CIPHER NOTA

Privacy-First Encrypted Storage with Secure AI Knowledge Agents

Security Architecture, Cryptographic Design, Multi-Cloud Storage and Secure RAG

Core proposition

Cipher Nota separates storage infrastructure, cryptographic authority and AI access into distinct trust boundaries. Customers retain encrypted storage while granting revocable, least-privilege access to selected content for search, retrieval-augmented generation and AI-agent workflows.

Final Technical White Paper

Version 1.0
15 January 2023
Classification: Public

Trust the cryptographic design, not the storage location.

Document Status and Control Legend

This white paper consolidates the original Cipher Nota product brief, production architecture confirmations supplied by the product owner, and security/AI enhancements proposed for the target platform. To prevent roadmap features from being mistaken for deployed controls, every material design statement is classified using the status model below.

Status	Meaning
VERIFIED	Confirmed by the product owner as part of the production design or current operational model.
TARGET	A secure architectural decision adopted by this white paper as the intended production end state.
RECOMMENDED	A security or product enhancement proposed for implementation and assurance.
CONDITIONAL	A capability that is safe only when a stated trust boundary, deployment mode or customer choice applies.
RISK	A known trust exposure or limitation that must be disclosed and mitigated.

Publication discipline

Only VERIFIED controls should be marketed as currently deployed. TARGET, RECOMMENDED and CONDITIONAL capabilities may be described as architecture direction, roadmap or optional deployment modes until implementation and independent validation are complete.

Contents

1. Executive Summary

Cloud storage provides availability, durability and global accessibility, but it does not by itself remove the storage operator from the data trust model. Cipher Nota addresses that problem by inserting an independent cryptographic and orchestration layer between users and storage infrastructure. The source brief describes client-side encryption, TLS, user-managed key concepts, encrypted virtual drives, multi-provider storage, secure sharing, backup, snapshots and recovery as the product foundation.

The final architecture evolves that foundation into a controlled-key, zero-plaintext operational model. Account Root Keys are maintained in HashiCorp Vault as non-exportable key material; user key protection uses Argon2id; authenticated symmetric encryption uses AES-256-GCM; post-quantum key establishment uses ML-KEM-1024; user virtual drives are protected by LUKS and CryFS; and customer recovery requires user-controlled recovery material. The platform is designed so that ordinary storage systems, external cloud providers and routine application components do not need direct access to customer plaintext.

VERIFIED | Production cryptographic controls

Argon2id is the production password KDF. Account Root Keys are maintained in HashiCorp Vault and configured non-exportable. AES-GCM is the authenticated encryption mode, with Vault aes256-gcm96 used for AES-256-GCM operations. ML-KEM is deployed, with the reported ML-KEM-1024 key and ciphertext dimensions matching FIPS 203 parameter sizes.

RISK | Vault is part of the trusted cryptographic computing base

Non-exportable does not mean Vault is irrelevant to decryption. Authenticated identities permitted to invoke Vault cryptographic operations can still cause authorised cryptographic operations. Cipher Nota therefore should not represent the current model as pure client-only zero knowledge. The defensible description is zero-plaintext operational architecture with controlled key access.

The strategic expansion proposed in this paper adds a second product layer: customer-controlled secure AI. A customer may select specific files or folders, create an isolated AI Workspace and grant a secure agent time-limited, read-scoped access to only that collection. The resulting agent can semantically search, cite, summarise, compare and reason over private content without receiving the customer Account Root Key or unrestricted access to the Virtual Drive.

The preferred AI model is a capability-based architecture. AI access becomes a revocable cryptographic grant rather than an implicit consequence of storing data with Cipher Nota. Secure RAG, encrypted or isolation-protected vector stores, content provenance, source citations, prompt-injection controls, an agent policy gateway, short-lived workload identities, human approval for side effects and optional confidential computing collectively extend Cipher Nota from encrypted Storage-as-a-Service to encrypted Knowledge-and-Agent-as-a-Service.

Strategic outcome

Cipher Nota can differentiate on a unified proposition: encrypted multi-cloud storage plus privacy-preserving AI intelligence, with explicit customer control over which data an AI system may process, for what purpose and for how long.

2. Product Thesis and Strategic Positioning

2.1 The cloud trust problem

Provider-side encryption is valuable, but the provider may still operate the surrounding identity, application and key-management systems. Cipher Nota changes the trust arrangement by separating the entity storing the ciphertext from the authority governing how that ciphertext can be decrypted. The storage service becomes primarily an availability boundary; the cryptographic control plane becomes the confidentiality boundary.

2.2 Product definition

Recommended product definition

Cipher Nota is a privacy-first encrypted storage and knowledge platform that orchestrates user data across encrypted virtual drives and supported cloud storage while controlling access through dedicated cryptographic, identity and AI authorization boundaries.

The original brief correctly warns against positioning Cipher Nota as merely another cloud drive. Its differentiator is the encryption and orchestration layer that can sit above existing cloud infrastructure, combined with user-specific encrypted virtual drives and secure data lifecycle controls.

2.3 From storage to knowledge

The AI extension should not be marketed as an unrestricted chatbot over a customer account. It should be presented as a controlled knowledge service. Customers create bounded AI collections, grant selected files to an agent, and receive answers grounded in those authorised sources. Storage remains the system of record; AI becomes a temporary or persistent interpretation layer operating under explicit policy.

TARGET | Product category

Evolve from encrypted Storage-as-a-Service to encrypted Knowledge-and-Agent-as-a-Service while preserving the principle that AI access is separate from storage ownership and independently revocable.

3. Security Objectives and Design Principles

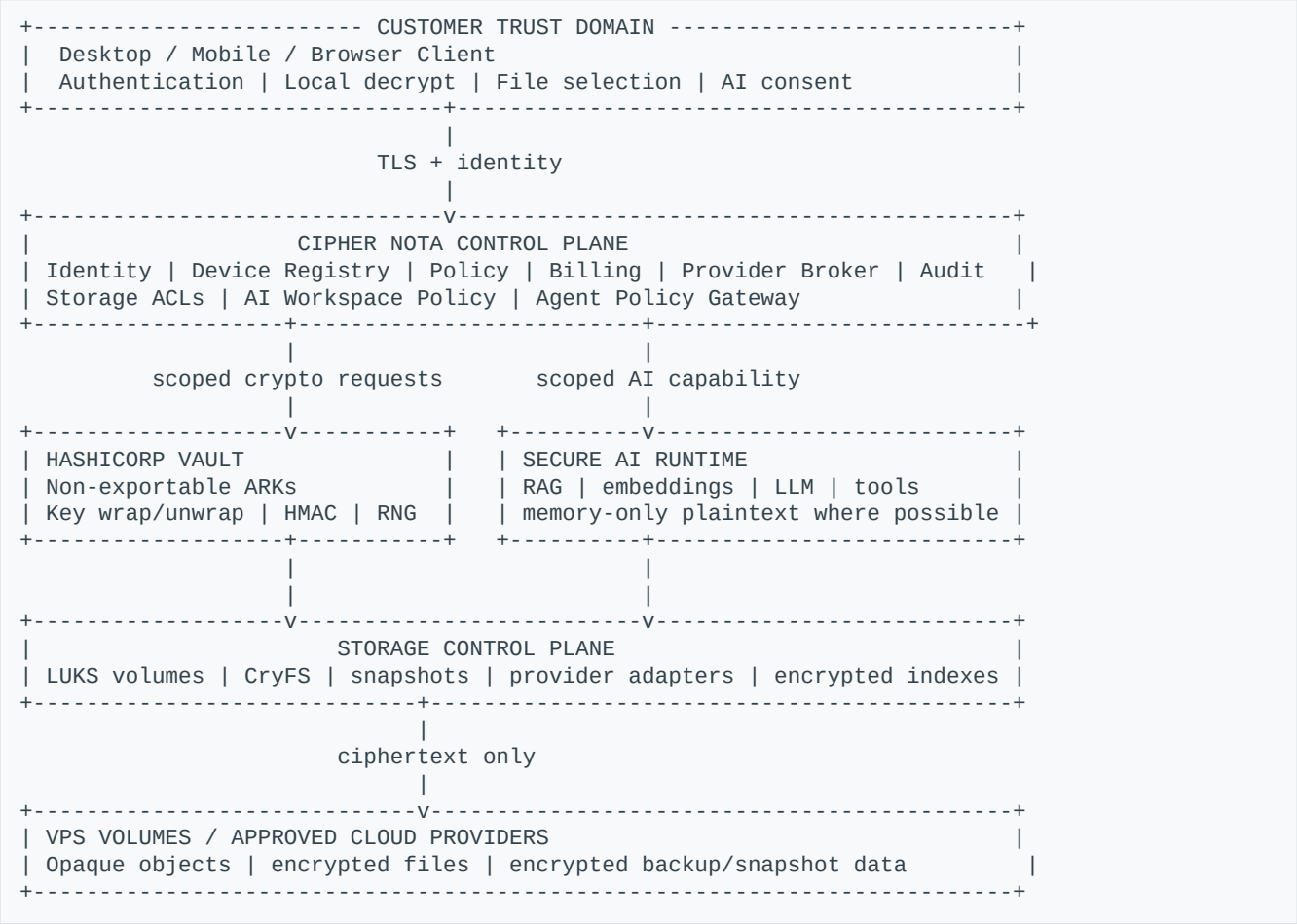
Principle	Security objective
Confidentiality	Plaintext file content and unwrapped content keys should be exposed only inside explicitly authorised client or secure processing contexts.
Integrity	Authenticated encryption, version manifests and signed policy objects should make undetected modification, substitution and rollback materially harder.
Least privilege	Every human, workload, device, cloud provider and AI agent receives only the minimum resource-level authority required.
Key separation	Authentication keys, Account Root Keys, file DEKs, device keys, sharing keys, filesystem keys, recovery material, release-signing keys and AI Workspace keys are distinct.
Crypto agility	Encrypted objects identify suite and key versions so algorithms can migrate without redesigning the platform.
Recoverability	Recovery is available to the customer without giving Cipher Nota independent recovery authority.
Revocability	Devices, sessions, provider tokens, shares and AI workspaces can be independently revoked.
Metadata minimisation	Routine metadata collection is minimised, purpose-bound, access controlled and subject to retention.
Auditability	High-impact cryptographic, administrative and agent actions are attributable and reviewable without logging customer plaintext.
Explicit trust	Vault, user endpoints, AI runtimes, storage providers and administrative systems are separate trust domains with documented assumptions.

3.1 Zero trust as an authorization model

Cipher Nota should apply zero-trust principles to service-to-service, device and AI access: no implicit trust based on network location; explicit workload identity; resource-specific policy; short-lived credentials; and continuous verification for sensitive actions. NIST SP 800-207 provides the reference conceptual model for this approach [R5].

4. Trust Boundaries and Reference Architecture

Figure 1. Proposed Cipher Nota security and AI trust domains



4.1 Primary trust assumptions

- HashiCorp Vault is trusted to enforce key non-exportability, access policy, cryptographic operations and auditability.
- Customer endpoints are trusted while unlocked; endpoint compromise remains a residual risk.
- External storage providers are not trusted with customer plaintext or user content keys.
- AI runtimes are trusted only for explicitly granted workspaces and only for the lifetime and capabilities of their grant.
- A model itself is never treated as an authorization engine. Deterministic application policy decides whether data or tools may be accessed.
- Administrative access is privileged, named, audited and separated from customer encryption authority where practical.

5. Cryptographic Architecture and Key Management

5.1 HashiCorp Vault

VERIFIED | Account Root Key protection

Account Root Keys are stored and managed in HashiCorp Vault and configured non-exportable. Application servers do not persist plaintext Account Root Keys. The secure target is Vault-native key generation or wrapped import so plaintext ARKs are never introduced into general application memory during provisioning.

Vault Transit is suitable as the cryptographic control plane because it provides cryptography-as-a-service, key rotation, HMAC, random generation and envelope-encryption workflows while retaining keys within the Vault trust boundary [R1][R2]. Exportability is disabled for production ARKs.

TARGET | ARK generation

Where the current generation location is not independently evidenced, the required target design is to generate ARKs directly within Vault or import them only through a wrapped key-import mechanism. This eliminates an unnecessary plaintext key-generation path.

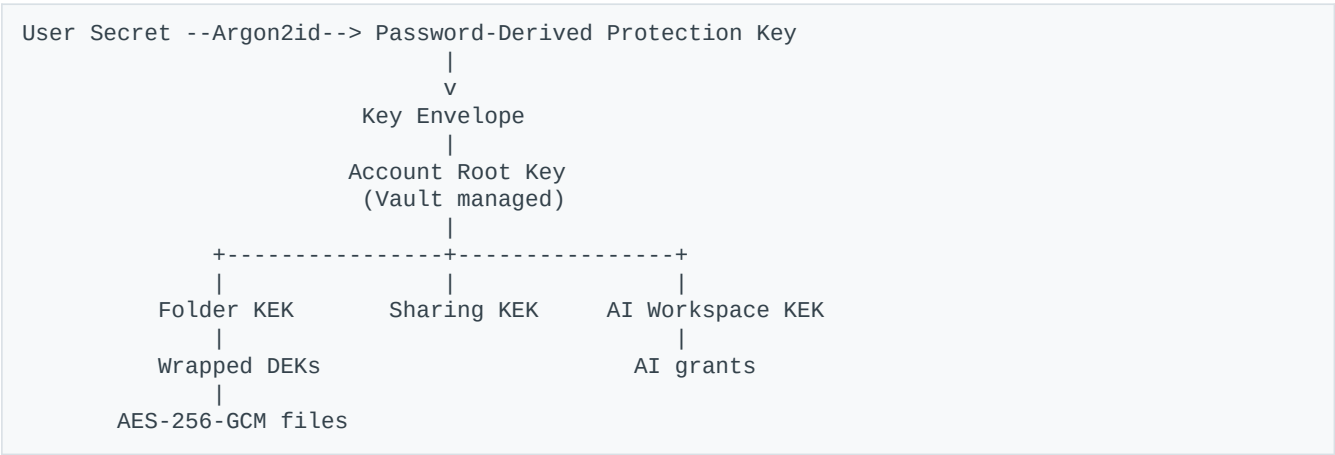
5.2 Password KDF

VERIFIED | Argon2id

Argon2id is the production password KDF. It should use a unique random salt per credential/key envelope, versioned cost parameters and parameter tuning based on supported endpoint classes. RFC 9106 provides the reference definition and test vectors [R3].

5.3 Envelope encryption and key hierarchy

Figure 2. Recommended logical key hierarchy



The preferred file design uses per-file or appropriately scoped Data Encryption Keys (DEKs). Higher-level keys wrap DEKs instead of encrypting every large object directly with a single account key. This reduces blast radius, supports re-wrapping during recovery and enables selective sharing and revocation. The design aligns with envelope-encryption patterns documented by HashiCorp and general key-lifecycle guidance in NIST SP 800-57 [R2][R10].

5.4 AES-256-GCM and nonce management

VERIFIED | Authenticated encryption mode

AES-GCM is the deployed authenticated encryption mode. Vault uses the aes256-gcm96 profile, which combines a 256-bit AES key with a 96-bit nonce [R1].

Nonce uniqueness is a critical safety property. For non-convergent encryption, Vault-generated/random nonce handling should remain the default. Key rotation thresholds should be conservative and monitored by key version. Convergent encryption should not be enabled for ordinary customer content because it creates equality leakage and imposes stricter caller-side nonce/context requirements.

5.5 Directory and filename protection

VERIFIED | Filename encryption choice

Filenames can be encrypted based on user choice.

RECOMMENDED | Privacy-default filename mode

Make encrypted filenames the default. If a compatibility mode leaves filenames visible, the UI should explicitly state the privacy consequence before the customer enables it.

VERIFIED | Directory lookup

Directory structures are currently represented through a key/value mapping based on a hash of the directory path.

TARGET | Keyed directory identifiers

Replace bare deterministic path hashing with a keyed construction such as HMAC-SHA-256 over a canonicalised path and tenant-specific metadata key. This prevents inexpensive dictionary recovery of common paths such as /Documents or /Invoices.

5.6 ML-KEM-1024

VERIFIED | Post-quantum KEM deployment

ML-KEM is deployed. The reported ML-KEM-1024 dimensions are 1,568-byte encapsulation keys, 3,168-byte decapsulation keys and 1,568-byte ciphertexts, matching the ML-KEM-1024 parameter sizes in NIST FIPS 203 [R4].

The public claim should be “ML-KEM-1024 conforming to FIPS 203” only when implementation testing verifies the complete standard, including key generation, encapsulation, decapsulation, validation, randomness and error behaviour. Matching sizes alone are not sufficient evidence of conformance.

5.7 RSA-2048

VERIFIED | Current RSA roles

RSA-2048 is used primarily for digital signatures and secure server-channel authentication.

RECOMMENDED | Protocol simplification

Use established TLS for server-to-server secure channels rather than custom RSA key exchange. Maintain RSA only where compatibility or signature requirements justify it, and plan independent migration of signature algorithms through crypto agility.

5.8 Cryptographic key classes

Key class	Purpose	Protection	Status
Account Root Key	Account root authority	Vault; non-exportable	Verified
Password-derived key	Protect user key envelope	Client/secure auth flow	Verified
File DEK	Encrypt object content	Wrapped; object-scoped	Target
Folder/metadata key	Compartmentalise metadata	Wrapped	Target
Device key	Device cryptographic authority	Vault-generated current model	Verified
Recovery key	Customer recovery authority	Customer controlled	Verified
Sharing key	Protect shared content	Scoped share context	Verified/Target
LUKS MEK	Encrypt block device	Storage boundary	Verified
CryFS key	Encrypt filesystem layer	Storage boundary	Verified
ML-KEM keypair	Post-quantum key establishment	Versioned crypto suite	Verified
Release signing key	Software provenance	Protected signing service	Verified/Target
AI Workspace Key	Bound AI collection authority	Vault; workspace-scoped	Target

6. Virtual Drive, Multi-Cloud Storage and Resilience

6.1 Virtual Drive

VERIFIED | Virtual Drive implementation

Each user receives a LUKS-protected partition. User files are stored in the user Virtual Drive, with CryFS providing an additional filesystem encryption layer. Virtual-drive partitions are distributed across VPS volumes.

Figure 3. Layered Virtual Drive protection

```
User Virtual Drive
      |
    CryFS              <- file/folder level encrypted filesystem
      |
    LUKS               <- block-device encryption boundary
      |
VPS storage volume
```

The two layers should use independent key material. LUKS primarily protects block media and VPS volume exposure, while CryFS protects logical filesystem content and metadata. Client-side/object encryption remains the primary content-confidentiality mechanism; server-side layers are defence in depth rather than the basis for the customer privacy claim.

6.2 Provider abstraction layer

The source brief lists Google Drive, Dropbox, Mega, OneDrive, AWS S3 and iDrive among supported or represented providers. The product should expose those integrations through a normalised provider adapter that handles upload, retrieval, deletion, versioning, quota, retries, integrity validation, region policy and provider authentication.

VERIFIED | OAuth token protection

Provider OAuth tokens are protected through transport encryption, cryptographic binding and strict storage protocols.

RECOMMENDED | Provider credential isolation

Store or wrap provider credentials through a dedicated Vault namespace, isolate decrypt privileges from the general web tier, minimise OAuth scopes, rotate tokens and support immediate customer revocation.

6.3 Multi-provider resilience

RECOMMENDED | Replication and backup modes

Offer encrypted replication to multiple providers or a primary-plus-encrypted-backup model. Replication should move ciphertext; it should not require Cipher Nota to materialise customer plaintext.

```
            Encrypted Object
                /      \
               /        \
      Provider A          Provider B
      (primary)           (replica/backup)
```

6.4 Ransomware and rollback resilience

- Immutable or retention-locked encrypted snapshots where the storage backend supports them.
- Mass-delete and mass-rewrite anomaly detection.
- Device quarantine when a newly enrolled or anomalous device rewrites large portions of a vault.
- Authenticated version manifests and monotonic revision identifiers to detect rollback to older but cryptographically valid ciphertext.
- Known-good snapshot restoration without decrypting content in the storage control plane.

7. Identity, Devices, Recovery and Secure Sharing

7.1 Authentication

The product currently uses email/password, two-factor authentication and recovery flows. The recommended progression is password plus TOTP as the baseline, passkeys/WebAuthn as the preferred phishing-resistant path, and hardware-backed authenticators for high-assurance accounts. Authentication authority must remain separate from content-decryption authority.

RECOMMENDED | Passkeys/WebAuthn

Adopt WebAuthn-compatible public-key credentials as the preferred sign-in factor as the product matures [R9].

7.2 Device keys and enrolment

VERIFIED | Device key generation

Device keys are currently generated using HashiCorp Vault.

RECOMMENDED | Hardware-backed device identity

For higher assurance, generate the device private key on the endpoint when a Secure Enclave, TPM, Android Keystore or equivalent protected keystore is available. Register only the public key and attested device metadata with Cipher Nota. Vault may remain the server-side wrapping/policy authority.

7.3 Lost-device revocation

VERIFIED | Recovery-key revocation

A lost device is revoked using the recovery key provided by the user.

VERIFIED | Historical access limitation

A revoked device may still decrypt historical content for which it previously obtained sufficient key material.

Revocation therefore has two levels. Access revocation terminates sessions, refresh tokens, new key retrieval and future synchronization. Cryptographic revocation additionally rotates affected wrapping keys and, when necessary, re-encrypts content for which old file keys are assumed compromised.

7.4 Account recovery

VERIFIED | Current recovery behaviour

Account recovery regenerates the master key and re-encrypts previously stored files using the new master-key context. Cipher Nota cannot perform recovery independently; customer recovery material is required.

TARGET | DEK re-wrapping recovery

For ordinary recovery where data-encryption keys are not suspected compromised, re-wrap existing DEKs under new account protection rather than decrypting and re-encrypting all bulk content. Reserve full content re-encryption for actual or suspected key compromise. This reduces I/O, plaintext exposure and recovery time.

7.5 Secure sharing

VERIFIED | Recipient public-key sharing

Recipient public-key sharing is implemented.

VERIFIED | Password-assisted sharing

Sharing passwords are processed through a Vault-backed secure API.

TARGET | Password sharing hardening

Passwords must not become raw encryption keys. Derive protection material using a memory-hard KDF, combine with a high-entropy capability token, rate-limit attempts, allow expiry/revocation, and avoid placing decryption secrets in ordinary URL query parameters.

RECOMMENDED | Recipient-key verification

For high assurance, support key fingerprints, QR verification or a transparency-backed key directory to reduce the risk of malicious recipient-key substitution.

8. Privacy, Metadata, Logging, Residency and Deletion

8.1 Metadata boundary

VERIFIED | Routine retained metadata

The production architecture retains the user IP address as routine user-origin metadata.

VERIFIED | Retention

IP metadata may be retained for up to 365 days for active accounts and is deleted when a user remains inactive for three months.

VERIFIED | Access

Only privileged system administrators are authorised to access retained metadata.

RECOMMENDED | Privacy retention tiers

Reduce routinely accessible IP history where operationally possible: short hot retention for security operations, restricted archive thereafter, and earlier deletion/pseudonymisation where legal and abuse-prevention requirements allow.

8.2 Logging

VERIFIED | Filename logging posture

Filenames and object names are not intentionally logged, except that they may appear in error paths.

TARGET | Error-log redaction

Implement structured logging and redaction so customer filenames, object names, secrets, file content, recovery values and cryptographic material cannot persist in routine error logs.

8.3 Non-US data residency

VERIFIED | Residency capability

The non-US storage requirement is technically enforceable.

To make this auditable, the product should implement explicit region allow-lists, infrastructure policy-as-code, provider-region validation and deployment evidence. For customer-connected third-party providers, the claim should be scoped to approved regions and configurations rather than applied universally to every provider account a customer might connect.

8.4 Snapshot deletion and cryptographic erasure

VERIFIED | Snapshot disposal

Snapshots are deleted using Linux operating-system commands followed by storage scrambling.

VERIFIED | Cryptographic erasure

Cryptographic erasure is performed by securely deleting or overwriting the Media Encryption Key used to protect the storage device.

TARGET | Layered sanitisation

Treat key destruction as the primary cryptographic-erasure mechanism and logical deletion/storage sanitisation as additional controls. Verify that all usable MEK copies, LUKS key slots, replicas, recovery copies and disaster-recovery copies are destroyed before declaring cryptographic erasure complete. NIST SP 800-88 Rev. 2 provides current sanitisation guidance [R8].

9. Secure Software Supply Chain and Independent Assurance

9.1 Release signing

VERIFIED | Signed client releases

Cipher Nota client releases are signed and validated using Cipher Nota developer/release public verification keys. The corresponding private signing keys must remain protected and separate from ordinary developer credentials.

TARGET | Release-key isolation

Protect release signing keys through Vault/HSM controls, require dual approval for production signing, retain provenance and support emergency revocation/rotation.

9.2 Third-party code

VERIFIED | Third-party scripts

Third-party scripts are present; they are open source and have been audited.

RECOMMENDED | Supply-chain hardening

Pin dependency versions, generate SBOMs, minimise third-party JavaScript in cryptographic interfaces, use restrictive Content Security Policy, enforce code review, scan dependencies, verify build provenance and adopt reproducible builds where practical. NIST SP 800-218 provides a secure-development baseline; SP 800-218A extends secure-development practices for generative AI systems [R11][R12].

9.3 Independent review

VERIFIED | Independent cryptographic review

The cryptographic implementation has been independently reviewed by a contracted auditing team.

RECOMMENDED | Public assurance evidence

Publish a non-sensitive audit summary identifying auditor, scope, date, methodology, high-level findings, remediation status and re-test status. Repeat assessment after material cryptographic or AI authorization changes.

10. Secure AI Product Vision

AI should extend the value of encrypted storage without silently expanding the trust boundary. A normal RAG system cannot semantically search ciphertext in the general case; selected content must become readable somewhere to be parsed, embedded and supplied to a model. Cipher Nota should therefore make the AI plaintext boundary explicit and customer controlled.

Core AI principle

AI access is a revocable cryptographic grant - not an implicit consequence of storing data with Cipher Nota.

10.1 Proposed product name and experience

A customer-facing capability such as “Cipher Nota AI Vault”, “Private Knowledge Agent” or “Secure AI Workspace” can represent this boundary. The customer creates a workspace, selects files/folders, chooses capabilities, selects an AI privacy mode, chooses retention and sets an expiry. Files outside the grant remain inaccessible to the agent.

10.2 Default AI posture

- Read-only by default.
- No entire-account indexing by default.
- No model training on customer content by default.
- No external model provider unless explicitly selected or policy approved.
- No delete, share, move or external-communication capability unless separately granted.
- Source-grounded answers with document citations and version identifiers.
- Workspace key and index destruction when AI access is revoked.

11. AI Workspace and Revocable Cryptographic Grants

11.1 AI Workspace object

Example AI Workspace policy

```
AI Workspace: Project Phoenix
-----------------------------------------------------------
Files/Folders:   selected objects only
Agent:           phoenix-analyst-01
Access:          READ + SEARCH + RAG + SUMMARISE + COMPARE
Write:           disabled
External share:  disabled
Expiry:          24 hours
Memory:          session-only
AI mode:         Confidential AI
-----------------------------------------------------------
```

The workspace is a first-class policy object, not merely a chat session. It records tenant, workspace ID, allowed object IDs, permissions, model policy, retention, expiry, classification and optionally tool permissions. Filenames do not need to appear in bearer tokens; opaque object identifiers should be used.

11.2 AI Workspace Key

TARGET | Separate AI cryptographic authority

Create an AI Workspace Key or AI Access Key that is scoped to one workspace. The agent never receives the Account Root Key. Vault mediates or wraps the key material necessary for only the granted object set.

```
Account Root Key
      |
      +-- Storage keys
      |
      +-- AI Workspace A Key --> grants to Object 44, 51, 93
      |
      +-- AI Workspace B Key --> grants to Object 102, 130
```

11.3 Capability token

An AI workload should obtain a short-lived signed capability that identifies the workspace, workload identity, permissions, object IDs, model policy and expiry. Capability validation occurs before retrieval and again before any tool invocation. The capability must not contain customer decryption keys or plaintext paths.

12. Secure RAG Architecture

Figure 4. Secure RAG ingestion and query lifecycle

```
Customer selects files
        |
Create AI Workspace + policy
        |
Short-lived workload identity
        |
Retrieve authorised ciphertext
        |
Scoped key operation / unwrap
        |
Decrypt only authorised content inside approved runtime
        |
Parse -> chunk -> classify -> embed
        |
Tenant/workspace-isolated vector + chunk store
        |
Query embedding -> ACL filter -> similarity search
        |
Retrieve minimal authorised chunks
        |
LLM inference
        |
Output validation + source citations
        |
Customer answer
```

12.1 Ingestion controls

- Virus/malware and file-type validation before parsing.
- Document parser isolation and resource limits.
- Chunk provenance linking every chunk to tenant, workspace, object, document version and location.
- Classification labels inherited from source files where available.
- No cross-tenant deduplication that creates semantic or access leakage.
- Embedding-model version recorded so re-indexing is deterministic and auditable.
- Raw chunks encrypted at rest and temporary plaintext removed after processing.

12.2 Query controls

Authorization must occur before the LLM sees content. The retrieval layer first restricts candidate objects by tenant, workspace, document ACL and policy, then performs similarity search over that authorised set, then validates each retrieved chunk again before context construction. The system prompt is not an authorization mechanism.

12.3 Grounded answers and citations

TARGET | Storage-native citations

Every RAG answer should cite the underlying Cipher Nota object, version and location (page, section, paragraph, spreadsheet range or transcript timestamp). Clicking a citation should open the encrypted source through the customer authorization path. This creates provenance and makes hallucinations easier to detect.

13. AI Privacy Modes and Confidential Computing

Mode	Plaintext processing location	Privacy posture	Use case
Private Local AI	Customer endpoint	Highest; plaintext need not leave endpoint	High-sensitivity personal or regulated data
Cipher Nota Confidential AI	Attested isolated compute/TEE	High; server-side processing with data-in-use protection	Premium secure RAG and agent workloads
Approved External AI	Approved third-party model endpoint	Provider-dependent; explicit policy/consent required	Customers prioritising model capability or BYO provider

13.1 Private Local AI

The customer device decrypts selected files, computes embeddings and runs retrieval and inference locally. Cipher Nota continues to provide encrypted storage and workspace policy but does not require plaintext AI context. This is the preferred maximum-privacy mode where device resources permit.

13.2 Cipher Nota Confidential AI

RECOMMENDED | Confidential-compute AI runtime

Run secure RAG and model inference inside an attested Trusted Execution Environment where supported. Vault or the key broker should release workspace decryption authority only after validating workload identity, measurement, environment policy and freshness of attestation. NIST IR 8320E (initial public draft, 2026) specifically addresses confidential computing for cloud workloads and AI use cases [R7].

Confidential computing is a recommended target, not a current production claim unless independently implemented and attested. The design should also account for side-channel limitations, host I/O, debug modes, image provenance and secrets released to the enclave.

13.3 Approved external AI

CONDITIONAL | Third-party model use

Permit only when the customer or enterprise policy explicitly allows it. Enforce provider allow-lists, data-classification rules, contractual no-training/no-retention requirements where applicable, regional routing, minimal RAG context, and disclosure of what content will leave Cipher Nota-controlled compute.

14. Vector, Embedding and Retrieval Security

Embeddings are derived from customer content and must be treated as sensitive. They are not harmless metadata. OWASP includes vector and embedding weaknesses among the 2025 LLM application risks because isolation, access-control and data-poisoning failures can undermine RAG systems [R14].

14.1 Storage model

TARGET | Tenant/workspace isolation

Every vector record is bound to tenant ID, workspace ID, object ID, document version, chunk ID, classification, embedding-model version and ACL metadata. Cross-tenant retrieval is structurally prevented rather than left to prompt instructions.

TARGET | Encryption at rest

Encrypt vector indexes and raw chunk stores at rest. For ordinary vector databases, similarity search may require vectors to be readable inside the authorised retrieval engine; do not claim fully homomorphic or searchable ciphertext unless such a scheme is actually implemented. Confidential AI mode can move the retrieval engine itself into the attested runtime.

14.2 Secure retrieval ordering

1. Authenticate the requesting user/workload.
2. Resolve the AI Workspace and verify expiry/revocation.
3. Apply tenant, workspace, object and classification ACL filters.
4. Perform vector similarity search only within the authorised candidate set.
5. Re-check each selected chunk before decrypting context.
6. Provide the minimum relevant context to the model.
7. Return grounded citations that map to the authorised source version.

15. Prompt Injection and Agent Security

15.1 Retrieved documents are hostile input

A PDF, email, webpage, source file or document may contain instructions intended to manipulate the model. That is an indirect prompt-injection path. Retrieved content must always be treated as untrusted data, never as policy. OWASP identifies prompt injection as LLM01:2025 and excessive agency as a separate risk category [R13][R15].

Security rule

SYSTEM POLICY is authoritative. USER REQUEST is constrained by policy. RETRIEVED CONTENT is untrusted evidence. The model may interpret retrieved content, but it may not elevate that content into new permissions.

15.2 Agent Policy Gateway

Figure 5. Deterministic agent authorization

The LLM never decides whether it is authorised to delete, share, email, move or export data. The agent proposes an action; deterministic policy evaluates tenant, workspace grant, tool allow-list, object ACL, data classification, requested destination, session risk and approval requirements.

15.3 Capability taxonomy

Capability	Default	Control
AI_READ	Allow in authorised workspace	Read selected objects only
AI_SEARCH	Allow	Workspace/ACL filtered retrieval
AI_SUMMARISE	Allow	No side effects
AI_COMPARE	Allow	No side effects
AI_CLASSIFY	Allow if configured	Writes labels only after policy
AI_CREATE_FILE	Deny by default	Explicit grant; confined output folder
AI_MOVE_FILE	Deny	Explicit grant + human confirmation
AI_SHARE_FILE	Deny	Explicit grant + destination approval

Capability	Default	Control
AI_DELETE_FILE	Deny	High-risk approval and recovery window
AI_EXTERNAL_COMMUNICATION	Deny	Explicit connector/provider allow-list

15.4 Human approval

Read-only RAG should be the initial product. Any action that changes storage state or sends data outside the workspace should require a separately granted capability, and high-impact actions should require a human confirmation step. This materially reduces excessive-agency risk.

16. AI Memory, Revocation, Audit and Governance

16.1 Memory tiers

Memory type	Retention	Customer control
Session memory	Expires with session or short TTL	Default; encrypted; no permanent profile
Workspace memory	Optional workspace lifetime	Visible, editable and deletable
Permanent memory	Explicit opt-in only	Specific items selected by customer; separately revocable

Memory should never become a hidden copy of the customer's vault. Stored memory is an independent data class with its own encryption, retention, export and deletion controls.

16.2 Cryptographic AI revocation

TARGET | Destroy the AI knowledge boundary

When a workspace is revoked, terminate sessions, invalidate capability tokens, revoke workload identity, destroy or retire the AI Workspace Key, delete vector indexes and raw cached chunks, invalidate model-context caches and remove workspace memory according to policy. Original encrypted files remain unchanged.

16.3 AI audit events

- AI_WORKSPACE_CREATED
- AI_FILE_GRANTED
- AI_FILE_REVOKED
- AI_QUERY_EXECUTED
- AI_DOCUMENT_RETRIEVED
- AI_MODEL_INVOKED
- AI_TOOL_REQUESTED
- AI_TOOL_APPROVED
- AI_TOOL_DENIED
- AI_WORKSPACE_DESTROYED

Routine security logs should record identities, object IDs, model/runtime, event type, counts, decision and timing - not full prompts, plaintext chunks, encryption keys or sensitive answers unless the customer explicitly chooses content logging for a regulated workflow.

16.4 AI governance

NIST AI 600-1 provides a Generative AI Profile for applying AI risk-management practices across governance, measurement and management [R6]. Cipher Nota should combine that governance baseline with NIST SSDF/SSDF-A practices for AI software development [R11][R12] and OWASP GenAI guidance for application-level risks such as prompt injection, excessive agency and vector weaknesses [R13-R15].

- Maintain an AI system inventory: model, embedding model, provider, version, data classes, capabilities and owner.
- Pre-deployment security evaluation for each new model/provider and major prompt/tool-policy change.

- Red-team indirect prompt injection, cross-tenant retrieval, data exfiltration, excessive agency and poisoned documents.
- Define incident response for AI data exposure, malicious tool actions, compromised model providers and contaminated indexes.
- Provide enterprise policy controls for model allow-lists, data classifications, retention and external processing.

17. AI Product Feature Catalogue

Feature	Customer value	Roadmap
Ask My Files	Natural-language questions over selected authorised files	Phase 1
Semantic Search	Find concepts rather than exact keywords or filenames	Phase 1
Document Summaries	Summaries with storage-native citations	Phase 1
Cross-Document Analysis	Compare contracts, reports, versions or policies	Phase 1
Private Knowledge Agent	Persistent read-only agent over an AI Workspace	Phase 1/2
Contract Agent	Identify clauses, dates, obligations and conflicts	Phase 2
Compliance Agent	Locate control evidence and policy references	Phase 2
Research Agent	Synthesize selected research with citations	Phase 2
Finance Agent	Analyse selected invoices/reports; read-only initially	Phase 2
Secure Coding Agent	Search selected private architecture/source documentation	Phase 2
Sensitive Data Detection	Identify PII/secrets in opted-in content	Phase 2
AI Smart Tags	Generate encrypted semantic tags	Phase 2
Duplicate Detection	Find duplicate and near-duplicate material	Phase 2
Secure OCR + RAG	Extract and query scanned documents	Phase 2
Meeting Knowledge Vault	Transcribe and query opted-in recordings	Phase 3
Backup Intelligence	Detect unusual changes and destructive patterns	Phase 3
Storage Optimisation Agent	Recommend archive/retention actions; no auto-delete by default	Phase 3
Share Assistant	Explain exposure and policy before sharing content	Phase 3

17.1 AI Collections

AI Collections should be the primary product interaction. A collection is an authorised subset of the customer's encrypted storage, coupled to a specific agent policy and privacy mode. Separate collections can support Legal, Finance, Engineering or personal use without granting a universal agent access to the whole vault.

17.2 Model gateway

TARGET | Model abstraction layer

Route model calls through a Cipher Nota AI Gateway. The gateway enforces tenant/model policy and can route to local models, Cipher Nota-managed models, confidential-compute models, customer-supplied models or approved external providers. Data policy - not model preference - determines whether a request may leave the Cipher Nota-controlled environment.

17.3 Visible AI data boundary

Before AI access starts, the interface should show selected files, model/privacy mode, permissions, retention, expiry, external-processing status and whether workspace memory is enabled. This makes the privacy transition explicit and supports informed customer consent.

18. Threat Model and Risk Registers

18.1 Core platform threat model

Risk	Severity	Primary treatment
Vault compromise or policy abuse	Critical	Network isolation, non-exportable keys, least privilege, MFA/PAM, audit, dual control
Endpoint compromise	Critical	Hardware-backed device keys, endpoint security, revocation, user education
Recovery-key theft	Critical	Recovery hardening, monitoring, optional multi-factor recovery
GCM nonce misuse	Critical	Vault-managed nonce, monitored key rotation, no ad-hoc convergent mode
Release-signing compromise	Critical	Vault/HSM signing, dual approval, provenance, emergency rotation
Revoked device retains old data	High	Access revocation plus optional cryptographic rotation/re-encryption
Directory hash dictionary attack	High	Keyed HMAC directory identifiers
OAuth/provider token theft	High	Vault protection, narrow scopes, rotation/revocation
Rollback/ransomware	High	Version manifests, immutable snapshots, anomaly detection
Metadata/privacy over-retention	Medium	Purpose limitation, tiered retention, deletion/pseudonymisation

18.2 AI threat model

AI risk	Severity	Primary treatment
Indirect prompt injection	Critical	Treat retrieved content as data; policy gateway; tool isolation; red-team tests
Cross-tenant vector retrieval	Critical	Tenant/workspace ACL before similarity search; isolation tests
Excessive agent agency	Critical	Read-only default; capability allow-list; human approval
AI provider data leakage	High	Explicit policy, provider allow-list, no-training/no-retention contracts, minimal context
Embedding leakage	High	Sensitive-data classification, encryption at rest, isolated retrieval engine
Poisoned documents/index	High	Source provenance, ingestion scanning, trust labels, re-index controls

AI risk	Severity	Primary treatment
AI memory leakage	High	Encrypted scoped memory, visibility, deletion and TTL
Model/tool supply-chain compromise	High	Gateway, version pinning, provider assessment, SSDF-A controls
Hallucinated claims	Medium	Grounded citations, confidence UX, source preview
Prompt/answer logging exposure	High	Metadata-only security logs by default; content logging opt-in

18.3 Residual risks

- An unlocked or malware-compromised endpoint may expose plaintext currently available to the user.
- A revoked device cannot be guaranteed to erase plaintext or keys it previously obtained legitimately.
- Vault is a high-impact trusted cryptographic component; compromise combined with sufficient authorization may enable sensitive cryptographic operations.
- Confidential computing reduces but does not eliminate risk; side channels, workload identity, host I/O and software vulnerabilities still matter.
- AI models may produce incorrect output even when retrieval is accurate; citations provide evidence but do not guarantee reasoning correctness.
- Traffic timing, object size and usage patterns can remain observable even when content is encrypted.
- Loss of required recovery material may make customer data unrecoverable by design.

19. Security Invariants and Publication Claims

1. Application databases and ordinary application servers do not persist plaintext Account Root Keys.
2. Production ARKs are non-exportable from Vault.
3. Passwords are not used directly as symmetric encryption keys; Argon2id is the production password KDF.
4. AES-GCM nonce uniqueness is treated as a mandatory invariant.
5. Authentication, storage encryption, recovery, device identity, sharing and AI access use separate authorities/key classes.
6. Devices can be revoked; revocation blocks future authority but does not promise remote erasure of historical plaintext.
7. Customer recovery requires customer-controlled recovery material; Cipher Nota cannot independently perform recovery.
8. Cloud/VPS storage receives encrypted content under protected workflows.
9. Provider credentials do not become customer data-decryption keys.
10. AI agents never receive the Account Root Key.
11. AI retrieval is restricted by tenant/workspace/object policy before model context is constructed.
12. The LLM is not an authorization engine.
13. External AI processing is explicit and policy controlled.
14. AI workspaces are independently revocable and their derived indexes/memory are deletable.
15. High-impact agent actions require explicit capability grants and, where appropriate, human approval.

19.1 Approved positioning

- Privacy-first encrypted multi-cloud storage.
- AES-256-GCM authenticated encryption.
- Argon2id password-based key derivation.
- HashiCorp Vault-managed non-exportable Account Root Keys.
- ML-KEM-1024 post-quantum key encapsulation, subject to verified FIPS 203 conformance wording.
- Encrypted LUKS/CryFS Virtual Drives.
- Recipient public-key encrypted sharing.
- Customer-controlled recovery authority.
- Secure AI Workspaces with least-privilege access to selected files (when implemented).
- Secure RAG with source-grounded answers and storage-native citations (when implemented).
- Confidential-compute AI as an optional target deployment mode, not a current production claim unless deployed and attested.

19.2 Claims to avoid

- Unbreakable encryption.
- Impossible to hack.
- 100% anonymous.
- Cipher Nota knows nothing about users.
- Quantum proof.
- Zero-knowledge proofs unless an actual ZKP protocol is implemented.
- No administrator can ever participate in a cryptographic operation.
- AI can safely access the entire vault by default.
- Encrypted embeddings are fully searchable ciphertext unless a specialised cryptographic search scheme is actually deployed.

Recommended public security statement

Cipher Nota uses a zero-plaintext operational architecture with controlled cryptographic key access. Customer data is protected by layered encryption and policy boundaries, while AI access - when enabled - is limited to customer-selected content through independently revocable workspace grants.

20. Delivery Roadmap

Phase	Outcome
Phase 0 - Publication hardening	Confirm/document Vault ACL separation; implement directory HMAC; eliminate filename error-log leakage; document audit scope; record ML-KEM conformance evidence; publish region enforcement evidence.
Phase 1 - Storage crypto maturity	Per-file DEKs/envelope encryption; DEK re-wrap recovery; authenticated version manifests; provider token isolation; formal key lifecycle standard.
Phase 2 - Identity and device hardening	Passkeys/WebAuthn; hardware-backed device identities; richer revocation; privileged access management; release-key Vault/HSM controls.
Phase 3 - Secure RAG MVP	AI Workspaces; Ask My Files; semantic search; summaries; cross-document analysis; encrypted/isolation-protected vector store; source citations; read-only agents.
Phase 4 - Agent security	Agent Policy Gateway; capability taxonomy; human approval; workspace memory controls; AI audit events; model gateway; prompt-injection red-team programme.
Phase 5 - Confidential AI	Attested confidential-compute RAG; Vault key release based on workload attestation; enterprise model/data policy; approved external provider mode.
Phase 6 - Resilience and enterprise	Multi-provider replication; immutable snapshots; enterprise organisation accounts; BYO storage/model; compliance evidence; published assurance reports.

20.1 AI MVP recommendation

The first AI release should remain deliberately narrow: Ask My Files, semantic search, document summaries, cross-document comparison and a read-only Private Knowledge Agent. This creates meaningful customer value without granting autonomous agents destructive or external communication authority. Tool-enabled agents should follow only after the policy gateway, capability model and human-approval workflow are independently tested.

21. Final Architecture and Conclusion

Cipher Nota is best positioned not as another encrypted drive and not as another general-purpose AI chatbot. Its opportunity is to combine encrypted storage, dedicated cryptographic control and selectively authorised AI into one privacy-first platform.

Final architecture definition

Cipher Nota is a privacy-first encrypted multi-cloud storage and knowledge platform. It combines AES-256-GCM authenticated encryption, Argon2id password-derived protection, HashiCorp Vault-managed non-exportable root key material, ML-KEM-1024 post-quantum key establishment, recipient public-key sharing, LUKS/CryFS virtual-drive protection and customer-controlled recovery. Its proposed Secure AI layer lets customers grant revocable, least-privilege access to selected files for RAG and agent workflows without giving the AI unrestricted access to the customer account or Account Root Key.

The architecture deliberately exposes its trust assumptions. Vault is part of the trusted cryptographic computing base. User endpoints remain trusted while unlocked. Secure AI must process authorised plaintext somewhere, so the location and lifetime of that plaintext are explicit product choices: local endpoint, attested confidential runtime or customer-approved external provider. This transparency strengthens rather than weakens the security proposition.

The unifying design principle is separation of authority. Storage infrastructure stores ciphertext. Vault controls cryptographic authority. The customer controls recovery and AI grants. The AI policy plane controls what an agent may retrieve or do. No single ordinary storage component should independently provide sufficient authority to recover customer plaintext, and no AI model should independently obtain permission to access or modify customer data.

Final product proposition

Secure storage when you need durability. Secure knowledge when you need answers. Customer-controlled cryptographic authority across both.

Appendix A. Verified Production Controls

Control	Confirmed production position
Password KDF	Argon2id.
Account Root Keys	Maintained in HashiCorp Vault and configured non-exportable.
Plaintext ARK on application server	No persistent unencrypted server-side ARK.
Authenticated encryption	GCM; Vault aes256-gcm96 for AES-256-GCM operations.
Nonce handling	Vault manages/generates GCM nonce under the deployed cryptographic path.
Filename encryption	Available based on user choice.
Directory structure	Key/value mapping derived from hashed directory path.
Device keys	Generated using HashiCorp Vault.
Lost-device revocation	Requires the user recovery key.
Historical access after revocation	Possible for content/keys previously obtained by the device.
Recovery	New master-key context is generated and previous files are re-encrypted in the current process.
Independent provider recovery	Cipher Nota cannot complete recovery without customer recovery authority.
Sharing passwords	Processed through a Vault-backed secure API.
Recipient public-key sharing	Implemented.
Post-quantum KEM	ML-KEM deployed; reported ML-KEM-1024 dimensions 1568/3168/1568 bytes.
RSA-2048	Primarily signatures and secure server-channel authentication.
Virtual Drive	Per-user LUKS partition with files stored in the user Virtual Drive.
Physical storage	Virtual-drive partitions spread across VPS volumes.
Additional filesystem encryption	CryFS.
Routine metadata	User IP address.
Metadata retention	Up to 365 days for active users; deletion after three months of inactivity.
Metadata access	Privileged system administrators only.
Filename/object logging	Not intentionally logged, except potential error paths.
OAuth tokens	Protected through transport encryption, cryptographic binding and strict storage protocols.
Non-US storage	Technically enforceable.
Snapshot deletion	Linux OS deletion plus storage scrambling.
Cryptographic erase	Secure destruction/overwrite of the Media Encryption Key.

Control	Confirmed production position
Release signing	Cipher Nota-controlled release signatures verified with public keys.
Third-party scripts	Present, open source and audited.
Independent review	Completed by a contracted auditing team.

Appendix B. Target Security Decisions

- Generate ARKs inside Vault or use wrapped import; never route plaintext root keys through general application provisioning.
- Use per-file DEKs and envelope encryption to reduce blast radius and enable efficient recovery/re-wrapping.
- Replace deterministic directory-path hashing with tenant-keyed HMAC identifiers.
- Make filename encryption the privacy-default setting.
- Prefer endpoint-generated hardware-backed device private keys where supported.
- Use DEK re-wrapping for ordinary recovery and full re-encryption only for compromise response.
- Protect release signing keys with Vault/HSM controls and dual approval.
- Implement structured error-log redaction.
- Use region allow-lists and policy-as-code as evidence for residency claims.
- Make read-only AI the default and require separate capabilities for side effects.
- Never give an AI agent the Account Root Key.
- Use short-lived AI workspace capabilities and independently revocable workspace keys.
- Apply authorization before vector similarity search and again before context construction.
- Treat retrieved content as hostile input for prompt-injection purposes.
- Use an Agent Policy Gateway for deterministic tool authorization.
- Provide source citations and object/version provenance for RAG answers.
- Use confidential computing as a premium target where server-side plaintext AI processing is required.
- Do not use customer content for model training unless the customer explicitly opts in under a separate policy.

Appendix C. AI Security Control Matrix

Control area	Required control
AI workspace creation	Strong user auth; explicit selected objects; model/privacy choice; TTL
Key access	Workspace-specific authority; no ARK exposure to model/agent
Ingestion	Parser isolation; malware checks; provenance; classification
Chunk storage	Encrypted at rest; tenant/workspace isolation; deletion on revoke
Embeddings	Sensitive data; isolated vector store; ACL before similarity search
Retrieval	Tenant/workspace/object filtering; re-validation before context
Prompt injection	Retrieved data non-authoritative; injection tests; policy isolation
Model invocation	Approved model gateway; minimal context; classification policy
Tool use	Capability allow-list; Agent Policy Gateway; human approval for high risk
Memory	Session default; optional workspace/permanent; encrypted, visible, deletable
External provider	Explicit customer/policy approval; no-training/no-retention terms where applicable
Confidential compute	Attestation-gated key release; hardened image; debug disabled

Control area	Required control
Audit	Metadata event trail; no routine plaintext prompt/chunk logging
Revocation	Invalidate tokens; kill sessions; destroy workspace key/index/cache
Assurance	Red team; RAG isolation tests; provider review; SSDF-A/AI RMF alignment

Appendix D. Reference Standards and Guidance

The following external references inform the recommended controls. They do not imply certification or conformance unless Cipher Nota separately demonstrates and documents that status.

R1. HashiCorp. HashiCorp Vault Transit Secrets Engine - API. 2026. https://developer.hashicorp.com/vault/api-docs/secret/transit *Defines non-exportable Transit keys and aes256-gcm96.*

R2. HashiCorp. Vault Transit - Envelope Encryption. 2026. https://developer.hashicorp.com/vault/docs/secrets/transit/envelope-encryption *Reference pattern for DEKs/EDKs and client-side envelope encryption.*

R3. IETF/IRTF RFC Editor. RFC 9106: Argon2 Memory-Hard Function for Password Hashing and Proof-of-Work Applications. 2021. https://www.rfc-editor.org/info/rfc9106/ *Argon2id definition and test vectors.*

R4. NIST. FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard. 2024. https://csrc.nist.gov/pubs/fips/203/final *Defines ML-KEM-512, ML-KEM-768 and ML-KEM-1024.*

R5. NIST. SP 800-207: Zero Trust Architecture. 2020. https://csrc.nist.gov/pubs/sp/800/207/final *Zero-trust principles and reference architecture.*

R6. NIST. AI 600-1: Artificial Intelligence Risk Management Framework - Generative AI Profile. 2024; updated 2026. https://www.nist.gov/publications/artificial-intelligence-risk-management-framework-generative-artificial-intelligence *Generative AI risk-management profile.*

R7. NIST. IR 8320E: Hardware-Enabled Security - Confidential Computing of Data in Cloud Workloads (Initial Public Draft). 2026. https://csrc.nist.gov/pubs/ir/8320/e/ipd *Confidential computing and AI cloud-workload example architecture; draft status.*

R8. NIST. SP 800-88 Rev. 2: Guidelines for Media Sanitization. 2025. https://csrc.nist.gov/pubs/sp/800/88/r2/final *Media sanitisation and cryptographic erase guidance.*

R9. W3C. Web Authentication: An API for accessing Public Key Credentials - Level 3. 2026 Candidate Recommendation. https://www.w3.org/TR/webauthn-3/ *Public-key credential API for strong authentication.*

R10. NIST. SP 800-57 Part 1 Rev. 5: Recommendation for Key Management - General. 2020. https://csrc.nist.gov/pubs/sp/800/57/pt1/r5/final *General key-management guidance.*

R11. NIST. SP 800-218: Secure Software Development Framework (SSDF) Version 1.1. 2022. https://csrc.nist.gov/pubs/sp/800/218/final *Secure software development practices.*

R12. NIST. SP 800-218A: Secure Software Development Practices for Generative AI and Dual-Use Foundation Models. 2024. https://csrc.nist.gov/pubs/sp/800/218/a/final *Generative AI secure-development community profile.*

R13. OWASP GenAI Security Project. LLM01:2025 Prompt Injection. 2025. https://genai.owasp.org/llmrisk/llm01-prompt-injection/ *Prompt and indirect prompt injection risk.*

R14. OWASP GenAI Security Project. OWASP Top 10 for LLM Applications 2025 - Vector and Embedding Weaknesses. 2025. https://genai.owasp.org/resource/owasp-top-10-for-llm-applications-2025/ *RAG/vector/embedding risk guidance.*

R15. OWASP GenAI Security Project. LLM06:2025 Excessive Agency. 2025. https://genai.owasp.org/llmrisk/llm062025-excessive-agency/ *Agent/tool authority risk and mitigations.*

Source Material

S1. Cipher Nota White Paper Brief (WHITEPAPER_BRIEF.md), supplied by the product owner. The brief describes the existing product positioning, storage features, claimed security model, privacy posture, provider support, plans, evidence map and unresolved publication gaps.

S2. Product-owner architecture confirmations supplied during the white-paper review on 15 August 2026, covering Argon2id, Vault non-exportable root keys, AES-GCM, nonce handling, device/recovery behaviour, ML-KEM, LUKS/CryFS, metadata, residency, deletion, release signing and independent review.

End of document

Version 1.1 - Final Technical White Paper - 16 August 2026