1st Cyber

Penetration test report

Sample report: online shop and API

Prepared for Example Retail (fictional)

0	1	2	2	1
CRITICAL	HIGH	MEDIUM	LOW	FOR INFORMATION

REFERENCE	10004
DATE	15 November 2026
SCOPE	Version 1
TESTING	15 November 2026 to 19 November 2026, 00:00–23:59 London time
CLASSIFICATION	Confidential

Sample report: online shop and API

Example Retail (fictional)

Penetration test report | 15 November 2026

Scope version 1 | Web application, API

Contents

Document control 2
Executive summary 2
Findings summary 3
Approved scope and testing authorisation 4
Methodology 5
Findings 6
Appendix: severity ratings 11

DOCUMENT CONTROL	
Report	Penetration test report, version 1
Date	15 November 2026
Reference	10004
Scope	Version 1, Web application, API
Prepared by	1ST Cyber
Classification	Confidential: for Example Retail (fictional) and 1ST Cyber only

Executive summary

This is a sample report for a fictional company, Example Retail: its people, systems, findings and dates are invented, to show what a 1ST Cyber report contains. We tested the online shop at shop.example.com and its API at api.example.com in Example Retail's staging environment, as a visitor and as two signed-in test customers. We found one high, two medium and two low severity issues, and one for information. The most serious lets any signed-in customer read and change other customers' orders through the API, and should be fixed first. Each finding below gives the affected assets, the business impact, how to reproduce it and how to fix it.

FINDINGS BY SEVERITY

0	1	2	2	1
CRITICAL	HIGH	MEDIUM	LOW	FOR INFORMATION

FINDINGS SUMMARY

Ref	Finding	Severity	CVSS	Status
F-01	Any signed-in customer can read and change other customers' orders	High	7.1	Open
F-02	A script in a product review runs for every shopper who reads it	Medium	5.4	Open
F-03	No limit on failed sign-in attempts	Medium	5.3	Open
F-04	A session still works on the server after sign-out	Low	3.1	Open
F-05	The shop does not tell browsers to use HTTPS alone	Low	3.1	Open
F-06	The web server names its version	For information	–	Open

REMEDIATION PRIORITIES

Severity	Findings	What to do
High	1	Prioritise immediate review and containment.
Medium	2	Schedule remediation according to business risk.
Low	2	Schedule remediation according to business risk.
For information	1	No action needed; listed for the record.

Approved scope and testing authorisation

AUTHORISATION	
Scope agreed by	Lead tester, 1ST Cyber, on 15 November 2026 at 22:02 GMT
Authorised by	Alex Sample (Head of IT) (alex.sample@example.com)
Authorised on	15 November 2026 at 22:02 GMT

SCOPE	
Organisation name	Example Retail (fictional)
Assets in scope	https://shop.example.com | URL https://api.example.com/v1 | URL
Assets excluded	None
Environment	Staging
Testing objectives	Find what an attacker could do to the shop's customers and their orders, signed in and not.
Exclusions	Denial of service, the payment provider's pages, and email to real customers.
Prohibited techniques	Not specified
Rate limits	Not specified
Exploitation limits	Not specified
Authenticated or unauthenticated testing?	Authenticated
User roles to test as	Customer
Do the test accounts need a code or an app to sign in (MFA)?	No
Is it a mostly static site, with little or no interaction?	No
About how many functions does the application have?	12
API style: REST, GraphQL or other	REST
Authentication mechanism	Bearer tokens, issued when a customer signs in

SCOPE (CONTINUED)	
What documentation does the API have?	OpenAPI or Swagger
About how many operations does the API have?	20
How many of those operations take more than 15 inputs?	2
Does a web or mobile application use this API?	One in this request
Should we also test the API on its own?	Yes
Booked testing dates	15 November 2026 to 19 November 2026, 00:00–23:59 London time

CLIENT CONTACTS AT THE TIME OF THIS REPORT	
Client contact	Alex Sample, alex.sample@example.com, +44 20 7946 0122
Emergency contact	Jo Sample, +44 20 7946 0123, 08:00–18:00 weekdays

AUTHORISATIONS UNDER WHICH TESTING TOOK PLACE
Scope version 1, authorised by Alex Sample (Head of IT) on 15 November 2026 at 22:02 GMT. Testing from 15 November 2026 at 22:02 GMT, in progress at the time of this report.

Methodology

Manual testing of the authorised scope, supported by automated tools: how the shop and its API sign customers in and keep their sessions, what each customer may see and change, how they handle what customers send them, and how they protect data in transit. We tested as a visitor and as the two test customers Example Retail created, and used no real customer's data.

Findings

F-01 Any signed-in customer can read and change other customers' orders — High

SEVERITY	CVSS 3.1	STATUS	OBSERVED
High	7.1	Open	15 November 2026 at 22:02 GMT

Affected assets	https://api.example.com/v1/orders
Description	The orders API returns an order to any signed-in customer who asks for its number, whoever placed it, and accepts a change to its delivery address. Order numbers run in sequence, so every order can be found by counting through them.
Impact	A customer could read the names, delivery addresses and order history of every other customer, and send their deliveries elsewhere. The names and addresses are personal data under the UK GDPR.
Reproduction steps	1. Sign in to the shop as test customer A and note the number of one of your orders. 2. Sign in as test customer B and request customer A's order with customer B's access token, as in the request below. 3. The API returns customer A's order, with its delivery address. 4. Send a PUT request for the same order with a new delivery address: the API accepts it.
Request	GET /v1/orders/100482 HTTP/1.1 Host: api.example.com Authorization: Bearer [REDACTED]
Response	HTTP/1.1 200 OK Content-Type: application/json {"id": 100482, "customer": "test-customer-a", "deliveryAddress": "[REDACTED]", "status": "dispatched"}
Evidence	Tested with the two test customer accounts Example Retail created; no real customer's data was read or changed.
Remediation guidance	Check on the server, on every request to /v1/orders, that the order belongs to the signed-in customer, for reading and for changes. Order numbers that cannot be guessed would also make orders harder to find, but only beside the check, never in place of it.
References	CWE-639: Authorization Bypass Through User-Controlled Key OWASP API Security Top 10 2023: API1 Broken Object Level Authorization

F-02 A script in a product review runs for every shopper who reads it

Medium

SEVERITY	CVSS 3.1	STATUS	OBSERVED
Medium	5.4	Open	15 November 2026 at 22:02 GMT

Affected assets	https://shop.example.com/products
Description	Product reviews are shown to every visitor without their HTML being encoded, so a script written into a review runs in the browser of anyone who opens the product's page (stored cross-site scripting).
Impact	A signed-in customer could post a review whose script changes what other shoppers see, or acts in the shop as them while they are signed in. The session cookie is HttpOnly, so a script cannot read it.
Reproduction steps	1. Sign in as test customer A. 2. Post a review of the test product with the text in the request below. 3. Open the product's page as test customer B: an alert box shows the shop's domain.
Request	POST /v1/products/2207/reviews HTTP/1.1 Host: api.example.com Authorization: Bearer [REDACTED] Content-Type: application/json {"rating": 5, "text": "Great value <img src=x onerror=alert(document.domain)>"}
Response	HTTP/1.1 201 Created
Evidence	The review was posted on a test product and deleted after the test.
Remediation guidance	Encode reviews for the page when they are shown, and accept no HTML in them. A Content-Security-Policy that refuses inline scripts would limit the harm of any that is missed.
References	CWE-79: Improper Neutralization of Input During Web Page Generation ('Cross-site Scripting') OWASP Top 10 2021: A03 Injection

F-03 No limit on failed sign-in attempts

Medium

SEVERITY	CVSS 3.1	STATUS	OBSERVED
Medium	5.3	Open	15 November 2026 at 22:02 GMT

Affected assets	https://shop.example.com/account/sign-in
Description	The sign-in page accepted 500 wrong passwords in a row for one test account, with no delay, lock or challenge.
Impact	An attacker could try many passwords against one customer's account, or a few common passwords against many accounts, until one works.
Reproduction steps	1. Send 500 sign-in requests for test customer A, each with a wrong password. 2. Each is answered at once, with the same error. 3. Sign in with the right password: it works at once.
Evidence	500 requests in about two minutes. Example Retail reset the account's password after the test.
Remediation guidance	Slow down and then limit repeated failed sign-ins, for each account and each address, and alert on bursts of them. Offer customers two-step sign-in.
References	CWE-307: Improper Restriction of Excessive Authentication Attempts OWASP Top 10 2021: A07 Identification and Authentication Failures

F-04 A session still works on the server after sign-out

Low

SEVERITY	CVSS 3.1	STATUS	OBSERVED
Low	3.1	Open	15 November 2026 at 22:02 GMT

Affected assets	https://shop.example.com
Description	Signing out removes the session cookie from the browser, but the session itself stays valid on the server: a copy of the cookie still opens the account afterwards.
Impact	Anyone holding a copy of a customer's session cookie, from a shared computer or a log, could keep using the account after the customer has signed out.
Reproduction steps	1. Sign in as test customer A and copy the session cookie. 2. Sign out. 3. Request the account page with the copied cookie: the page is returned.

Request	`GET /account HTTP/1.1`
	`Host: shop.example.com`
	`Cookie: session=[REDACTED]`

Response	`HTTP/1.1 200 OK`
Remediation guidance	End the session on the server when a customer signs out, and end sessions left idle.
References	CWE-613: Insufficient Session Expiration OWASP Top 10 2021: A07 Identification and Authentication Failures

F-05 The shop does not tell browsers to use HTTPS alone

Low

SEVERITY	CVSS 3.1	STATUS	OBSERVED
Low	3.1	Open	15 November 2026 at 22:02 GMT

Affected assets	https://shop.example.com
Description	The shop does not send the Strict-Transport-Security header, so a browser does not know to use HTTPS alone. A visit that starts over HTTP can be intercepted before the shop redirects it.
Impact	On a hostile network, such as public Wi-Fi, an attacker could keep a customer's visit unencrypted and read or change what they send.
Reproduction steps	1. Request https://shop.example.com/. 2. The response has no Strict-Transport-Security header.
Remediation guidance	Send Strict-Transport-Security: max-age=31536000; includeSubDomains on every HTTPS response, once every subdomain serves HTTPS.
References	CWE-319: Cleartext Transmission of Sensitive Information OWASP Top 10 2021: A05 Security Misconfiguration

F-06 The web server names its version

For information

SEVERITY	CVSS	STATUS	OBSERVED
For information	Not scored	Open	15 November 2026 at 22:02 GMT

Affected assets	https://shop.example.com
Description	Every response from the shop, and from its API, names the web server and its version in the Server header.
Impact	It tells an attacker which published vulnerabilities to look for, and saves them a step.
Reproduction steps	1. Request any page or API route. 2. The Server header names the server and its version.
Response	HTTP/1.1 200 OK Server: nginx/1.24.0
Remediation guidance	Remove the version from the Server header.
References	CWE-497: Exposure of Sensitive System Information to an Unauthorized Control Sphere OWASP Top 10 2021: A05 Security Misconfiguration

Appendix: severity ratings

Severity	CVSS score	What it means
Critical	9.0–10.0	Exploitable with severe impact. Prioritise immediate review and containment.
High	7.0–8.9	A serious weakness. Prioritise immediate review and containment.
Medium	4.0–6.9	Exploitable under some conditions. Schedule remediation according to business risk.
Low	0.1–3.9	Limited impact. Schedule remediation according to business risk.
For information	–	No direct risk. Good practice to consider; it never needs a retest.

WHAT HAPPENS NEXT

Fix each finding, then request a retest in the client portal. A finding is resolved only when a tester confirms it is fixed.