

ILLUSTRATIVE SAMPLE REPORT

Secure Source Code Review

Security assessment report | fictional scenario

Prepared by Ovid Technologies Private Limited, Visakhapatnam, India

Document	Illustrative example only
Scenario	A fictional Java and Spring Boot service
Status	Fictional findings; no customer systems assessed
Distribution	Public sample - contains no customer information

This document demonstrates a possible report format. Every organization, asset, observation, risk and remediation example is fictional. It is not evidence of an engagement, certification, compliance or a guarantee of security.

1. Executive summary

This sample illustrates how a scoped review could communicate risks and practical next steps to management and technical teams. The fictional scenario is A fictional Java and Spring Boot service. No live testing was performed for this document.

Illustrative risk profile

Priority	Sample count	Interpretation
High	1	Address promptly after validating business context and change dependencies.
Medium	2	Schedule improvements according to exposure and operational priorities.
Low / informational	0	No example findings in this sample.

Key decisions for management

Assign accountable owners, confirm the affected inventory and approve a remediation schedule. Where operational changes are required, agree maintenance windows, backups, testing and rollback before implementation.

Limitations

A real engagement is limited to its authorized scope, access, timing and agreed testing methods. Severity depends on context and should be reassessed by the customer. An assessment cannot establish that a system is completely secure.

2. Scope and assessment approach

Area	Illustrative engagement boundary
System	A fictional Java and Spring Boot service
Access	Customer-approved test accounts, configuration exports or controlled access as appropriate.
Excluded	Unapproved assets, destructive testing, social engineering and physical access.
Authorization	Written authorization and rules of engagement are agreed before testing.
Production	Non-disruptive methods preferred; intrusive testing requires separate approval.

Methodology

1. Confirm business objectives, assets, contacts and written authorization. 2. Establish rules of engagement and test conditions. 3. Review the agreed architecture and configurations. 4. Perform authorized automated and manual checks. 5. Validate observations, assess impact and document evidence. 6. Agree findings, remediation priorities and any retesting.

Risk classification

The sample uses High and Medium as qualitative priorities based on plausible impact and exposure. These are not calculated CVSS scores. A real assessment may use CVSS where appropriate, with assumptions and environmental factors documented.

Handling and confidentiality

A real report and its evidence should be shared only through approved channels. Avoid including real credentials or sensitive records in public enquiries. Confidential customer information is not shared with external AI services without appropriate authorization.

3. Illustrative findings register

ID	Priority	Finding
S-01	High	Tenant identifier omitted from repository query
S-02	Medium	Sensitive token written to application log
S-03	Medium	Dependency version not tracked in remediation workflow

The following findings are invented examples. Evidence descriptions are synthetic and are not observations from a customer environment.

S-01 | High | Tenant identifier omitted from repository query

Illustrative evidence: Illustrative repository code filters by record ID but does not also constrain tenant ID.

Potential business impact: A caller that can influence identifiers may cross tenant boundaries.

Recommended action: Require tenant context in repository queries and add integration tests for denial cases.

Verification: Confirm the approved change is deployed, exercise the original test condition and relevant negative cases, and retain appropriately protected evidence.

S-02 | Medium | Sensitive token written to application log

Illustrative evidence: A sample exception handler logs a complete bearer token with a request failure.

Potential business impact: Log readers or downstream log systems could gain access to reusable credentials.

Recommended action: Remove token values from logs, redact sensitive fields and rotate exposed test credentials.

Verification: Confirm the approved change is deployed, exercise the original test condition and relevant negative cases, and retain appropriately protected evidence.

S-03 | Medium | Dependency version not tracked in remediation workflow

Illustrative evidence: A fictional Maven dependency lacks a documented owner and update decision.

Potential business impact: A vulnerable component could remain in use without an explicit risk decision.

Recommended action: Maintain an inventory, review advisories and track upgrades or approved exceptions.

Verification: Confirm the approved change is deployed, exercise the original test condition and relevant negative cases, and retain appropriately protected evidence.

4. Remediation and verification plan

Stage	Suggested activity	Owner
1. Triage	Confirm affected assets and business impact; assign priority.	Customer owner
2. Plan	Approve change scope, maintenance window, backup and rollback.	Change owner
3. Implement	Apply the agreed engineering or configuration correction.	Delivery team
4. Verify	Test intended behavior and negative cases; check for regressions.	Customer / agreed reviewer
5. Close	Record evidence, residual risk and approval or exception.	Risk owner

Retesting

Retesting can be agreed as a separate engagement component. A real verification report should identify the original finding, changes examined, test date, test result, residual limitations and evidence reference. Remediation is performed only under an agreed scope and change authorization.

Independence

If a contract requires independent assessment, any potential conflict when OVID performs both assessment and remediation must be disclosed and managed under the agreed engagement terms.

5. Technical references

Assessment methods can be informed by OWASP Top 10, OWASP Web Security Testing Guide, OWASP API Security Top 10, OWASP ASVS, CIS Benchmarks, CVSS, CWE, CVE and NIST cybersecurity guidance as applicable. These references do not imply accreditation or endorsement.

Official resources: owasp.org | cisecurity.org | first.org/cvss | cwe.mitre.org | cve.org | nist.gov/cyberframework

Discuss a project: sales@ovid.in | +91 85229 01661 | ovid.in

6. Assessment planning worksheet

The following worksheet illustrates the information a delivery team would agree before assessing A fictional Java and Spring Boot service. Entries are hypothetical and would be replaced with customer-approved details in an actual engagement.

Planning item	Illustrative entry
Business objective	Understand exposure and prioritize practical corrections.
Asset owner	Customer-appointed technical owner and escalation contact.
Environment	A fictional non-production environment with representative configuration.
Test identities	Separate least-privilege test identities for each relevant role.
Data handling	Synthetic records only; approved secure evidence storage.
Change control	No changes without separate authorization and rollback planning.
Stop conditions	Unexpected service degradation, unapproved asset discovery or customer request.
Communications	Agree status updates, escalation channels and incident contacts.

Scope confirmation

Before work starts, the customer and assessment team should confirm in-scope assets, exclusions, test hours, access boundaries, data handling, emergency contacts and the expected deliverables. Availability of a particular test depends on access and the agreed scope.

7. Test scenarios and coverage

These examples show how the fictional findings could be investigated. They are not an exhaustive security checklist or evidence of tests performed.

Scenario	Illustrative validation and expected result
T-01 | Tenant identifier omitted from repository query	Review the relevant configuration or test path. Expected: controls prevent the condition and the result is recorded without accessing real customer data.
T-02 | Sensitive token written to application log	Review the relevant configuration or test path. Expected: controls prevent the condition and the result is recorded without accessing real customer data.
T-03 | Dependency version not tracked in remediation workflow	Review the relevant configuration or test path. Expected: controls prevent the condition and the result is recorded without accessing real customer data.

Coverage and constraints

Document tested assets, versions, roles and time window alongside the results. A control not assessed should be marked not tested rather than passed. Manual validation complements automated checks; the depth of review depends on the agreed access and scope.

Evidence quality

Record the test context, sanitized request or configuration excerpt, expected result, observed result and enough detail to reproduce safely. Protect raw evidence and remove secrets and personal data before including excerpts in a report.

8. Worked finding and evidence record

The example below demonstrates a technical finding record. Values and identifiers are invented for this document.

Field	Illustrative value
Finding ID	S-01
Title	Tenant identifier omitted from repository query
Priority	High - qualitative example; not a CVSS score
Affected asset	sample-system.example.invalid - fictional asset
Preconditions	Authorized test identity and an approved test environment.
Evidence	Illustrative repository code filters by record ID but does not also constrain tenant ID.
Expected result	The relevant security control prevents the condition and records the event where appropriate.
Observed result	The fictional observation in this example differs from the expected result.
Impact	A caller that can influence identifiers may cross tenant boundaries.
Remediation	Require tenant context in repository queries and add integration tests for denial cases.

Evidence handling

Example evidence is descriptive and contains no actual credentials, payloads or customer information. A real report may include sanitized screenshots, configuration excerpts or references to securely stored evidence, as agreed with the customer.

9. Detailed remediation workplan

Suggested sequencing is illustrative. A real plan depends on ownership, system dependencies, business criticality and approved change windows.

Finding	Change preparation	Acceptance criteria
S-01: Tenant identifier omitted from repository query	Identify owner; review dependencies; back up configuration; agree rollback.	Require tenant context in repository queries and add integration tests for denial cases. Verify expected operations and denied cases.
S-02: Sensitive token written to application log	Identify owner; review dependencies; back up configuration; agree rollback.	Remove token values from logs, redact sensitive fields and rotate exposed test credentials. Verify expected operations and denied cases.
S-03: Dependency version not tracked in remediation workflow	Identify owner; review dependencies; back up configuration; agree rollback.	Maintain an inventory, review advisories and track upgrades or approved exceptions. Verify expected operations and denied cases.

Change safety

Before implementation, verify recovery arrangements and obtain explicit approval. Test in staging where available, monitor the change, document the outcome and retain a rollback path. Escalate unexpected impact to the customer contact and stop work when agreed conditions require it.

Residual risk and exceptions

Where an action is deferred, document the reason, owner, compensating controls, review date and approval. Record residual risk as a customer decision rather than representing an open finding as resolved.

10. Verification checklist and report glossary

Use this checklist as an example of the information a customer may request when verification is included in scope.

Check	Evidence to retain
Scope confirmed	Approved asset and finding list; authorized test conditions.
Fix deployed	Change reference, deployment date and configuration or version.
Original condition	Sanitized result showing whether the original condition remains.
Negative tests	Results for unauthorized roles, paths or configurations where applicable.
Regression tests	Evidence that intended application or infrastructure functions still work.
Closure	Reviewer decision, residual limitations, customer owner and date.

Glossary

Finding: a validated observation relevant to security. Evidence: information supporting an observation. Risk: potential impact considered with exposure and context. Remediation: an authorized change to reduce risk. Retest: verification of an agreed correction. Residual risk: risk remaining after treatment.

Sample document notice

All systems, findings and evidence in this document are fictional. The format illustrates possible deliverables; actual scope, methods, findings, severity, evidence and report contents are agreed for each engagement.