

Dawnbrief Morning brief

NorthWind Mobile + https://api.northwind.io 2026-08-19

Site: Dawnbrief
Brief: Morning brief
Date: 2026-08-19
Target: NorthWind Mobile + https://api.northwind.io
Coverage: 20:00 previous day 04:00 (delivered 07:00 local)
Findings: 5 total 2 critical 1 high 1 medium 1 low 5 new since yesterday

Description	Qty	Unit	Amount
2 Critical block merges on the affected routes	2	\$0.00	\$0.00
1 High schedule in the current sprint	1	\$0.00	\$0.00
1 Medium triage during the next security review	1	\$0.00	\$0.00
1 Low hygiene / rolled into next housekeeping PR	1	\$0.00	\$0.00

=== Top fix today ===F-2026.08.19-01 CVSS 9.0Stale device-bound JWT accepted on /devices/refresh after long absenceWhy now: A revoked device token gets a fresh 90-day JWT on the next call; the auth layer skips the issuer ping entirely. Pull the device->token rotation onto the success of a JWKS validation that fails closed.=== Prioritized fix order ===P1. F-2026.08.19-01 CVSS 9.0 Stale device-bound JWT accepted on /devices/refresh after long absenceP2. F-2026.08.19-02 CVSS 8.5 Race condition in refund endpoint allows double-spendP3. F-2026.08.19-03 CVSS 7.4 Server-Side Template Injection in survey answersP4. F-2026.08.19-04 CVSS 6.1 Open redirect on social-login callback via ?next= parameterP5. F-2026.08.19-05 CVSS 2.7 Response banner leaks internal cluster name on health endpoint=== Findings (grouped by severity) ===== Critical (2) ---Recommended fix: Pin refresh to a JWKS lookup that includes a `revoked\_at` claim per device. Reject refresh when the cached revocation timestamp is newer than the JWT issue time, and log the attempt.F-2026.08.19-01 CVSS 9.0 Critical CWE-287 (see cwe.mitre.org/data/definitions/287.html) Stale device-bound JWT accepted on /devices/refresh after long absenceEndpoint: POST /api/v1/devices/refresh Status: new since yesterday Since: 2026-08-19Summary: Issued JWT carries only a device claim; the auth middleware never cross-checks device revocation before minting a refreshed token.Impact: Stolen phones keep API access even after the user reports them lost. Direct path to order history, saved payment methods, and the push-token rotation endpoint.Reproduction: 1. Sign in on a mobile client and capture the issued JWT. 2. Revoke the device from the admin console; wait 30 days without uninstalling. 3. Call /api/v1/devices/refresh with the revoked token; server mints a fresh JWT instead of returning 401.F-2026.08.19-01 reproducer:curl -sS https://api.northwind.io/api/v1/devices/refresh \-H "Authorization: Bearer \$REVOKED_JWT" \-H "x-device-id: \$DEVICE_UUID" \-i | head -n 12Recommended fix: Wrap the refund transaction in a per-order row lock or a stable hash token returned at refund creation. Add an integration test that replays N concurrent refunds and asserts one succeeding row.F-2026.08.19-02 CVSS 8.5 Critical CWE-362 (see cwe.mitre.org/data/definitions/362.html) Race condition in refund endpoint allows double-spendEndpoint: POST /api/v1/refunds Status: new since yesterday Since: 2026-08-19Summary: Refund handler reads the order total, deducts from the wallet, then writes the ledger but never takes a per-order lock, so concurrent calls observe the same pre-balance.Impact: Trivially scriptable balance drain for any signed-in user with one successful refund to their name.Reproduction: 1. Issue a refund for order ord_2a1b (\$48.20). 2. Replay the same refund request 10x in parallel from one shell loop. 3. All ten replies return 200; the wallet debits 10x the refund amount.F-2026.08.19-02 reproducer:for i in \$(seq 1 10); do curl -sS -X POST https://api.northwind.io/api/v1/refunds \-H "Authorization: Bearer \$JWT" \-H "content-type: application/json" \-d '{"orderId":"ord_2a1b","reason":"duplicate"}' &done; wait--- High (1) ---Recommended fix: Render survey text through a non-executing renderer (markdown with HTML stripped), never the same template engine that renders admin pages. Add a write-time filter that rejects {{...}}

patterns.F-2026.08.19-03 CVSS 7.4 High CWE-94 (see cwe.mitre.org/data/definitions/94.html) Server-Side Template Injection in survey answersEndpoint: POST /api/v1/surveys/:id/answer Status: new since yesterday Since: 2026-08-19Summary: Survey answers are stored verbatim and fed into the admin console template engine without escaping or a sandbox.Impact: Read arbitrary template globals (config, secrets, environment) on the admin console running as the renderer.Reproduction: 1. Submit a survey answer containing `{{7*7}}` as the free-text value. 2. Open the same survey in the internal admin console. 3. The page renders 49 where the curly-brace expression used to be sandboxed execution confirmed.F-2026.08.19-03 reproducer:curl -sS https://api.northwind.io/api/v1/surveys/S-29/answer \ -H "Authorization: Bearer \$JWT" \ -H "content-type: application/json" \ -d '{"text":"Hello {{config.SECRET_KEY}}"}'--- Medium (1) ---Recommended fix: Allowlist ``next`` to a relative path or a host in the same apex. Reject absolute URLs that dont match the configured allowlist before issuing the 302.F-2026.08.19-04 CVSS 6.1 Medium CWE-601 (see cwe.mitre.org/data/definitions/601.html) Open redirect on social-login callback via `?next=` parameterEndpoint: GET /auth/social/callback Status: new since yesterday Since: 2026-08-19Summary: The social-login callback passes through whatever ``next`` value the client posts, including off-host URLs.Impact: Phishing chain after a successful login a real-looking post-login page on the attacker domain keeps the users session cookie alive in their browser context.Reproduction: 1. Click a crafted link with `?next=https://attacker.example`. 2. The OAuth provider returns to /auth/social/callback and the server 302s the browser to the attacker domain. 3. The user has no indication their session cookie was just exchanged for an attacker-controlled landing page.F-2026.08.19-04 reproducer:curl -sI "https://api.northwind.io/auth/social/callback?code=&next=https://attacker.example" \ | grep -i '^location:'--- Low (1) ---Recommended fix: Strip the diagnostic fields from the public response shape. Gate the path at the edge with the same internal-only allowlist block as the rest of the /internal/ tree.F-2026.08.19-05 CVSS 2.7 Low CWE-200 (see cwe.mitre.org/data/definitions/200.html) Response banner leaks internal cluster name on health endpointEndpoint: GET /api/v1/_internal/ping Status: new since yesterday Since: 2026-08-19Summary: A /ping diagnostic leaks the AWS EKS cluster short-name on the wire. The endpoint was meant to be internal-only but the edge router doesnt strip the path.Impact: Helps target reconnaissance against the staging cluster name pattern.Reproduction: 1. Hit the internal ping endpoint signed in as any user. 2. The response body contains the short cluster name (e.g. "nwm-prod-7af3") alongside the OK status. 3. Combine with the framework banner from CWE-general endpoint for fingerprinting.F-2026.08.19-05 reproducer:curl -sS https://api.northwind.io/api/v1/_internal/ping \ -H "Authorization: Bearer \$JWT"