CONFIDENTIAL TECHNICAL DOCUMENT

AI Pentest Platform Technical Architecture & Security Readiness

Professional technical document prepared for project positioning, operational readiness, and leadership-level security assurance.

Prepared for: **Senior Management & CISO** | Project: **DI - ASH AI Pentest Platform** | Date: **February 11, 2026**

Deployment Model
Single-host, on-prem style platform

Primary Use Case
Automated web pentesting + AI-assisted reporting

Audience
Security leadership, engineering, operations

Report Scope
Current implemented architecture and controls

1) Executive Technical Snapshot

The platform is a functional AI-assisted penetration testing system combining automated scanners (Nuclei, Nikto, SQLMap, Katana), a FastAPI backend, local LLM analysis via **llama.cpp**, and professionally formatted multi-view reports (combined, executive, technical). It is suitable for internal security operations, scoped assessments, and leadership reporting.

BACKEND FRAMEWORK	SCANNER ENGINES
FastAPI	**4**
LLM RUNTIME	**REPORT MODES**
Local	**4**

Current status: technically strong for controlled internal usage; requires final hardening controls before external-facing production pitch.

2) Current System Architecture

[User Browser Dashboard] | v [Frontend: HTML/CSS/JS] -> [Search / Scan Initiation / Scan History / AI Chat] | v [FastAPI Backend (main.py)] - Authentication (HTTP Basic) - Scan orchestration and process lifecycle - Findings normalization and CVSS v4.0 mapping - Report generation (combined / executive / technical / compliance) - Posture summary and framework mapping (target-level) - AI chat and explanation endpoints | +--> [Scanner binaries: nuclei, nikto, sqlmap, katana] +--> [SQLite DB: data/pentest.db] +--> [Report artifacts: /reports/*.html + markdown] +--> [LLM: llama-server/llama-cli + local GGUF model]

Key Backend Paths

- Core app/API orchestration: `backend/main.py`
- Frontend dashboard: `frontend/index.html`, `frontend/dashboard.js`, `frontend/styles.css`
- Tool binaries and templates: `tools/` and `tools/nuclei-templates/`
- Data and logs: `data/`, `logs/`, `backend/server.log`

3) Technology Stack (Implemented)

Layer	Technology	Version / Implementation	Purpose
Frontend	HTML5, CSS3, Vanilla JavaScript	Custom dashboard UI	Scan control, live status, chat operations, reporting access
Backend API	FastAPI + Uvicorn	FastAPI 0.128, Uvicorn 0.40	API routing, auth, orchestration, report services
Database	SQLite	`data/pentest.db`	Users, scan metadata, scan result references
Pentest Engines	Nuclei, Nikto, SQLMap, Katana	Local binaries under `tools/`	Vulnerability detection, web misconfig checks, SQLi testing, endpoint discovery
AI Analysis	llama.cpp server/CLI + GGUF model	Qwen 2.5 3B (preferred), fallback models	Natural-language explanations, risk summaries, report narrative support
Reporting	HTML template pipeline	Combined + Executive + Technical outputs	Stakeholder-friendly outputs with findings and remediation context

4) Scan-to-Report Data Flow

Operational Flow

- User initiates scan from New Scan module.
- Backend validates tool and creates scan entry (`pending`).
- Background task executes scanner with timeout controls.
- Outputs are normalized into finding objects (severity/title/evidence).
- CVSS v4.0 style score strings are generated and assigned.
- Reports are rendered and stored in `/reports`.

Chat + Explanation Flow

- Chat command intent parser detects scan/report/explain requests.
- For scan-specific explanations, chat summarizes findings using scan evidence and risk language.
- For topic questions (for example MIME/header issues), chat returns issue description, impact, pentest validation logic, remediation, and references without requiring a scan ID.
- If LLM is unavailable, deterministic fallback guidance is provided.
- Scan start actions auto-refresh scan list and route user to Scan History.
- News ticker and AI status are fetched via authenticated API endpoints.

```
Core APIs: /api/scan, /api/scans, /api/scan/{id}, /api/scan/{id}/report,
/api/report/{id}/html, /api/report/target/{target}/compliance_html,
/api/posture/summary, /api/chat, /api/ai/status, /api/admin/audit/report/html
```

5) Security Controls Currently Implemented

Control Area	Implementation in Current Build	Status
Authentication	Role-aware authentication, first-login password rotation, and step-up action password on sensitive scan/report operations.	**Strong**
Password Storage	Passwords hashed with bcrypt.	**Strong**
Response Hardening	Security headers set (CSP, X-Frame-Options, X-Content-Type-Options, Referrer-Policy, Permissions-Policy).	**Strong**
Scan Isolation	Tool runs as subprocess with per-tool timeouts and stop control.	**Strong**
Auditability	Structured audit events for login, scan/report actions, and admin operations with downloadable audit report.	**Strong**
AI Resilience	LLM server fallback and deterministic fallback messaging are present.	**Strong**

6) Readiness Gaps to Close Before Final Pitch

Priority	Gap	Observed Behavior	Recommended Action
P1	Default admin bootstrap password	Admin default is seeded in startup logic for first run.	Force password change at first login and remove hardcoded bootstrap secret.
P1	CORS wildcard	`allow_origins=["*"]` currently broad.	Restrict to trusted dashboard origin(s) only.
P1	Development reload in runtime scripts	Uvicorn reload enabled in launch paths.	Use production profile (no reload), supervised process manager.
P2	Access control model	Single-admin style operation.	Add RBAC roles (analyst, lead, reviewer, admin) + action audit trail.
P2	Auth defense depth	No explicit login rate limiting or lockout policy.	Implement rate limiting, lockouts, and session hardening.

7) Compliance & Standards Mapping (Ali & Sons Context)

Applicable Security Standards

- **ISO/IEC 27001:** Risk treatment, asset protection, security operations governance.
- **SOC 2:** Security and monitoring evidence mapping for assurance workflows.
- **NIST:** Testing methodology and control-centric remediation tracking.
- **OWASP:** Web vulnerability categories and secure development alignment.
- **CIS Controls + UAE IAS:** Baseline hardening and regional governance expectations.

How This Platform Supports Compliance

- Retains tool evidence and output metadata for audit traceability.
- Generates executive, technical, combined, and separate compliance summary reports.
- Provides posture heat map, framework cards, and target risk cards from the same normalized findings set.
- Supports repeatable retest flow by scan ID and target-based consolidated reporting.

8) Recent Enhancements Delivered (CISO Brief)

Module	Implemented Improvement	Operational Benefit
SQLMap Reliability	Added SQLMap preflight plus multi-variant endpoint discovery (http/https and apex/www), parameter harvesting from Katana/Nikto, and fallback seed crawling.	Lower scan failure rate and broader real parameter coverage for SQLi validation.
False Positive Suppression	System-wide filtering removes SQLMap heuristic/unexploitable indicators from actionable findings and suppresses noisy Nikto remote-read-limit warnings from findings tables.	Cleaner, decision-ready report outputs with reduced analyst confusion.
Report Consistency	Executive "High-Level Findings Overview" now follows total finding counts and no longer truncates small result sets unexpectedly.	Consistent numbers across executive, technical, combined, posture, and compliance views.
Posture + Compliance	Added dashboard posture heat map, framework alignment cards, target risk cards, and target-level compliance report mapped to ISO 27001, SOC 2, NIST, OWASP, CIS, and UAE IAS.	Professional governance reporting directly from operational data.
Light-Mode UX	Refined light-mode login/loading backgrounds, improved dragon/logo contrast, and professionalized report dropdown placement and readability in dark/light themes.	Better usability and presentation quality for leadership demos.
Data Hygiene	Legacy test targets are excluded from scans/reports/posture/compliance summaries to avoid contaminating production metrics.	Accurate and trustworthy posture statistics for business reporting.
GitHub Backup	Git backup automation scripts and service workflow retained for full-project continuity snapshots.	Improved business continuity and change traceability.

9) Operational Runbook (Recommended)

Daily

- Verify scanner binary availability and AI engine health.
- Review failed/time-out scans and rerun targeted assessments.
- Validate report generation for latest scan IDs.

Weekly

- Patch tool binaries and nuclei templates in controlled change window.
- Review critical/high findings trend by business system.
- Run backup and restore validation for `data/`, `reports/`, and `logs/`.

Monthly

- Review risk register alignment with remediation closure SLAs.
- Validate management dashboard metrics and quality of AI explanations.
- Conduct internal control test against ISO 27001/NIST evidence expectations.

10) Senior Management Pitch Narrative (Ready-to-Use)

What it is: A centralized AI-assisted pentest platform that automates detection, contextualizes risk, and outputs leadership-ready reports.

Why it matters: Faster discovery-to-remediation cycles, consistent reporting quality, and improved evidence posture for governance and audits.

What makes it strategic: Local AI analysis engine for data sovereignty, integrated multi-tool validation, and business-readable findings translation.

What is needed next: Final hardening controls (auth, CORS, production runtime profile, RBAC) and formal operations playbook adoption.

11) 90-Day Roadmap (Production Readiness)

Window	Focus	Deliverables
0-30 days	Security hardening baseline	Default credential removal, CORS allowlist, TLS reverse proxy, production server profile.
31-60 days	Governance and control maturity	RBAC, audit log model, retention policy, remediation SLA dashboard.
61-90 days	Scale and assurance	Regression test suite, scheduled scans by asset group, management KPI pack, DR drill.

This document is generated from the current project implementation and aligned for senior management and CISO presentation use. Treat as confidential internal security architecture material.