Guardian One Repository Architecture and Governance

Guardian One documentation | Version 1.1 | Reviewed 30 July 2026

Purpose and Audience

Current version	1.1
Last reviewed	30 July 2026
Document owner	Product owner: Adam Timothy
Change summary	Manager-readability and controlled-publication review

Version history

Version	Date	Change	Status
1.0	Initial issue	Initial controlled document	Superseded
1.1	30 July 2026	Manager-readability and publication review	Current

This briefing explains how Guardian One collects, stores, assesses, and presents SQL Server estate evidence, together with the governance decisions required before deployment. It is intended for IT directors, technical owners, security and governance stakeholders, and architects evaluating operational fit and readiness.

Executive View

Guardian One provides scheduled, evidence-led assessment of SQL Server estates through four service levels: Foundation, Advisor, Compliance, and Quantum.

Business Outcomes

- Repeatable assessment evidence for inventory, configuration, backups, security, and performance.
- Prioritised technical risks and remediation recommendations.
- Retained assessment evidence for governance and audit use.
- A future controlled AI layer for explanation and guided remediation.

Governance Decisions

Before deployment, agree data ownership, instance scope, service accounts, network access, assessment frequency, report recipients, retention, support responsibilities, and change authority. AI use requires separate approval covering data handling, model provider, human review, and prohibited automation.

Current Readiness

The application uses Windows authentication, writes evidence to a SQL Server repository, applies numbered migrations, and centrally verifies signed entitlements. Automated unit tests cover core discovery, persistence, licensing, failure handling, and reporting. The database-backed estate portal reads current repository evidence per request; report artifacts are catalogued, digest-verified and can be recoverably archived through a plan/apply retention command. CI, packaged secret delivery, production signing-key custody, an approved customer retention policy and schedule, production portal authentication, and complete run observability remain release gaps.

Runtime Flow

Guardian One follows five clear stages for each scheduled assessment:

- Verify the signed licence and the purchased service levels.
- Connect to the target SQL Server and automatically attempt permitted Windows evidence locally or through WinRM; classify and log unavailable access without stopping SQL assessment.
- Store the collected evidence in the central Guardian One repository.
- Apply assessment and compliance rules, calculate health, and record findings, recommendations, confidence, and collection limitations.
- Present the stored results through the individual report or the enterprise portal.

The config/.env file identifies the SQL Server endpoint and repository database. SQL collection uses Windows authentication through pyodbc. Guardian One automatically attempts bounded agentless Windows collection locally or through WinRM under the Guardian One service identity; blocked permissions or firewall access are logged as unavailable evidence and do not stop SQL assessment or cause missing metrics to be treated as a pass.

Assessment data is stored in GuardianOne_Toolkit by default. The enterprise portal builds its pages from that database when requested; HTML is not stored in the database and the portal does not depend on a report folder. Customers may optionally create standalone HTML exports for controlled sharing. Guardian One records the export against its assessment run and stores an integrity digest so the file can be checked later.

Portable Repository Boundary

Portable Assessment Mode changes persistence, not collection. Target collectors still connect to the configured real SQL Server and use the same entitlement boundary, rules and report generator. Repository calls are routed to a local SQLite database, so runtime configuration must not attempt a connection to GuardianOne_Toolkit. The complete assessment result is retained as JSON with a SHA-256 digest; HTML is generated separately.

PortableInstanceSlot is an append-only product registry with five numbered slots. The canonical identity returned by SQL Server through SERVERPROPERTY('ServerName') is compared case-insensitively. Reassessing an existing identity updates its last-assessment time without consuming another slot. Application code exposes no deletion, replacement or import operation and never derives available capacity from deletable assessment history.

SQLite does not provide transparent encryption in this implementation. Deployment must protect the database and backups using Windows ACLs, customer-approved disk encryption and controlled service identity access. The product’s lack of delete/import controls is a supported-application boundary, not a claim that a customer with filesystem-administrator access cannot tamper with a local file.

Database-level Portable encryption is a tracked security design item, not an installer switch that can be enabled safely without migration and key recovery. The approved direction is to evaluate a SQLCipher-compatible provider with a random database key protected by Windows DPAPI/CNG, including atomic conversion of existing repositories, backup recovery, key rotation, service-identity changes and failure-safe rollback. Until that work is implemented and release-tested, use customer-approved whole-volume encryption so the database, journal/WAL files, temporary files and backups are covered together.

Customer machine-build policy is repository-backed. Environment templates contain expected drives and mount points, while ordered name rules classify machines from names such as UKSQLDEV01. Each assessment snapshots the resulting classification and comparison into its persisted evidence so the report explains matched, missing, unexpected, or unavailable evidence. Report-presentation settings are loaded at assessment time and stored with the run; disabling a chart affects presentation only and does not suppress collection or delete evidence.

Assurance Expectations

- Principle of least privilege and read-only access to assessed SQL Servers.
- Backend enforcement of purchased service levels.
- Traceable rules, findings, evidence, and entitlement decisions.
- Tested rollback, backup, retention, and incident procedures.
- Product-owner approval before production rollout.

Design Principles

Collect shared evidence once and reuse it consistently. Apply assessment rules, compliance policies, recommendations, reports, and future AI interpretation to stable stored evidence. Verify purchased capabilities before those operations start; hiding controls on a page is not sufficient enforcement.

Legacy Compatibility

Some older internal components are temporarily retained to protect compatibility. New development uses the main src/core architecture. Removal requires regression testing and a documented migration path so existing command-line operation, stored data, database interfaces, and report content are not broken unexpectedly.