

2 October 2026

Penetration Test Report (Sample)

Security assessment of the shipper portal and API

Prepared for

Northwind Freight, Inc.

Prepared by

Zonix Labs, Inc.

Lead Penetration Tester, OSCP

Confidential. Prepared for Northwind Freight, Inc.. This document contains proprietary information and may be shared with Northwind Freight, Inc.'s auditors under non-disclosure agreement.

1. Introduction

1.1 Confidentiality statement

This document is the exclusive property of Northwind Freight, Inc. and Zonix Labs, Inc. It contains proprietary and confidential information. Duplication, redistribution or use, in whole or in part, in any form requires the consent of Northwind Freight, Inc. Northwind Freight, Inc. may share this document with its auditors under non-disclosure agreement to demonstrate penetration testing compliance.

1.2 Version control

VERSION	DESCRIPTION	DATE	AUTHOR
0.1	Draft issued for client review	14 August 2026	Lead Penetration Tester, OSCP
1.0	Final	18 August 2026	Lead Penetration Tester, OSCP
1.1	Reissued following remediation retest	02 October 2026	Lead Penetration Tester, OSCP

1.3 The team

ROLE	QUALIFICATIONS
Lead Penetration Tester	Offensive Security Certified Professional (OSCP)

Zonix Labs, Inc. is organisationally independent of Northwind Freight, Inc. and was not involved in the development or operation of the systems tested.

1.4 Scope

ASSET	IDENTIFIER	ROLES TESTED	TENANTS
Shipper portal	https://app.northwind.example	Admin, Dispatcher, Viewer	2
Public API	https://api.northwind.example	Admin, Dispatcher, Viewer	2
Partner webhook receiver	https://hooks.northwind.example	unauthenticated	n/a

Out of scope: the marketing site (`www.northwind.example`), Northwind's corporate network and endpoints, physical premises, personnel, and all third-party SaaS the application integrates with, including the payment processor and the email provider.

Testing mode: grey box. Northwind supplied two accounts for each of the three roles, split across two separate tenants, which is the precondition for testing cross-tenant access.

1.5 Limitations

- The partner webhook receiver enforces HMAC request signing. Northwind supplied a signing key for the test tenant only, so webhook testing was limited to that tenant's signature scope.
- Rate limiting on `POST /v2/auth/login` was left enabled at Northwind's request. Credential stuffing resistance was assessed by configuration review rather than by live testing, and is reported at lower confidence as a result.
- The scheduled-report export job runs nightly. One finding (PT-7) was confirmed across a single execution window rather than repeatedly.

No other material limitations were encountered. Credentials were available on day one and the Tester's source addresses were allowlisted before the window opened.

1.6 Timeline

	DATE
Testing start	11 August 2026
Testing end	14 August 2026
Draft report issued	14 August 2026
Final report issued	18 August 2026
Remediation retest	01 October 2026
Report reissued (v1.1)	02 October 2026

1.7 Disclaimer

This test is a point-in-time assessment and the environment may have changed since it was performed. There is no guarantee that all possible security issues have been identified, and new vulnerabilities may have emerged since testing concluded. This report is a guiding document and not a warranty that it represents the full set of risks facing the systems assessed.

2. Executive summary

Northwind Freight commissioned this assessment to satisfy the penetration testing expectation for its SOC 2 Type II audit period ending 31 December 2026, and to answer the security questionnaires it receives from enterprise shippers.

Testing covered the shipper portal, its public API and the partner webhook receiver, across three user roles in two separate tenants, over four days.

Eight findings were identified: one critical, three high, two medium, two low.

Three are worth leadership attention:

Any dispatcher could read another customer's shipment records. The API returned shipment documents to any authenticated user who knew or guessed the shipment identifier, without checking which company that shipment belonged to. Identifiers are sequential, so an attacker with one valid account could enumerate the entire customer base. For a logistics platform this exposes commercial terms, delivery addresses and consignee contact details belonging to companies that are not the attacker's own.

A viewer could promote themselves to administrator. The role field was accepted from the client when updating a user profile, and the server did not verify that the requester was entitled to set it. This turns the lowest-privilege account into a full administrator in one request.

Invoice totals could be altered after approval. The approval workflow re-read the line items from the request body rather than from the stored record, so a dispatcher could approve an invoice at one value and have a different value recorded.

The first two are access-control failures of the same kind: the application establishes *who you are* reliably, but does not consistently check *what you are allowed to reach*. We would prioritise a review of authorisation at the API layer over any other remediation work.

As of the retest on 1 October 2026, **all critical and high-severity findings are resolved**. Two low-severity findings remain open and have been accepted as risk by Northwind.

3. Findings

3.1 Severity distribution

CRITICAL	HIGH	MEDIUM	LOW	INFORMATIONAL
1	3	2	2	0

The critical finding permits cross-tenant data access from any authenticated account. The high findings comprise a privilege escalation, a server-side injection, and a business-logic flaw that affects financial records. Medium findings relate to session handling and to information disclosure through an export path. The low findings are configuration issues with limited direct impact.

3.2 Findings summary

ID	TITLE	SEVERITY	STATE
PT-1	Cross-tenant IDOR on shipment documents exposes all customers' records	CRITICAL	Fixed
PT-2	Mass assignment on profile update allows privilege escalation to Admin	HIGH	Fixed
PT-3	Blind SQL injection in the shipment search filter	HIGH	Fixed
PT-4	Invoice totals recalculated from client input during approval	HIGH	Fixed
PT-5	Session not invalidated on password change	MEDIUM	Fixed
PT-6	Scheduled export delivers full dataset regardless of role scope	MEDIUM	Fixed
PT-7	Webhook replay accepted within a 24-hour signature window	LOW	Risk Accepted
PT-8	Security headers absent on API responses	LOW	Risk Accepted

3.3 Detailed findings

3.3.1 PT-1 - Cross-tenant IDOR on shipment documents exposes all customers' records

Severity: CRITICAL · **State:** Fixed · **Identified:** 11 August 2026 · **Resolved:** 24 September 2026

Severity rationale: rated Critical although its CVSS score of 7.7 falls in the High band. Any single low-privilege account reads every other customer's shipment records, and Northwind's customers are frequently each other's competitors.

Affected asset: GET `https://api.northwind.example/v2/shipments/{id}/documents` Reachable by any authenticated user, including the Viewer role.

Description

The endpoint resolves a shipment by its numeric identifier and returns the attached documents. It verifies that the caller holds a valid session, but never checks that the shipment belongs to the caller's tenant. Shipment identifiers are sequential integers issued globally rather than per tenant, so they are trivially enumerable.

Using a Viewer account in tenant A, documents belonging to tenant B were retrieved by incrementing the identifier. Fifty consecutive identifiers were sampled; forty-three returned documents, of which thirty-one belonged to tenants other than the caller's.

Business impact

Any customer with a single low-privilege account can read the shipment records of every other customer on the platform. The documents include bills of lading, commercial invoices and delivery notes, which together disclose negotiated freight rates, consignee names and addresses, and shipment contents. For a platform whose customers are frequently competitors, this is a direct commercial confidentiality breach rather than only a privacy one, and it is reachable without any privilege escalation. Depending on the personal data in consignee records it may also trigger notification obligations under GDPR Article 33.

Exploitation requires no special tooling, only a valid account and an HTTP client.

Reproduction steps

1. Authenticate as `viewer@tenant-a.example` and capture the bearer token.
2. Request a shipment known to belong to the caller's own tenant to confirm the baseline:

```
GET /v2/shipments/48812/documents HTTP/1.1
Host: api.northwind.example
Authorization: Bearer <tenant A viewer token>
```

1. Decrement the identifier and repeat with the same token:

```
GET /v2/shipments/48811/documents HTTP/1.1
Host: api.northwind.example
Authorization: Bearer <tenant A viewer token>
```

1. Observe `200 OK` with a document set whose `tenant_id` differs from the caller's:

```
HTTP/1.1 200 OK
Content-Type: application/json

{"shipment_id":48811,"tenant_id":"t_9f32c1","documents":[
  {"type":"bill_of_lading","url":"https://cdn.northwind.example/d/48811/bol.pdf"},
  {"type":"commercial_invoice","url":"https://cdn.northwind.example/d/48811/ci.pdf"}]}
```

The `tenant_id` in the response is sufficient to confirm cross-tenant access without retrieving the documents themselves.

Evidence

Request and response pairs for three cross-tenant identifiers are retained in the engagement evidence set. Document contents were not downloaded beyond the first page of one file, which was used to confirm the records were genuine, and that capture is redacted.

Remediation

Scope the lookup to the caller's tenant in the query itself rather than filtering after retrieval:

```
SELECT * FROM shipments WHERE id = :id AND tenant_id = :caller_tenant_id
```

Return `404` rather than `403` when the row is absent, so the endpoint does not confirm that an identifier exists in another tenant. Apply the same scoping to the other eleven endpoints that accept a shipment identifier; they were not individually tested but share the controller.

Consider replacing sequential identifiers with UUIDs. That is defence in depth rather than a fix, and should not be done instead of the authorisation check.

References

CVSS v3.1 7.7 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:C/C:H/I:N/A:N) · WSTG-ATHZ-04 · CWE-639

Retest, 1 October 2026: Resolved. Tenant scoping is enforced in the query and the endpoint now returns 404 for identifiers outside the caller's tenant. Verified across twenty identifiers in both directions between the two test tenants. The sibling endpoints were also corrected.

3.3.2 PT-2 - Mass assignment on profile update allows privilege escalation to Admin

Severity: HIGH · **State:** Fixed · **Identified:** 11 August 2026 · **Resolved:** 22 September 2026

Affected asset: PATCH `https://api.northwind.example/v2/users/me` Reachable by any authenticated user.

Description

The profile update handler binds the request body directly onto the user model. The `role` attribute is part of that model and is not excluded from binding, and the handler does not check whether the caller is permitted to change it. A Viewer account was promoted to `admin` in a single request and immediately gained access to the tenant administration console.

Business impact

The lowest-privilege role becomes the highest in one request. An attacker needs only a valid account of any kind, which in Northwind's model includes accounts created through the customer self-service invite flow. Combined with PT-1 the impact compounds: an attacker can escalate within their own tenant and then read every other tenant's records with administrator scope.

Reproduction steps

1. Authenticate as `viewer@tenant-a.example`.
2. Confirm the current role:

```
GET /v2/users/me HTTP/1.1
Authorization: Bearer <viewer token>

HTTP/1.1 200 OK
{"id":"u_41d9","email":"viewer@tenant-a.example","role":"viewer"}
```

1. Send a profile update including the `role` attribute:

```
PATCH /v2/users/me HTTP/1.1
Authorization: Bearer <viewer token>
Content-Type: application/json

{"display_name":"QA Test","role":"admin"}
```

1. The response reflects the new role:

```
HTTP/1.1 200 OK
{"id":"u_41d9","email":"viewer@tenant-a.example","role":"admin"}
```

1. Request a token refresh and confirm that `GET /v2/admin/users` now returns 200 rather than 403.

Remediation

Bind only an explicit allow-list of attributes a user may change about themselves: `display_name` , `phone` , `timezone` , `notification_preferences` . Role changes belong on a separate administrative endpoint with its own authorisation check. Audit-log every role change with the acting principal.

Northwind's framework supports strong-parameter filtering; the other seven update handlers should be reviewed for the same pattern.

References

CVSS v3.1 8.8 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:H) · WSTG-INJT-20 · CWE-915

Retest, 1 October 2026: Resolved. The handler now binds an explicit attribute list and rejects unknown fields with 422 . Role changes were moved to `PUT /v2/admin/users/{id}/role` , which enforces an administrator check. Verified from both Viewer and Dispatcher accounts.

3.3.3 PT-3 - Blind SQL injection in the shipment search filter

Severity: HIGH · **State:** Fixed · **Identified:** 12 August 2026 · **Resolved:** 22 September 2026

Affected asset: `GET https://api.northwind.example/v2/shipments?sort={field}` Reachable by any authenticated user.

Description

The `sort` parameter is interpolated into the `ORDER BY` clause after a filter that strips quote characters. Parameterisation cannot be applied to an identifier position, and the filter does not prevent the injection of a subquery. Boolean and timing payloads both succeeded; a payload using `CASE WHEN ... THEN pg_sleep(5) END` produced a reproducible five-second delay while the control payload returned in under 200ms.

Data was not extracted beyond confirming the database version string, which was sufficient to demonstrate execution.

Business impact

Arbitrary query execution against the application database, reachable by any authenticated user. Northwind's schema holds customer records, shipment data and invoice history in the same database, so a successful attacker can read across all tenants regardless of the application's authorisation logic. Blind extraction is slow but entirely practical to automate.

Reproduction steps

1. Authenticate as any user and establish a timing baseline:

```
GET /v2/shipments?sort=created_at HTTP/1.1
Authorization: Bearer <token>
```

Response: 200 OK in ~180ms.

1. Submit a conditional delay payload:

```
GET /v2/shipments?sort=(CASE%20WHEN%20(1=1)%20THEN%20pg_sleep(5)%20ELSE%20created_at%20END)
HTTP/1.1
Authorization: Bearer <token>
```

Response: 200 OK in ~5.2s.

1. Invert the condition to (1=2) and confirm the response returns in ~180ms, demonstrating that the delay is conditional on the injected expression rather than on load.

Remediation

Do not interpolate the sort field. Map the client-supplied value through an allow-list of permitted column names and reject anything not on it:

```
SORTABLE = {"created_at", "updated_at", "status", "eta"}
if sort not in SORTABLE: raise BadRequest
```

Apply the same treatment to the order parameter, which accepts asc / desc and is interpolated in the same statement.

References

CVSS v3.1 8.1 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:H/A:N) · WSTG-INJT-05 · CWE-89

Retest, 1 October 2026: Resolved. The parameter is validated against an allow-list and unknown values return 400 . Timing payloads no longer produce a differential.

3.3.4 PT-4 - Invoice totals recalculated from client input during approval

Severity: HIGH · **State:** Fixed · **Identified:** 13 August 2026 · **Resolved:** 26 September 2026

Affected asset: POST <https://api.northwind.example/v2/invoices/{id}/approve> Requires the Dispatcher role or above.

Description

The approval endpoint accepts the invoice line items in the request body and recomputes the total from them, rather than reading the stored invoice. A dispatcher can therefore approve an invoice whose recorded total differs from the one that was reviewed, and the approval audit entry records only the identifier and the approving user, not the values approved.

This is a business-logic flaw rather than a technical vulnerability: every individual request is well-formed and authorised. It is the kind of issue a scanner cannot find, because nothing about the traffic is malformed.

Business impact

An insider with the Dispatcher role, or an attacker who has obtained such an account, can alter the financial value of an approved invoice without leaving a record of the discrepancy. Because the audit trail does not capture the approved amounts, reconciliation against the stored invoice is the only way to detect it, and that reconciliation is not currently performed.

Reproduction steps

1. As a Dispatcher, create or locate an invoice and note its stored total:

```
GET /v2/invoices/9931 HTTP/1.1
→ {"id":9931,"status":"pending","total":"18400.00","lines":[{"sku":"FRT-STD","qty":8,"unit":"2300.00"}]}
```

1. Approve it, supplying modified line items:

```
POST /v2/invoices/9931/approve HTTP/1.1
Content-Type: application/json

{"lines":[{"sku":"FRT-STD","qty":8,"unit":"115.00"}]}
```

1. Re-read the invoice and observe that the stored total has changed:

```
GET /v2/invoices/9931 HTTP/1.1
→ {"id":9931,"status":"approved","total":"920.00", ...}
```

1. Inspect the audit log entry and confirm it records `invoice 9931 approved by u_8821` with no amounts.

Remediation

The approval endpoint should accept only the invoice identifier and an idempotency key. Read the line items from storage, compute the total server-side, and reject the request if the client supplied any pricing fields at all rather than silently preferring them.

Record the approved total and a hash of the line items in the audit entry, so a later discrepancy is detectable.

References

CVSS v3.1 7.1 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:H/A:N) · WSTG-BUSL-03 · CWE-840

Retest, 1 October 2026: Resolved. The endpoint now accepts only `{idempotency_key}` and returns `422` if pricing fields are present. The audit entry records the computed total and a SHA-256 of the line items.

3.3.5 PT-5 - Session not invalidated on password change

Severity: MEDIUM · **State:** Fixed · **Identified:** 12 August 2026 · **Resolved:** 22 September 2026

Affected asset: POST `https://api.northwind.example/v2/auth/password`

Description

Changing a password issues a new session but does not revoke existing ones. A session captured before the change continues to authenticate successfully until its natural 30-day expiry.

Business impact

A user who changes their password in response to a suspected compromise does not actually evict the attacker. This is the control users most reliably reach for after a phishing incident, and it currently does not do what they believe it does. The 30-day session lifetime means an attacker retains access for up to a month after the user has taken the one action they know to take.

Reproduction steps

1. Authenticate as a Dispatcher in browser A and retain the session cookie.
2. In browser B, authenticate as the same user and change the password via `POST /v2/auth/password`.
3. In browser A, request `GET /v2/shipments`. Observe `200 OK`; the original session remains valid.
4. Repeated at 24 hours with the same result.

Remediation

On a successful password change, revoke all sessions for that user except the one performing the change, and notify the user by email that other sessions were signed out. The same should apply on email change and on MFA enrolment or removal.

References

CVSS v3.1 5.4 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:L/I:L/A:N) · WSTG-SESS-07 · CWE-613

Retest, 1 October 2026: Resolved. Sessions are revoked on password change and a notification is sent. Verified across three concurrent sessions.

3.3.6 PT-6 - Scheduled export delivers full dataset regardless of role scope

Severity: MEDIUM · **State:** Fixed · **Identified:** 13 August 2026 · **Resolved:** 26 September 2026

Affected asset: Nightly scheduled report job, delivered to the address configured at `https://app.northwind.example/settings/reports`

Description

A Viewer may schedule a report. The interactive report view correctly limits rows to the lanes that role is assigned, but the nightly job that generates the scheduled export runs with a service identity and applies no role filter, so the delivered file contains every shipment in the tenant.

Business impact

Role scoping is enforced in the interface but not in the export, so the least-privileged role can obtain the full tenant dataset by scheduling a report and reading the email. Northwind assigns the Viewer role to external parties including customs brokers and some consignees, so this discloses the tenant's complete shipment history to parties deliberately given limited access.

Reproduction steps

1. As a Viewer restricted to the `EU-INBOUND` lane, open the shipments view and note that 212 rows are visible.
2. Navigate to report settings and schedule a daily shipment export to a controlled address.
3. Retrieve the delivered CSV the following morning and count the rows: 4,318, spanning all lanes including `US-DOMESTIC` and `APAC-OUTBOUND`.
4. Confirm the file contains lane values the account cannot view interactively.

Remediation

Generate the export under the scheduling user's identity and apply the same scoping the interactive view uses, rather than running the job as a service principal. Where a service identity is unavoidable, pass the scheduling user's role scope into the job and filter on it explicitly.

References

CVSS v3.1 6.5 (CVSS:3.1/AV:N/AC:L/PR:L/UI:N/S:U/C:H/I:N/A:N) · WSTG-ATHZ-02 · CWE-280

Retest, 1 October 2026: Resolved. The job now runs with the scheduling user's scope. A Viewer restricted to one lane receives only that lane's rows.

3.3.7 PT-7 - Webhook replay accepted within a 24-hour signature window

Severity: LOW · **State:** Risk Accepted · **Identified:** 14 August 2026

Affected asset: POST `https://hooks.northwind.example/v1/partner/status`

Description

Partner status callbacks are authenticated by an HMAC signature over the body. The signature includes a timestamp and the receiver rejects requests older than 24 hours, but it does not record which signatures have been seen. A captured request can therefore be replayed any number of times within that window, producing duplicate status transitions.

Business impact

An attacker positioned to observe one valid callback, or a partner acting in bad faith, can replay a delivery confirmation repeatedly. The practical effect is duplicated status history and potentially premature invoicing, rather than data disclosure. Exploitation requires access to a signed request, which limits the realistic attacker set to the partner themselves or someone on the network path.

Reproduction steps

1. Capture a valid signed callback to `/v1/partner/status`.
2. Replay the identical request, unmodified, four times over ten minutes.
3. Observe `202 Accepted` on each, and four status transitions recorded against the shipment.

Remediation

Record a hash of each accepted signature with its timestamp and reject any repeat within the validity window. A 24-hour replay cache keyed on the signature is sufficient. Narrowing the window to five minutes would further reduce the exposure.

References

CVSS v3.1 3.7 (CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:N/I:L/A:N) · WSTG-BUSL-05 · CWE-294

Retest, 1 October 2026: Not remediated. Northwind has accepted this risk, noting that the partner set is contractually bound and small, and that duplicate transitions are caught in reconciliation. Northwind intends to narrow the validity window in a future release.

3.3.8 PT-8 - Security headers absent on API responses

Severity: LOW · **State:** Risk Accepted · **Identified:** 11 August 2026

Affected asset: All responses from `https://api.northwind.example`

Description

API responses omit `Strict-Transport-Security`, `X-Content-Type-Options` and `Content-Security-Policy`. The portal at `app.northwind.example` sets all three correctly; the API hostname does not.

Business impact

Limited in isolation. The absence of HSTS on the API hostname means a client that reaches it over plaintext on first contact is not protected against downgrade, which matters most for non-browser integrations that do not enforce TLS independently. The missing content-type option permits MIME sniffing of error responses in older browsers. No exploitation path with material impact was demonstrated.

Reproduction steps

```
curl -sI https://api.northwind.example/v2/shipments -H "Authorization: Bearer <token>"
```

Observe that the response contains no `Strict-Transport-Security`, `X-Content-Type-Options` or `Content-Security-Policy` header.

Remediation

Set `Strict-Transport-Security: max-age=31536000; includeSubDomains` and `X-Content-Type-Options: nosniff` at the API gateway. A restrictive `Content-Security-Policy` such as `default-src 'none'` is appropriate for a JSON API.

References

CVSS v3.1 3.1 (CVSS:3.1/AV:N/AC:H/PR:N/UI:R/S:U/C:L/I:N/A:N) · WSTG-CONF-07 · CWE-693

Retest, 1 October 2026: Not remediated. Northwind has accepted this risk pending a planned gateway migration in Q1 2027.

Appendices

A. Scope and methodology

Testing followed the **OWASP Web Security Testing Guide (WSTG)**, an industry-accepted penetration testing methodology, supplemented by PTES for engagement structure. Findings are rated using **CVSS v3.1**, with a business-impact rationale stated alongside each score.

The engagement proceeded in four stages: reconnaissance and attack-surface mapping; authenticated and unauthenticated discovery across all three roles in both tenants; exploitation of candidate issues; and review, in which the lead penetration tester reviewed every finding before it was written up.

Ambit's AI agent carried out the testing inside the authorized scope, under the direction of the lead penetration tester. **Every finding in this report was reviewed by the lead penetration tester before publication, and the report is signed by them.**

Testing modes

	BLACK BOX	GREY BOX	WHITE BOX
Access	No credentials or internal information	Credentials for each role, no source	Full source and infrastructure access
Strength	Most closely models an external attacker	Efficient; reaches authenticated surface that black box cannot	Most complete coverage
Weakness	Time spent on discovery rather than testing; authenticated flaws usually missed	Does not see flaws only visible in source	Requires most client effort; least like a real attack

This engagement was **grey box**, which is the standard choice for a multi-tenant SaaS application because authorisation flaws are only reachable with credentials, and the most serious findings here (PT-1, PT-2, PT-6) are all of that kind.

B. Vulnerability coverage

The following classes were explicitly tested. Coverage is reported so the reader can see what was assessed, not only what was found.

INJECTION AND EXECUTION	AUTHENTICATION AND ACCESS CONTROL
SQL injection (PT-3)	Broken authentication
NoSQL / LDAP injection	Missing authentication on critical functions
Cross-site scripting	IDOR and cross-tenant access (PT-1)
Server-side template injection	Improper access control (PT-2, PT-6)
Server-side request forgery	Cross-site request forgery
Command injection	Credential-stuffing resistance (configuration review only, see §1.5)
Insecure deserialisation	Session management (PT-5)

DATA AND LOGIC	CONFIGURATION, FILES AND CRYPTOGRAPHY
Business logic (PT-4, PT-7)	Unrestricted file upload
Sensitive information exposure	Path and filename control
Information leakage via errors	Open redirect
Directory listing	JWT signature verification
Web cache poisoning	Weak cryptography and TLS configuration
Cookie integrity and scoping	Security headers (PT-8)
Mass assignment (PT-2)	Hard-coded credentials

Not applicable to this engagement: file upload (the application accepts no user-supplied files), and XML processing (no XML endpoints are exposed).

C. Compliance mapping

SOC 2

CRITERION	WHAT IT ASKS	WHERE THIS REPORT EVIDENCES IT
CC4.1	Periodic separate evaluations confirming controls are present and functioning	This assessment in full: §1.4 scope, §1.6 timeline, Appendix A methodology, Appendix B coverage
CC7.1	Identification of vulnerabilities, and the response to them	§3.2 with State per finding, §3.3 detail, and the retest results closing each item

CC7.1 covers *responding* to vulnerabilities, not only finding them. The evidence chain an assessor expects is this report, Northwind's remediation tickets, and the retest confirmation. Two of the three are produced here; the tickets are Northwind's.

ISO/IEC 27001:2022

CONTROL	WHAT IT ASKS	WHERE
A.8.8	Management of technical vulnerabilities	§3 findings, Appendix D severity model
A.8.29	Security testing in development and acceptance	This assessment, Appendix A

Scope boundary

This engagement covered the application layer only. **It does not by itself satisfy PCI DSS Requirement 11.4**, which additionally requires network-layer testing and testing from inside the network. Northwind does not currently process cardholder data; this is noted so the boundary is explicit should that change.

D. Severity ratings

Severity combines exploitability with impact, scored using CVSS v3.1 and stated with a business rationale.

RATING	CVSS	MEANING
Critical	9.0–10.0	Directly exposes data across trust boundaries, or permits full compromise. Remediate immediately.
High	7.0–8.9	Significant confidentiality or integrity impact, reachable with low privilege. Remediate before the next release.
Medium	4.0–6.9	Material impact, or high impact with meaningful preconditions. Remediate on the normal cycle.
Low	0.1–3.9	Limited impact, or impact requiring unlikely preconditions. Remediate opportunistically.
Informational	0.0	No direct security impact. Hardening observation.

Exploitability accounts for the ease of exploitation, whether working exploit code exists, the level of access required, and the plausible attacker's motivation. A finding may be rated above its raw CVSS score where the business context warrants it; where that occurs it is stated in the finding.

E. Glossary

TERM	DEFINITION
Authentication	Verifying the identity of a user, device or system before granting access.
Authorisation	Determining what an authenticated party is permitted to access. Most findings in this report are authorisation failures, not authentication ones.
CVSS	Common Vulnerability Scoring System. A standard method for rating the severity of a vulnerability.
CWE	Common Weakness Enumeration. A catalogue of software weakness types.
HMAC	A keyed hash used to prove a message came from a party holding the key and was not altered.
IDOR	Insecure Direct Object Reference. An application exposes an internal identifier and acts on it without checking the requester is entitled to it.
Mass assignment	Binding client-supplied fields directly onto a server-side object, allowing a client to set fields it should not control.
Tenant	One customer's isolated data within a shared application. Cross-tenant access means one customer reaching another's data.
WSTG	OWASP Web Security Testing Guide. The methodology this engagement followed.