Core0

A self-hosted vault for personal data, built so that the vendor never receives it.

Technical whitepaper · August 2026 · Obsydia Technologies · obsydia.tech

> Every managed PII vault asks you to send it the data you are trying to protect. Core0 runs inside your network and makes no outbound call at runtime — no telemetry, no licence check, no vendor endpoint anywhere in the request path.

WHY THIS EXISTS

One engineer, two constraints

Core0 is built by one engineer. That is a limitation, and it is also the constraint that shaped the product.

Every self-hosted vault I evaluated assumed a platform team on the other side — someone to own the deployment, keep it patched and understand its failure modes. Most organisations that need a vault do not have that person. They have one senior developer who already has a full job. So the first requirement was a vault a competent engineer can install, operate and reason about without becoming a specialist in it. The complexity is not eliminated; it is not exposed.

The second requirement came from reading the contracts. Managed vaults solve key management by holding your keys, and solve data protection by holding your data. For an organisation whose whole reason for buying is that it cannot afford a third party to hold either, that is a strange trade. It is the one thing Core0 does not ask of you.

Who this is for

Organisations in the UK and EU that hold regulated personal data and have to demonstrate, not merely assert, that no third party can reach it: fintech, healthtech, legaltech, and any business whose auditors ask pointed questions about processor access.

What the auditor actually asks

The question is rarely "is it encrypted". It is whether any third party, including your cloud provider, has the technical ability to access personal data. With a managed vault the honest answer is "contractually, no". With Core0 the answer is that no path exists, and that difference is what can be put in front of an auditor.

AT A GLANCE

What it does	Stores personal data and documents outside your application database. Your app holds tokens; the data lives behind separate keys, access control and an audit trail.
Where it runs	Your infrastructure. Bare metal or private cloud. No component runs on ours.
What leaves your network	Nothing at runtime. Verifiable with a packet capture in under a minute.
Deployment	Measured at roughly 20 minutes for a full cluster including the key ceremony, on prepared machines. We quote under two hours to leave room for reality.
Who deploys it	One person who can tell an IP address from an SSH password. No DevOps specialism required.
Editions	Starter (3 nodes), Pro (6), Enterprise (10). Same binaries, same architecture, same installation procedure.
Compliance	Built around GDPR Art. 17, 20, 25 and 30. Suitable for ISO 27001 environments.

Encryption at rest answers the wrong question

Disk encryption protects you from someone stealing the server. Nobody steals the server. Data leaves through SQL injection, an over-permissive API, a leaked backup, or a support tool with more reach than anyone remembered granting it — and in every one of those cases the database was mounted, decrypted and answering queries as designed.

Tokenisation changes what the attacker ends up holding. The application database contains references. The personal data lives elsewhere, behind its own keys and its own access control. A full dump of the application database is then an inventory of meaningless strings.

Where the alternatives sit

Managed vaults

AWS Secrets Manager, Azure Key Vault, Skyflow, Basis Theory and similar services solve key management competently. They also relocate the sensitive material into infrastructure you do not control, which for a UK or EU organisation raises questions about residency, international transfer and processor access that no contract fully closes. Several also charge per operation, so protecting a customer costs money every time you do it — an incentive pointing the wrong way.

Self-hosted enterprise software

HashiCorp Vault is a serious tool and needs serious operators. For a twenty-person startup or a two-hundred-person regulated business, the operational overhead is the reason the project never ships. Misconfiguration in a complex system produces exactly the exposure the system was bought to prevent.

The erasure gap

Article 17 is not a DELETE statement. It requires that the data is gone from every layer that holds it, including write-ahead logs, replicas and backups. Most systems cannot demonstrate this because deletion was never modelled as a first-class operation. Core0 treats it as one.

Three zones, not one system

The design assumes your application will eventually be breached, and is built so that this is survivable rather than fatal.

Zone	Holds	What compromise yields
Application	Tokens	References with no meaning outside the vault. This is the zone attackers actually reach, and deliberately the one with nothing worth taking.
PII layer	Encrypted records and files	Ciphertext, plus the ability to answer requests it is authorised to answer. It cannot read key material.
KEYS layer	Key material, licence enforcement	The serious case — which is why it is the smallest, quietest and least connected part of the system, and holds no personal data itself.

The separation is the point: no single compromised component reconstructs a person's record.

Trust boundaries

Every call between zones is mutually authenticated, with short-lived certificates that rotate on a six-hour cycle and a fourteen-day proactive threshold. No internal channel runs unauthenticated or unencrypted, including between machines in the same rack. Traffic between nodes runs over an encrypted WireGuard mesh regardless of what your own network already provides, on the assumption that your network is not trustworthy either. Consul handles discovery and health checking.

Cryptography

`Record and file encryption`	AES-256-GCM, per record
`Signing and licence tokens`	ECDSA P-256
`Service authentication`	Mutual TLS, automatic rotation
`Key-encryption key`	Split into three Shamir shares at install time and handed to you to store offline; two reconstruct it, and no automatic path does so
`Storage`	BadgerDB

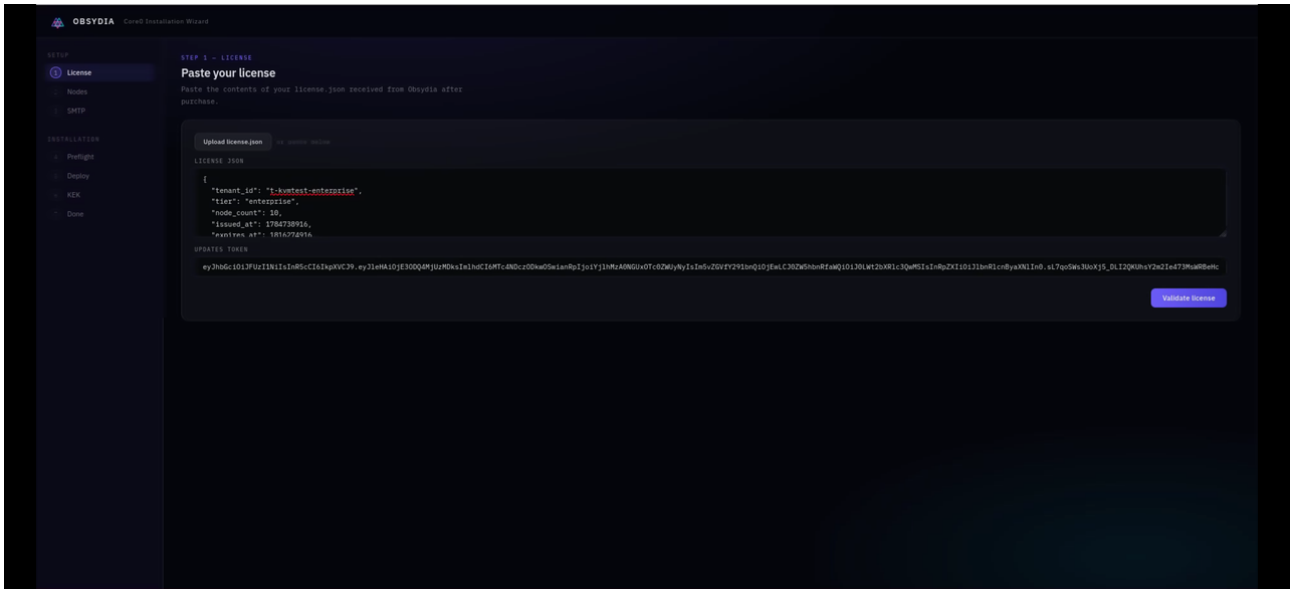
There is no vendor-held escrow and no recovery service. Losing enough shares means losing the data. We would rather state that plainly than imply a safety net that would itself be a vulnerability.

Key custody, stated precisely

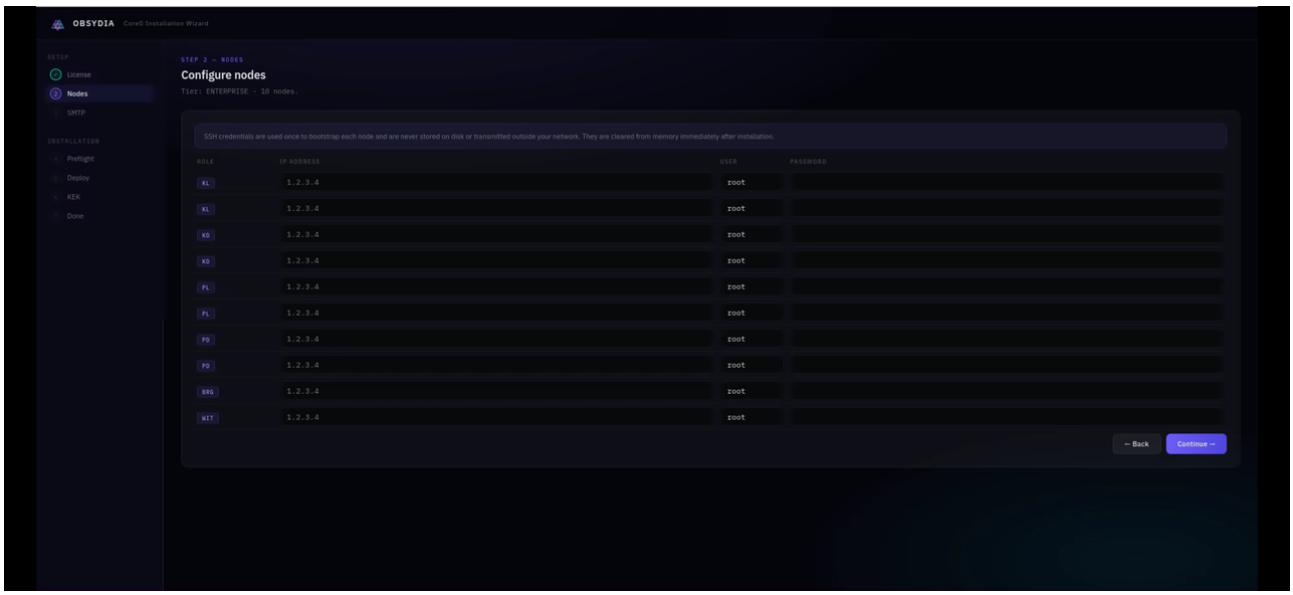
The ceremony is automated: the installer generates the shares and displays them once, for you to write to separate offline media. Nothing is transmitted and nothing is retained. Two points follow from this and are worth being explicit about. The operator running the installation sees all three shares at that moment, so the two-of-three threshold protects you against later loss of a share rather than against that operator — we recommend two people present for exactly this reason, though the software does not enforce it. And once the wizard closes, the shares exist only where you put them.

What installation actually looks like

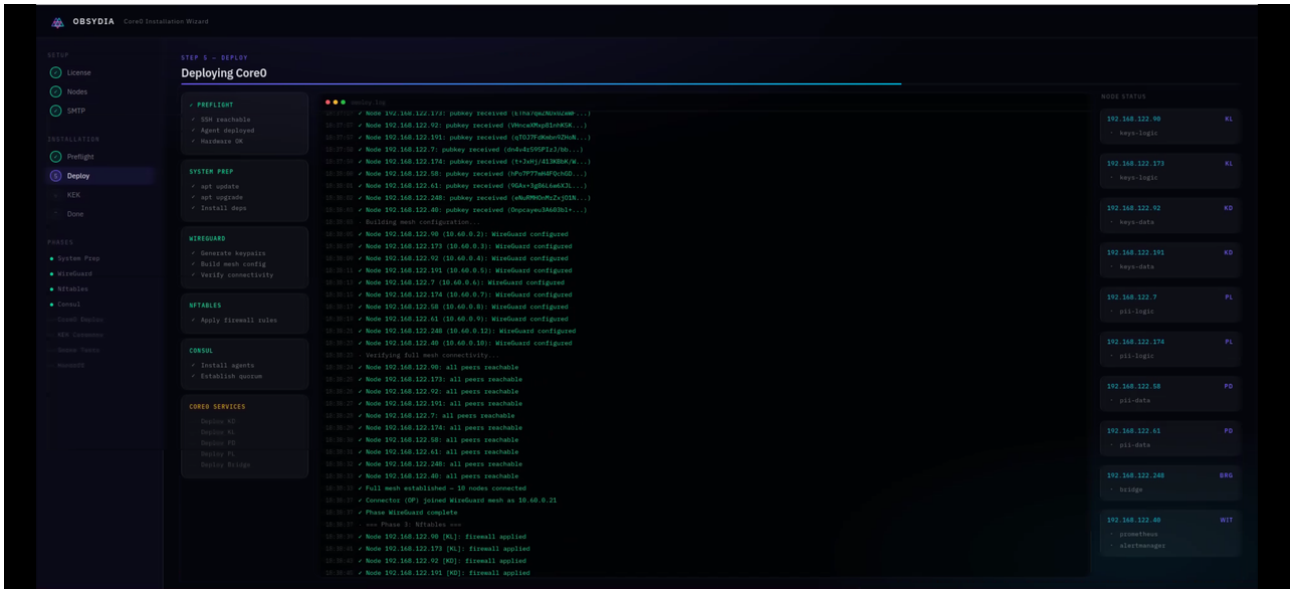
The wizard takes a licence file, a list of machines and SSH credentials, then builds the mesh, applies firewall rules, establishes quorum, deploys the services and runs the key ceremony. SSH credentials are used once per machine, are never written to disk, and are cleared from memory when installation finishes.



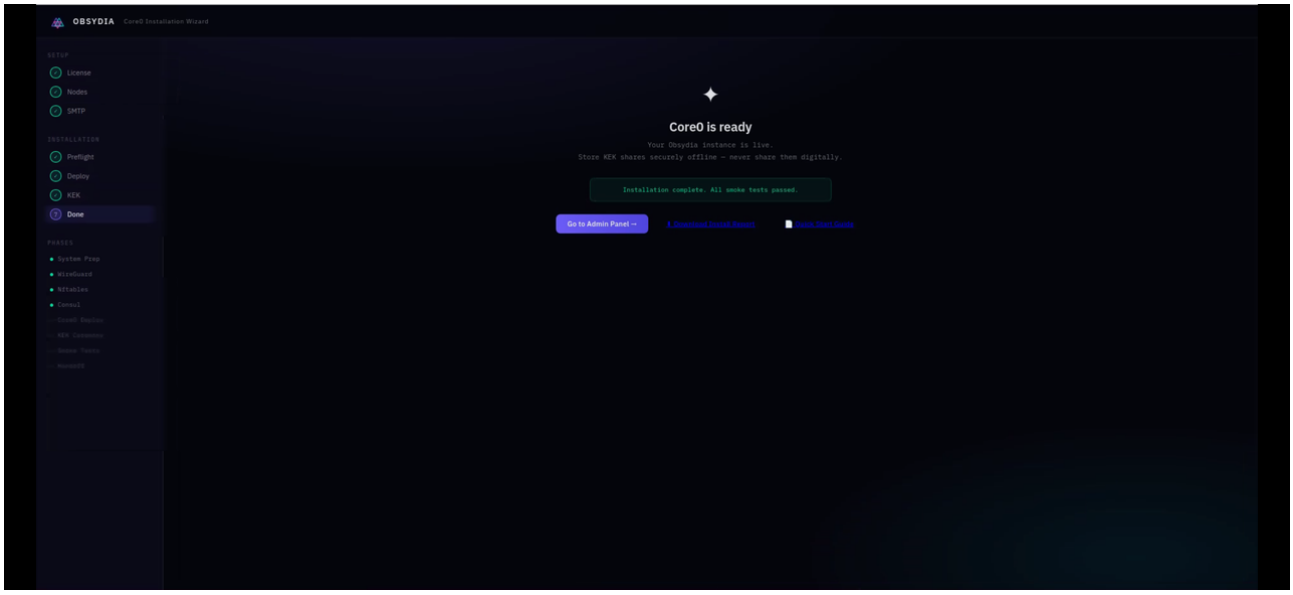
Step 1 — the licence file arrives by email after purchase and is loaded locally. Nothing is validated against a remote server, then or later.



Step 2 — machines and roles. The node count comes from the licence; the installation procedure is identical across editions.



Step 5 — ten nodes joining the WireGuard mesh, firewall rules applied, quorum established. Timestamps in this log are real.



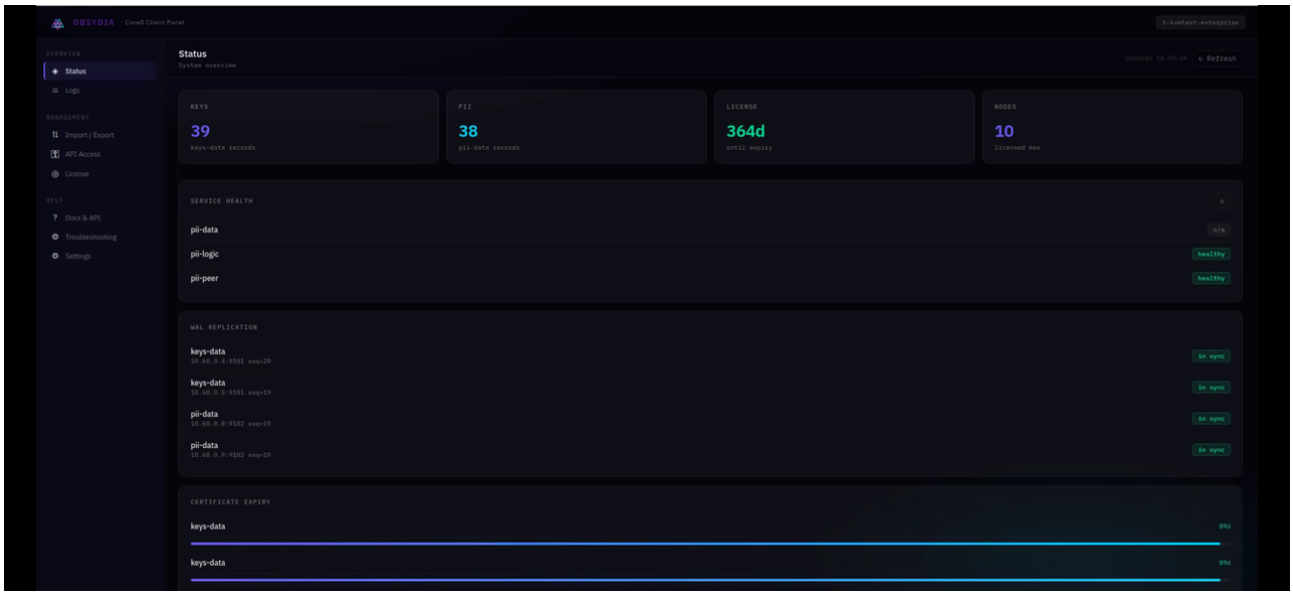
Done. Key shares are shown once for offline storage and are never transmitted or retained.

Timing, honestly

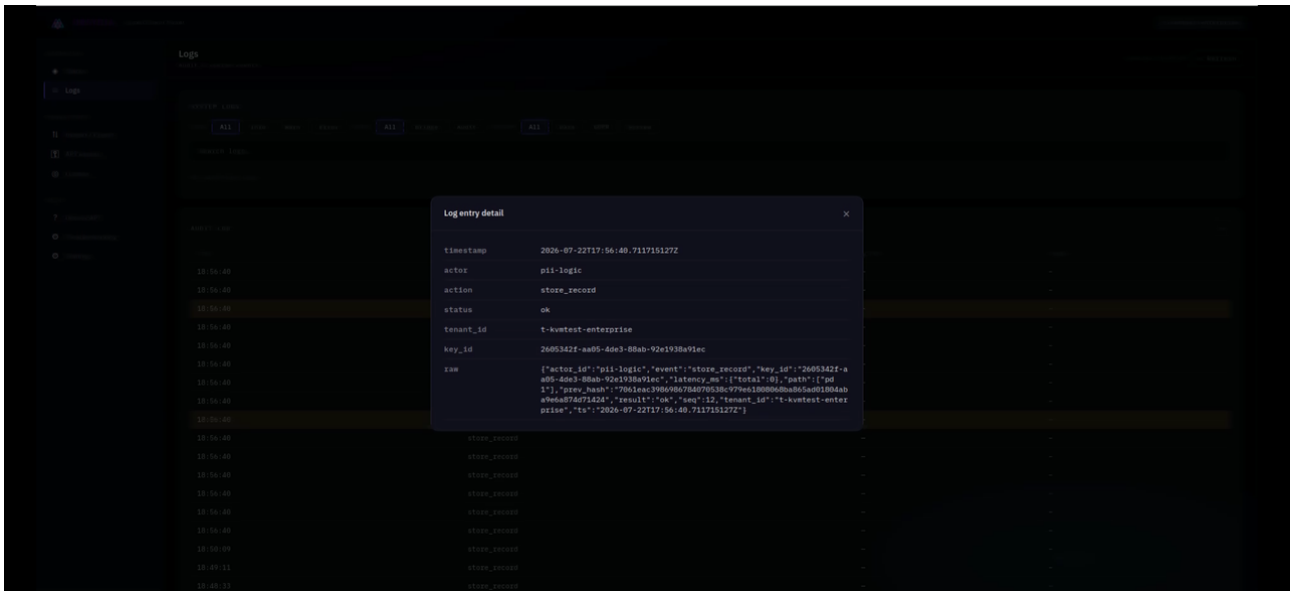
> A measured run on prepared machines completes in roughly twenty minutes, key ceremony included. We quote under two hours, because prepared machines are an assumption and yours may disagree. Every edition installs the same way — more nodes means more parallel work for the installer, not a longer or harder procedure.

The person running it needs to know what an IP address and an SSH password are. That is the intended bar, and it is the reason the wizard exists.

What you see afterwards



Service health, replication state per node pair and certificate expiry, in one view. Replication applies to Enterprise; see the limitations below.



Every API call records actor, action, tenant, key and latency. Entries are chained by sequence and previous-hash so that tampering is detectable.

The audit trail doubles as an Article 30 register: each processing activity carries purpose, actor and tenant context without anyone maintaining a spreadsheet alongside it.

GDPR, article by article

Article	How it is handled
Art. 17 — erasure	Deletes the record, purges the associated write-ahead log entries and returns a signed receipt with timestamp and operator identity.

Article	How it is handled
Art. 20 — portability	Exports every record for a tenant as machine-readable CSV: token, value, purpose, timestamp.
Art. 25 — by design	Personal data is tokenised before it reaches your application database. Your application never holds raw PII — by architecture, not by policy.
Art. 30 — records	The audit log is the register, maintained automatically.
Art. 32 — security	Encryption in transit and at rest, role-separated key custody, tested restore path.

LIMITATIONS

What Core0 does not do

Every vendor document ends with a summary of strengths. This one lists what the design does not cover, because the questions below will be asked either way and the answers are better coming from us.

Root on an unsealed node

An attacker with root on a running, unsealed node can read what that node can read. No vault design changes this, and any vendor claiming otherwise is selling something else.

Authorised misuse

An operator acting within their permissions is recorded, not prevented. An audit trail is a different guarantee from access control, and a weaker one.

High availability below Enterprise

Synchronous replication between node pairs, quorum-backed discovery and VIP failover are Enterprise features. Starter and Pro run single processes per layer with no replication: a node failure means downtime until it is restarted, and recovery comes from the snapshot schedule. This is a deliberate trade — most deployments do not need five nines and should not pay for them.

Audit chain across restarts

The audit log chains entries by sequence number and previous hash, which makes tampering within a run detectable. The chain currently restarts when the service does, so continuity across restarts is not yet guaranteed. This is a known defect with a fix pending, and it is listed here rather than left for you to discover.

Independent assessment

Core0 has not yet been through a third-party penetration test. The architecture and threat model are published so that they can be argued with in the meantime, and we would rather you ran a packet capture against a live node than took any of this on trust.

EDITIONS

Edition	Nodes	Replication
Starter	3	no
Pro	6	no
Enterprise	10	yes

Core0 is licensed annually, not metered. There is no per-operation charge, so protecting more customers never costs more. Current pricing is at obsydia.tech/pricing — deliberately not reproduced here, so that a copy of this document circulating in six months cannot quote a stale figure at you.

Questions, or a technical review before you commit to anything: hello@obsydia.tech · obsydia.tech/core0/architecture

Obsydia Technologies · August 2026. Figures and screenshots in this document are from a running deployment, not a mock-up.