PENETRATION TEST / FICTIONAL EXAMPLE

External security assessment

Executive summary — demonstration environment

Company: Example Operations LLC (fictional) Asset: portal.example.com (reserved example domain)
Assessment: illustrative only

This mock engagement demonstrates how a scoped penetration test report could present business risk and practical remediation. The systems, findings and evidence described here do not correspond to a real client or tested network.

At-a-glance findings

Risk tier	Example count	Primary concern
Critical	0	No critical issue represented
High	2	Identity and administrative access
Medium	3	Configuration and operational hygiene
Low	2	Hardening opportunities

Most important action

Review privileged account MFA coverage and tighten access to the administration surface. Treat the other findings as a prioritised backlog with an owner, target date and explicit acceptance of any residual risk.

Scope and limitations

Illustrative in-scope assets: public web portal and an example API endpoint. Excluded: social engineering, denial-of-service, production data extraction, third-party hosted systems without written permission, and infrastructure outside the agreed scope.

Illustrative conclusion: moderate-to-high remediation priority for the example administration surface. No real attack simulation or vulnerability exploitation was performed.

FINDINGS AND EVIDENCE FORMAT

Prioritised technical findings

ID / Severity	Finding	Recommended direction
PT-01 / High	Administrative accounts not consistently covered by MFA	Enforce phishing-resistant MFA and conditional access policies
PT-02 / High	Overly broad administrative role assignments	Reduce privileged membership and review service accounts
PT-03 / Medium	Outdated dependency in the sample portal	Upgrade, regression test and validate affected components
PT-04 / Medium	Security headers incomplete on selected pages	Apply policy headers appropriate to the application
PT-05 / Medium	Verbose error messages disclose internal details	Reduce disclosure and implement error monitoring

Example detailed finding: PT-01

What was observed (hypothetical): A sample admin sign-in flow allows password-only authentication for a privileged test account. In a real assessment, evidence would identify the specific authorised system, user role, timestamps, reproduction conditions and controls tested.

Why it matters: A stolen credential could allow unwanted administrative changes and access to confidential information.

Recommended fix: Enforce phishing-resistant MFA on privileged accounts; apply conditional access; verify logging; repeat a controlled validation after change approval.

Evidence format: Screenshots, affected paths and timestamps would appear in an access-controlled technical annex. This fictional report deliberately contains no real credentials, payloads or sensitive screenshots.

REMEDIATION / FOLLOW-UP

Turn findings into action

Suggested remediation register

Priority	Owner (example)	Next action
High	Identity administrator	MFA policy review and technical validation
High	IT manager	Privileged role clean-up and access approval
Medium	Application owner	Software updates and regression checks
Medium	Application owner	Security header and error-response review

Verification approach

A follow-up test should confirm whether the intended mitigation is effective. When the signed scope includes retesting, the report can record the original finding, customer change evidence, retest observation and current status: resolved, partially resolved, accepted risk or not tested.

How a real engagement differs

The final statement of work defines target assets, test windows, allowed techniques, people authorised to request a stop, severity approach, confidentiality, evidence handling, report delivery and retesting. Prices and service levels are confirmed in that statement of work.

Important disclaimer

This example exists solely to show potential report structure. It is neither a vulnerability disclosure nor evidence of KITAU having completed any named client engagement. Security testing does not guarantee that an environment is free from vulnerabilities.