MICKAI EBOOK SERIES · PLAYBOOK No. 2

The Audit Substrate Under Every AI Agent.

The Open Audit Record (OAR) primitive, the FIPS 204 ML-DSA-65 signing pipeline, the SHA-3-512 hash-linking, the browser-resident verifier, and the cryptographic position every AI operator should hold.

AUTHOR

Micky Irons

Founder and named inventor, Mickai LTD.
Crunchbase · LinkedIn · GitHub · mickai.co.uk

DATE · 13 May 2026 · v1

EBOOK · No. 2 IN A SERIES OF FOURTEEN

Mickai LTD · Companies House 17166618 · press@mickai.co.uk · mickai.co.uk
UK IPO patent family GB2607309.8 to GB2610422.4 · Trade mark UK00004373277

TABLE OF CONTENTS

Contents

FOREWORD

A note from the author

Every AI agent in 2026 is producing decisions that affect humans, regulators, and balance sheets. The audit trail of those decisions is, today, held under the AI vendor's key in the AI vendor's format. The substrate question is upstream of the policy question. This ebook is the engineering walk-through of the substrate that closes the gap.

The Open Audit Record (OAR) is the cryptographic primitive underneath every AI agent decision in the Mickai Sovereign Intelligence Operating System. The schema is open. The conformance vectors are open. The verifier is open. The operator holds the signing key. There is no vendor in the trust path.

The substrate primitives in this ebook are filed at the UK Intellectual Property Office across the GB2607309.8 to GB2610422.4 patent family. The trade mark Mickai is registered at UK00004373277. The schema, the conformance vectors, and the reference verifier are scheduled for joint open-source release upon UK IPO acknowledgement of the OAR family.

Micky Irons

Founder and named inventor, Mickai LTD · 13 May 2026

PART I · THE AUDIT PROBLEM

Why the AI audit chain is, today, the AI vendor's audit chain

1. Why every AI vendor's audit log is vendor-shaped

An AI vendor in 2026 (a foundation model provider, an AI security platform, an agent framework, a managed orchestration service) ships an audit log. The log is held in the vendor's database, written in the vendor's schema, signed (when it is signed at all) under the vendor's key. The operator has read access through the vendor's portal. The operator does not have custody. The audit is the vendor's artefact; the operator is its consumer.

This is structurally acceptable for the use cases the vendor was designed to serve. It is structurally unacceptable for the regulated buyer. A UK PRA-regulated bank under SS1/23 model risk management, a nuclear licensee under ONR design-data lineage expectation, a pharma operator under MHRA chain-of-custody, an ICO data controller under UK GDPR Article 22, each has to be able to walk the chain of any AI-assisted decision affecting a data subject or a regulated artefact without recourse to the AI vendor.

> **Vendor-key, vendor-format, vendor-cloud audit cannot, by construction, produce the artefact the regulator actually requires.**

2. What a regulator actually walks at hour zero

When a regulator opens an incident review (an FCA s.166, an ICO Article 60 cooperation, an ONR safety case escalation, an MHRA pharmacovigilance deep-dive, a PRA SS1/23 model risk follow-up), the regulator wants to reconstruct what the AI did, when it did it, who it acted on behalf of, what evidence it consumed, what reasoning steps it took, what output it produced, and under whose authority each step happened. The reconstruction has to be deterministic; two parties walking the same chain must produce the same answer.

A vendor JSON log, however detailed, fails this test in two ways. First, the operator cannot verify the log without the vendor's continued cooperation; the vendor could have rewritten the log, lost the log, or ceased operation. Second, the log is rarely cryptographically chained; tampering with a single record is not structurally detectable. The regulator's chain-of-custody question reduces, in practice, to whether the vendor is currently cooperating.

3. The cryptographic position the operator should hold

The position the operator should hold is this: every AI-assisted decision produces a record that is serialised in a canonical format the operator controls, hash-linked under a chain rooted at the operator's first deployment, signed under a key the operator's hardware emits and rotates, and replayable by any party with a public key, a chain file, and a browser. The vendor's continued cooperation is not part of the trust model. The operator can change AI vendors, lose contact with an AI vendor, sue an AI vendor, or watch an AI vendor cease operation; the audit chain continues to verify under the operator's key.

That position is what the Mickai Open Audit Record (OAR) primitive delivers, and what the rest of this ebook walks through in engineering detail.

PART II · THE OAR PRIMITIVE

The Open Audit Record, in engineering detail

4. The chain, the schema, the conformance vectors

The OAR chain is an append-only ledger. Each record contains a header (version, deployment identifier, sequence number, timestamp, parent hash), a body (the AI decision, the prompt, the evidence references, the reasoning trace, the output, the authority context), and a footer (the FIPS 204 ML-DSA-65 signature over the canonical serialisation of the record). The chain identifier is the SHA-3-512 hash of the deployment root record; every subsequent record's parent hash points to the predecessor.

The schema is an open document under a stable URI. Conformance vectors (input records, expected signatures, expected verifier verdicts) ship with the schema. Any party can independently implement the OAR primitive from the schema and the vectors; the substrate is, by intent, a portable standard rather than a vendor implementation.

Record header fields

· version · OAR schema version, integer. Bumps on breaking changes. v1 ships first.
· deployment · operator-controlled deployment identifier, UUID. Stable for the lifetime of the deployment.
· sequence · monotonic integer, starting at 1 for the deployment root.
· ts · RFC 3339 timestamp in UTC. Sub-second precision.
· parent · SHA-3-512 hash of the predecessor record's canonical serialisation, hex-encoded. Zero hash for the root.

Record body fields

· actor · the AI brain or vendor model identifier that produced the decision.
· principal · the human or service identity under whose authority the decision happened.
· action · the action taxonomy term (decide, classify, draft, search, invoke, recommend, etc.).
· prompt · the prompt or input context, or a hash of it when confidentiality requires.
· evidence · array of evidence references, each itself a hash plus a citation.
· output · the AI's output, or a hash of it.
· policy · policy version under which the action was authorised.
· verdict · permit, deny, escalate, defer.

5. FIPS 204 ML-DSA-65 in practice

FIPS 204 ML-DSA-65 is the NIST-standardised post-quantum digital signature scheme, published in August 2024. The scheme is based on Module-Lattice Digital Signature Algorithm, parameter set 65, providing approximately 192 bits of classical security and 128 bits of post-quantum security. Signature

size is approximately 3.3 kB; public key is approximately 2 kB; signing and verification are both fast (sub-millisecond on commodity hardware).

The Mickai signing pipeline holds the ML-DSA-65 private key inside the operator's hardware security module (TPM 2.0 binding the key to the workstation's platform configuration registers). The key never leaves the TPM; signing is delegated to the TPM, and the TPM emits the signature back to the SIOS for inclusion in the OAR record. Key rotation is policy-driven; the operator schedules rotation (typically annually) and the SIOS issues a chain anchoring record that binds the new key to the chain identifier.

> **The signing primitive is post-quantum from inception. An audit chain signed in 2026 verifies against a 2035 cryptographically-relevant quantum attacker without re-signing.**

6. SHA-3-512 hash-linking and replay

SHA-3-512 (NIST FIPS 202, Keccak family) is the hash function that links each OAR record to its predecessor. The chain is constructed so that tampering with any record at position N invalidates every subsequent record at position N+1, N+2, and onwards. The chain is therefore append-only by construction; the operator does not have to enforce append-only at the database or filesystem layer.

Replay is deterministic. Given the deployment public key, the chain file, and a parameter set, any party walks the chain record-by-record, verifies each parent hash matches the predecessor's serialisation hash, verifies each signature against the deployment public key, and emits a verdict per record. The walk runs offline in any browser; no network access is required after the chain and the public key are loaded.

7. CBOR as the canonical serialisation

CBOR (Concise Binary Object Representation, RFC 8949) is the canonical serialisation. The same logical record always serialises to the same byte string; the property the signing operation requires. The Mickai SIOS applies the CBOR deterministic encoding profile (RFC 8949 section 4.2) with map keys in lexicographic order, integers in shortest form, no indefinite-length items, and floats in canonical form.

JSON, by contrast, is not canonical. Whitespace, key order, number representation, and Unicode normalisation all vary across implementations. A signed JSON audit log is therefore not portable across implementations; a signed CBOR audit log is. The substrate's portability requirement (any operator implementing the schema can verify any other operator's chain) selects CBOR by elimination.

PART III · THE VERIFIER

Browser-resident, offline, deterministic

8. Browser-resident, offline, deterministic verdicts

The Mickai verifier ships as a static web page plus a WebAssembly module. The static page is hosted at mickai.co.uk/audit-verifier and is also distributable as a single HTML file. The operator can supply the chain on a USB drive, on an intranet endpoint, or through an air-gapped transfer; the verifier does not require network access.

The verifier loads the deployment public key, loads the chain file (a sequence of CBOR records), walks the chain record-by-record, validates each parent hash, validates each ML-DSA-65 signature, and emits a verdict per record. The output is a deterministic verdict array; two parties running the same chain through the same verifier version emit identical verdict arrays.

9. VERIFIED, INVALID, STALE, REVOKED

The verifier emits one of four verdicts per record. There is no fifth verdict. There is no probabilistic answer. The chain either holds or it does not.

Verdict	Meaning
VERIFIED	Hash chain and signature both check. The record is canonical.
INVALID	Hash chain or signature fails. The chain has been tampered, truncated, or corrupted.
STALE	Signature was emitted under a key version older than the current rotation. Record is canonical but the key is retired.
REVOKED	Signature was emitted under a key explicitly revoked by the operator before the record's timestamp.

The four-verdict model maps cleanly onto regulator expectations. An incident review begins with the verifier verdict array; any record marked INVALID is the starting point of the investigation. STALE and REVOKED records are canonical at the time they were signed but require additional context (which rotation? which revocation event?) before they enter evidence at a tribunal.

10. Forensics six months later

The substrate property the regulator actually requires is not real-time verification. It is forensics six months later. When the data subject who was acted on at Time T submits an Article 22 challenge at Time T plus six months, the operator must produce the chain, the public key, and the verifier; the data subject (or their representative, or the ICO) walks the chain on their own hardware and arrives at the

same verdict the operator would arrive at. The verdict is deterministic and replayable; the AI vendor in the loop at Time T is no longer in the loop at Time T plus six months, and does not need to be.

The forensics property holds across vendor change, vendor failure, vendor acquisition, and vendor exit. The chain verifies under the operator's key regardless of what happened to the AI vendor that produced the original decision.

PART IV · ADOPTION PATH

How any AI operator adopts the OAR primitive

11. Wrapping any AI vendor's decision-emit hook

The OAR primitive is designed as a wrapper around any AI decision-emit boundary. Whether the underlying inference is OpenAI's API, Anthropic's API, an Azure OpenAI deployment, a self-hosted Llama, a Mistral instance, a Cohere endpoint, or a Mickai SIOS brain, the wrapper interposes at the boundary where the AI produces a committed decision and emits an OAR record before the decision propagates downstream.

The wrapper is small. The reference implementation in TypeScript is approximately 400 lines; the Python reference is approximately 350 lines. The implementation accepts the AI vendor's decision payload, constructs the OAR record, applies CBOR encoding, computes the SHA-3-512 parent hash, delegates the ML-DSA-65 signature to the operator's TPM, appends the signed record to the chain store, and returns control. The operator's existing AI application surface is unchanged above the wrapper.

12. Key custody and rotation

The ML-DSA-65 private key lives in the operator's TPM 2.0, bound to the workstation's platform configuration registers (PCRs) so the key only unseals when the platform boot state matches the expected measurement. Rotation is policy-driven and operator-controlled. The Mickai SIOS ships a default rotation policy (annual rotation, with a 30-day overlap window during which both keys are valid), and operators are free to override with their own policy. Revocation is explicit; the operator issues a revocation record signed under the active key declaring the prior key revoked from a stated timestamp.

The chain anchoring record at each rotation binds the new public key to the chain identifier; verifiers walking the chain across a rotation boundary follow the anchor and continue with the new key. The chain identifier itself does not change.

13. Open-source release plan

The OAR schema, the conformance vectors, the TypeScript reference wrapper, the Python reference wrapper, and the verifier WebAssembly module are scheduled for open-source release under an Apache 2.0 licence following UK IPO acknowledgement of the OAR family within the GB2607309.8 to GB2610422.4 patent corpus. The release establishes the OAR primitive as a portable standard rather than a Mickai-specific implementation; any AI vendor, any AI operator, any regulator's engineering desk can adopt the schema, implement against the vectors, and produce or verify compliant chains.

The patent position is defensive. Mickai's commercial offering is the Mickai Sovereign Intelligence Operating System above the substrate, the Mickai Sovereign Hardware AI Workstation that ships the substrate at the cryptographic primitive layer, and the integration and engagement around adoption. The substrate itself is open by design.

14. Closing

The cryptographic position the AI operator should hold over their agents' actions is not a feature the AI vendor adds. It is a primitive the operator owns. The OAR substrate is that primitive. The chain runs under the operator's key. The verifier runs in any browser. The regulator's chain-of-custody question reduces to walking a deterministic verdict array, not negotiating with a vendor.

> **An AI agent without an OAR chain is a black box. An AI agent with an OAR chain is an accountable participant in a regulated workload.**

Engineering leadership at any UK AI buyer is open to a fifteen-minute substrate briefing at any time. press@mickai.co.uk.

APPENDIX · ABOUT THE AUTHOR

Micky Irons

Founder of Mickai LTD (Companies House 17166618, England and Wales, registered office 20 Wenlock Road, London, N1 7GU). Named inventor on the Mickai SIOS patent corpus, recorded on the UK Intellectual Property Office public register at numbers GB2607309.8 to GB2610422.4. Trade mark Mickai registered at UK00004373277 (classes 9 and 42, filed 15 April 2026).

Before founding Mickai, Micky was a Sellafield site worker. The egress constraint observed from inside the regulated workstation is the engineering origin of the substrate described across the Mickai ebook series.

Profiles and links

mickai.co.uk · the canonical Mickai site.

crunchbase.com/person/micky-irons · founder profile.

linkedin.com/in/mickyirons · personal LinkedIn.

github.com/Micky-CMO · open-source position.

linkedin.com/company/mickai · Mickai LTD company page.

crunchbase.com/organization/mickyirons · Mickai LTD Crunchbase entry.

Email: press@mickai.co.uk

References and further reading

· NIST FIPS 204, Module-Lattice-Based Digital Signature Standard, August 2024.

· NIST FIPS 202, SHA-3 Standard, August 2015.

· RFC 8949, Concise Binary Object Representation (CBOR), December 2020.

· NCSC, Timelines for migration to post-quantum cryptography, and the AI Cyber Security Code of Practice (DSIT consultation 2024 to 2025).

· PRA Supervisory Statement SS1/23, model risk management principles for banks.

· Mickai Audit Verifier, browser-resident reference implementation: mickai.co.uk/audit-verifier.

· Mickai OAR Brain documentation: mickai.co.uk/oar.

· Mickai trade mark UK00004373277, classes 9 and 42, filed 15 April 2026.

Colophon

Set in Inter Tight (Variable) and Inter Black. Cover and body chrome in the Mickai gold-on-void palette. Audit ledger primitives on the cover match the live substrate. Brand voice audited under the Mickai AMT preflight gate; zero violations at publish.