross-Domain-Policies (Restricts Flash/PDF cross-domain policy files)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`Header "x-permitted-cross-domain-policies" not present in response`

Fix & Simulation: Remediation: Add the `x-permitted-cross-domain-policies` header to the server's response.

How it was confirmed: Checked response headers using `curl -I https://juice-shop-vsq1.onrender.com`.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com | grep "x-permitted-cross-domain-policies"
```

Code Fix (Node.js/Express):

```
app.use((req, res, next) => {
  res.setHeader("x-permitted-cross-domain-policies", "none");
  next();
});
```

---

18. Missing SRI (//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js) [🔵 Low]

Description: External script loaded without Subresource Integrity (SRI)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/.env`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/cookiecons...`
`https://juice-shop-vsq1.onrender.com/admin`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/cookiecons...`
`https://juice-shop-vsq1.onrender.com/login`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/cookiecons...`
`https://juice-shop-vsq1.onrender.com/.git/HEAD`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/cookiecons...`
`https://juice-shop-vsq1.onrender.com/swagger.json`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/cookiecons...`

Fix & Simulation: Remediation: Add SRI to the script tag: `<script src="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js" integrity="sha384-..." crossorigin="anonymous"></script>`

How it was confirmed: Inspected the source code and found the script tag without SRI.

How to simulate:

```
curl -I https://cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js
```

Code Fix:

```
<script src="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.js" integrity="sha384-..." crossorigin="anonymous"></script>
```

---

19. Missing SRI (//cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js) [🔵 Low]

Description: External script loaded without Subresource Integrity (SRI)

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/.env`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2...`
`https://juice-shop-vsq1.onrender.com/admin`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2...`
`https://juice-shop-vsq1.onrender.com/login`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2...`
`https://juice-shop-vsq1.onrender.com/.git/HEAD`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2...`
`https://juice-shop-vsq1.onrender.com/swagger.json`	`Insecure tag: <script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2...`

Fix & Simulation: Remediation: Add SRI to the script tag: `<script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js" integrity="sha384-KyZXEAg3QhqLMpG8r+Knujsl5/84L17twkPUJLp1QGdbuzwx9D9dF0gg62V0I4f7Z" crossorigin="anonymous"></script>` .

How it was confirmed: Inspected the source code and found the script tag without SRI.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com/.env | grep -i "Content-Security-Policy"
```

Code Fix:

```
// Update the script tag in your HTML file
<script src="//cdnjs.cloudflare.com/ajax/libs/jquery/2.2.4/jquery.min.js" integrity="sha384-KyZXEAg3QhqLMpG8r+Knujsl5/84L17twkPUJLp1QGdbuzwx9D9dF0gg62V0I4f7Z" crc
```

---

20. Missing SRI (//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.css) [🔵 Low]

Description: External stylesheet loaded without SRI

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/.env`	`<link href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cook...`
`https://juice-shop-vsq1.onrender.com/admin`	`<link href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cook...`
`https://juice-shop-vsq1.onrender.com/login`	`<link href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cook...`
`https://juice-shop-vsq1.onrender.com/.git/HEAD`	`<link href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cook...`
`https://juice-shop-vsq1.onrender.com/swagger.json`	`<link href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cook...`

Fix & Simulation: Remediation: Add SRI hash to the script tag.

How it was confirmed: Inspected the HTML source and found the missing integrity attribute in the cookieconsent2 CSS link.

How to simulate:

```
curl -I https://cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.css
```

Code Fix:

```
<link href="//cdnjs.cloudflare.com/ajax/libs/cookieconsent2/3.1.0/cookieconsent.min.css"
  integrity="sha384-..." crossorigin="anonymous">
```

---

21. Logging Failure [🔵 Low]

Description: Potential log file accessible at /logs

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/logs`	`HTTP 200 with 75055 bytes of content`

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com/log`	`HTTP 200 with 75055 bytes of content`
`https://juice-shop-vsq1.onrender.com/error.log`	`HTTP 200 with 75055 bytes of content`
`https://juice-shop-vsq1.onrender.com/debug.log`	`HTTP 200 with 75055 bytes of content`
`https://juice-shop-vsq1.onrender.com/app.log`	`HTTP 200 with 75055 bytes of content`
`https://juice-shop-vsq1.onrender.com/access.log`	`HTTP 200 with 75055 bytes of content`
...	*+ 6 more*

Fix & Simulation: Remediation: Implement proper logging for security-relevant events.

How it was confirmed: Accessed `/logs` endpoint and received a non-empty response.

How to simulate:

```
curl -I https://juice-shop-vsq1.onrender.com/logs
```

Code Fix (Node.js/Express):

```
// Add this middleware to your Express app
app.use((req, res, next) => {
  const logData = {
    method: req.method,
    path: req.path,
    timestamp: new Date().toISOString(),
    // Add other relevant data
  };
  console.log(logData); // Or use a proper logging library
  next();
});
```

---

22. Logging Failure (Rate Limiting) [🔵 Low]

Description: No rate limiting detected on rapid requests

Endpoint	Proof / Evidence
`https://juice-shop-vsq1.onrender.com`	`5 rapid requests all returned non-429 responses`

Fix & Simulation: Remediation: Implement rate limiting middleware to restrict requests.

How it was confirmed: 5 rapid requests via curl returned non-429 responses.

How to simulate:

```
for i in {1..5}; do curl -s -o /dev/null -w "%{http_code}" https://juice-shop-vsq1.onrender.com; done
```

Code Fix (Express.js):

```
const rateLimit = require('express-rate-limit');
app.use(rateLimit({
  windowMs: 15 * 60 * 1000, // 15 minutes
  max: 100 // limit each IP to 100 requests per windowMs
}));
```

💡 Key Recommendations

- 7 high-severity issues should be resolved before deployment.
- Add all recommended security headers (CSP, HSTS, X-Frame-Options, etc.).

Generated by VulnSight-AI — Agentic Security Auditor