Web Application Security Testing Requirements Document

Prepared By:

Information Network Security Administration (INSA)

2025

Submitted by:

[Client's Name / Organization]

Submitted to:

Information Network Security Administration (INSA)

Cyber Security Audit Division Wollo Sefer, Addis Ababa, Ethiopia

Contact Person:

Tilahun Ejigu (Ph.D.)

Cyber Security Audit Division Head

✉: tilahune@insa.gov.et

📱: +251 937 456 374

Submission Date: [Insert Date]

Due Date for Response: Within Five (5) Working Days from the Date of Receipt

Submit the requirement document through INSA Cyber Audit Request Portal: apply by this link: ***https://cyberaudit.insa.gov.et/sign-up*** ***or*** ✉: tilahune@insa.gov.et

1. Background of INSA

The Information Network Security Administration (INSA) is Ethiopia's leading government body entrusted with safeguarding the country's digital infrastructure and cyber ecosystem. Established with a vision to ensure national cyber sovereignty and resilience, INSA performs critical functions such as conducting cyber security audits, providing policy direction, issuing compliance guidelines, and offering technical support across various governmental and private institutions.

INSA's Cyber Security Audit Division assesses the security posture of digital platforms including web applications, APIs, mobile applications, internal portals, and IT infrastructures. Our assessments identify vulnerabilities and recommend mitigation strategies aligned with global best practices.

2. Introduction

This document outlines the key requirements and submission expectations for organizations undergoing web application security testing with INSA. The aim is to ensure robustness, confidentiality, and integrity of digital assets by identifying and addressing vulnerabilities. INSA applies leading methodologies and standards including OWASP Top 10, NIST guidelines, and ISO/IEC 27001.

3. Objectives of the Audit Request

These assessments are vital for:

- Mitigating cyber risks and vulnerabilities.
- Ensuring compliance with Ethiopian and international cybersecurity standards.
- Strengthening the trust of stakeholders and customers.
- Safeguarding sensitive data and operations from cyber threats.

4. Required Submissions from the Client

You must submit the following via INSA's Audit Request Portal. If submissions include sensitive files (e.g., source code or APK files), deliver those specific files on CD/DVD to INSA's office or mail them to the address provided.

4.1 Legal and Administrative Documents (Mandatory)

1. Updated Trade License
2. TIN Number or National ID
3. System/Product Patent Certificate (if available)

4.2 Technical Documentation for Web Application Security Testing (Mandatory)

To initiate the application audit process, customers must submit all required documentation and information as outlined below. Compliance with these mandatory requirements is a prerequisite for the audit to proceed.

4.2.1 Business Architecture and Design

a) Data Flow Diagram (DFD)

Provide a visual representation of data flows within the application, showing external entities, data stores, and internal processes.

Purpose:

- Identify how data is input, processed, stored, and output.
- Highlight entry points, flows, and system boundaries.
- Support identification of risks such as data leakage, insecure data handling, and unauthorized access.

Required Submission:

- Context-Level DFD (Level 0) showing the system and external actors.
- Detailed DFDs (Level 1/2) showing internal processes and storage interactions.

b) System Architecture Diagram

Provide a high-level blueprint of the ecosystem (components, servers, databases, APIs, and third-party services).

Required Submission:

- **Deployment Architecture**: on-premise, cloud, or hybrid layout of web/app servers and databases.
- **Component Architecture:** modules, service-to-service communication, middleware and integration points.
- **Security Layers:** DMZ/VPN, SSL/TLS, WAF, IDS/IPS, and related controls.

c) Entity Relationship Diagram (ERD)

Illustrate data entities and relationships, including sensitive fields and keys.

Required Submission:

- Tables/entities (e.g., Users, Roles, Permissions, Transactions).
- Primary/foreign keys and relationships.
- Mark sensitive fields requiring encryption and access control.

4.2.2 Features of the Web Application

- Development frameworks (e.g., Django, Laravel, ASP.NET).
- Libraries or plugins integrated.
- Custom-developed modules or APIs.
- Third-party service integrations.
- Actor/user types (Admin, Guest, Super Admin, etc.).
- System dependencies or minimum hosting requirements.
- Implemented security standards or policies.
- Existing security infrastructure (e.g., WAF, Load Balancer, IDS/IPS).

4.2.3 Define Specific Testing Scope (Mandatory)

Provide a structured scope including assets, endpoints, and test credentials.

Asset Name	URL/IP Address	Test Account Credentials
Public Web Portal	www.yourdomain.com	testuser / password123
Internal Web Portal		admin / admin@123
Local Applications		tester1 / test@secure
Public Applications		test@user.com / securePass1

4.2.4 Security Functionality Document

- User roles and access control (RBAC, ABAC).
- Input validation and sanitization approaches.
- Session management (cookie flags, timeouts).
- Error handling and logging.
- Secure communications (TLS/SSL).
- Technical description of each function and control.

4.2.5 Secure Coding Standard Documentation (if available)

- Secure coding guidelines (e.g., OWASP Secure Coding Practices).
- Internal rules/checklists used during development.
- Practices preventing SQLi, XSS, CSRF.
- Secure input handling; file upload/download controls.
- Authentication & session management.
- Regular patching and library validation.

4.2.3 Functional Requirements

The application must provide documented functional behavior for audit purposes. Customers should provide:

- Core application workflows (user registration, login, transaction processing, etc.)
- Input/output validation rules
- API endpoints with request/response structures
- Role-based access control definitions
- Logging and auditing functionalities
- Error handling and exception management

4.2.4 Non-Functional Requirements

The application must document non-functional behaviors affecting security and performance:

- **Performance:** Expected response times, concurrent users, throughput
- **Availability:** Uptime requirements, failover mechanisms
- **Scalability:** Horizontal/vertical scaling plans
- **Reliability:** Backup and disaster recovery processes
- **Maintainability:** Code modularity, documentation, update procedures
- **Security:** Encryption, authentication, authorization, session management, and audit logging.

4.2.5 Threat Modeling

Customers must provide threat modeling documentation to support security analysis:

- Identified threats (SQL injection, XSS, CSRF, privilege escalation, etc.)
- Risk assessment for each threat (impact, likelihood, mitigation priority)
- Mitigation strategies implemented or planned
- Attack surface analysis (entry points, exposed APIs, third-party integrations)
- Security assumptions and dependencies

Required Submission:

- Threat model diagrams (STRIDE, DREAD, or equivalent)
- Risk assessment matrix or table
- Security control mapping to identified threats

4.2.6 Previous Security Testing Reports (if available)

- First phase security audit reports.
- Re-audit/remediation testing documents.
- Evidence of fixes implemented based on earlier findings.

4.2.7 Secure Coding Standard Documentation (if available)

- Secure coding guidelines (e.g., OWASP Secure Coding Practices).
- Internal rules/checklists used during development.
- Practices preventing SQLi, XSS, CSRF.
- Secure input handling; file upload/download controls.
- Authentication & session management.
- Regular patching and library validation

5. API Security Audit Requirements

5.1 Purpose

Define the minimum requirements for initiating an API security audit, ensure sufficient information is available for assessment, and align API operations with security best practices and standards.

5.2 Scope

Applies to all organizations that integrate with or expose APIs subject to the audit, including internal, partner, and public-facing APIs.

5.3 Responsibilities

- Requesting Organization (Customer) must provide all required documents, test accounts, and technical details.
- Audit Team must assess materials against best practices, compliance standards, and organizational policies.
- Third-Party Service Providers must disclose integrations that impact API security.

5.4 Detailed Requirements (Mandatory)

5.4.1 Request/Response File

- Provide sample request and response files for critical API operations.
- Include examples of successful and failed transactions.

5.4.2 Updated API Documentation

- Provide complete, updated API docs including parameters, headers, authentication flows, response codes, error handling.
- Include change logs and version history.

5.4.3 API Types

- Specify API type (REST, SOAP).
- Describe supported data formats (JSON, XML, etc.).

5.4.4 API Endpoints and Functionality

- Submit a full list of endpoints with purpose and supported HTTP methods.
- Categorize endpoints as public, private, or internal.

5.4.5 Authentication Mechanism

- Disclose authentication mechanisms (OAuth 2.0, JWT, API keys, Mutual TLS, etc.).
- Document token lifecycle, renewal, and revocation processes.

5.4.6 Third-Party Integrations

- Declare all third-party APIs and services; document data exchange mechanisms.
- Assess security risks from dependencies.

5.4.7 Compliance and Regulatory Requirements

- Specify applicable requirements (e.g., PCI DSS, GDPR).
- Include jurisdictional and industry-specific obligations.

5.4.8 Authorization and Access Control

- Define user roles and access rights for each API.
- Implement least privilege via RBAC/ABAC.

5.4.9 Test Account

- Provide a dedicated test account simulating multiple privilege levels (admin, user, guest).

6. Contact Information and Communication

Name	Role	Email Address	Phone Number

John Doe	IT Security Officer	john@yourorg.com	+2519XXXXXXXX
Jane Smith	Developer Lead	jane@yourorg.com	+2519XXXXXXXX

7. Submission Instructions

Submit all materials via INSA's Audit Request Portal. For sensitive files (e.g., source code or APKs), deliver those on CD/DVD to INSA's office or mail them to:

Information Network Security Administration (INSA)
Cyber Security Audit Division Wollo Sefer, Addis Ababa, Ethiopia
Contact: Tilahun Ejigu (Ph.D.), Division Head
Email: tilahune@insa.gov.et
Mobile: +251 937 456 374

8. Conclusion

INSA is committed to working collaboratively to improve your organization's digital security posture. By submitting the required documentation within the specified timeline, you enable a comprehensive and accurate audit that enhances compliance, strengthens service delivery, and reduces exposure to cyber threats.