



SENTRA  
DOCUMENTATION

PRODUCT SPECS

# SENTRA Security Architecture

How SENTRA protects its keys, its licences, its updates and your data

The trust model behind the SENTRA platform: an offline root key, an encrypted licensing vault, signed licences and signing authorities, signed software releases, two-factor sign-in everywhere, least-privilege roles, tamper-evident audit ledgers, what data leaves a box, fail-open design and service hardening.

---

Version ATS 2.26 · NTS 0.18 · GUARD & VIEW 1.25 · Licensing 1.2

27 October 2026

---

# Contents

- 01 Summary
  - 02 Trust model and keys
  - 03 The licensing chain
  - 04 Signed software releases
  - 05 Identity and access
  - 06 Audit ledgers
  - 07 Data handling
  - 08 Resilience and fail-open design
  - 09 Service hardening and network exposure
  - 10 What customers should do
-

# Summary

The SENTRA security architecture on one page.

SENTRA is a family of security products — **SENTRA AI Threat Shield (ATS)** for websites and APIs, **SENTRA Network Threat Shield (NTS)** for internet lines, **SENTRA GUARD** to run them for many customers and **SENTRA VIEW** to command a fleet of GUARDS — backed by two SENTRA services: the **Licensing Authority** (`licensing.sentra-ai.com`) and the **release server** (`updates.sentra-ai.com`).

The platform is built on five principles:

| Principle                                | In practice                                                                                                                                                                                                                       |
|------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>One root of trust, kept offline</b>   | A single Ed25519 root key. Its private half lives only in an encrypted, sealed vault (plus one offline recovery copy). Every product carries only the public half.                                                                |
| <b>Everything that matters is signed</b> | Licences, signing-authority certificates, release keys, release lists, packages and every file in them chain back to the root key. Products verify signatures themselves; they never trust a server just because they reached it. |
| <b>Least privilege and two people</b>    | Roles and per-user limits everywhere; two-factor sign-in; a fresh 2FA code for sensitive actions; four-eyes approval for licences and releases.                                                                                   |
| <b>Tamper-evident records</b>            | Administrative actions are written to hash-chained audit ledgers that reveal any edit, deletion or reordering.                                                                                                                    |
| <b>Protection never stops</b>            | Licensing and update problems never switch protection off; inspection components fail open rather than cutting a customer's traffic.                                                                                              |

## The chapters

1. [Trust model and keys](#)
2. [The licensing chain](#)
3. [Signed software releases](#)
4. [Identity and access](#)
5. [Audit ledgers](#)
6. [Data handling](#)
7. [Resilience and fail-open design](#)
8. [Service hardening and network exposure](#)
9. [What customers should do](#)

### NOTE

This document describes mechanisms, not operational secrets. It applies to ATS 2.26, NTS 0.18, GUARD & VIEW 1.25 and Licensing Authority 1.2.

# Trust model and keys

The SENTRA root key, the encrypted licensing vault, and the keys derived from it.

## The root key

SENTRA has one **root key** (Ed25519). Its public half is built into every ATS, NTS, GUARD and VIEW. Its private half:

- is stored only inside the Licensing Authority's **encrypted vault**, from which it is never shown or exported again;
- is kept, in addition, as **one offline recovery copy** under SENTRA's physical control;
- is never placed on any customer system, build server or code repository.

The root key signs: stand-alone and GUARD licences, signing-authority certificates for VIEWs and GUARDs, and the certificate of the release key. A root-key rotation is prepared in advance: products are first updated to trust the new public key alongside the old one; only then does the new key start signing.

## The vault

| Layer      | Mechanism                                                                                                                                                           |
|------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Master key | 256 random bits, created once, held <b>only in memory</b> while the vault is unsealed                                                                               |
| Wrapping   | AES-256-GCM under a key derived from the vault passphrase with <b>scrypt</b> ( $N = 2^{17}$ , $r = 8$ , $p = 1$ : about 128 MiB of memory per guess)                |
| Records    | Every record (users, customers, licences, keys, authorities, settings) encrypted with <b>AES-256-GCM</b> ; the record's type and ID are bound in as associated data |
| Sub-keys   | Derived with <b>HKDF-SHA256</b> : separate keys for records, the blind index, the audit trail and its MACs                                                          |
| Look-ups   | An <b>HMAC-SHA256 blind index</b> : names and IDs are never stored readably, not even a customer's name                                                             |
| Sealing    | After <b>any</b> restart the vault is sealed; nothing can be read or signed until an authorised person enters the passphrase                                        |

Backups are copies of the encrypted files: useless without the passphrase. The passphrase is not stored anywhere on the server and cannot be recovered.

## Derived keys

| Key           | Held by                                                       | Certified by                                                    | Signs                                                                    |
|---------------|---------------------------------------------------------------|-----------------------------------------------------------------|--------------------------------------------------------------------------|
| Root key      | The vault (+ offline copy)                                    | —                                                               | 0TG1 licences, 0TGA1 authority certificates, the release-key certificate |
| Authority key | A customer's VIEW or GUARD (generated there, never leaves it) | The root key, with limits                                       | 0TG2 licences for that customer's fleet                                  |
| Release key   | The vault                                                     | The root key (5-year certificate, limited to SENTRA's products) | Each product's release list                                              |

A compromise of one VIEW's authority key lets an attacker issue keys only within that authority's limits (products, modules, counts, latest expiry), and only until the certificate ends. A compromise of the release key can be answered by rotating it: products trust the root, not the release key, so a new certified release key replaces it at once.

# The licensing chain

Root-signed licences, signing authorities with limits, how products verify a licence, online check-in and revocation.

## Licence formats

| Format | Signed by          | Carries                                                                                          |
|--------|--------------------|--------------------------------------------------------------------------------------------------|
| 0TG1   | Root key           | Product, Installation ID (or any), customer, modules, start, expiry; seats and tier for a GUARD  |
| 0TGA1  | Root key           | An authority's own public key and its limits: products, modules, counts, latest expiry, validity |
| 0TG2   | An authority's key | The same as 0TG1, plus the authority's certificate and ID                                        |

Every part is compact JSON with sorted keys; signatures are Ed25519 over the exact bytes. A key can be pasted anywhere and verified offline.

## Signing authorities

Customers with a VIEW fleet, or a stand-alone GUARD, receive a **signing authority**: their own VIEW or GUARD generates a key pair and the Licensing Authority certifies the public key with limits. The authority then licenses the systems in its fleet by itself, even when disconnected, and reports every key it issues back to the Licensing Authority — online, signed with its own key, or as a signed ledger file.

## What a product verifies

For an authority-issued (0TG2) licence a product checks, offline:

1. the authority certificate is signed by a root key the product trusts;
2. the licence is signed by the certified authority key, and names that authority;
3. the authority may license this product and these modules;
4. the licence starts within the certificate's validity and doesn't run past the authority's latest expiry.

A key that fails any check is rejected with the reason in plain words.

## Online check-in and revocation

ATS and NTS with internet access check in with the Licensing Authority about two minutes after start and every six hours. They send only: the product, the Installation ID, the host name, the software version and the **IDs** of the licences they hold (not the keys). The answer gives each licence's status and any renewed keys for that installation.

A **revoked** licence is marked revoked on the box at its next check-in and stays revoked across restarts. Check-in can be switched off on a box that must never contact the internet; that box then relies on its keys' expiry dates.

### NOTE

Licensing never stops protection. A module without a valid licence has a 30-day grace period and then becomes *restricted*: protection keeps running as configured; only configuration changes for that module are blocked.

# Signed software releases

How SENTRA builds, approves, signs and distributes updates, and what every system checks before installing one.

## Building and approving

- An isolated **builder** service, running as its own unprivileged account, checks out each product from a read-only source, runs the product's tests and prepares a candidate. **It cannot sign or publish.**
- An authorised SENTRA approver reviews the candidate's test results and release notes and approves it with a **fresh two-factor code**. With four-eyes on, a second approver must approve too.
- Only then does the Licensing Authority's vault sign the product's release list with the **release key**, which the root key certifies.
- A published version can never be replaced; a bad release can be **withdrawn**, which re-signs the list so that no system offers or installs it again.

## What every system checks

```
root key (built in)
├─ release-key certificate (signed by the root; valid dates; allowed products)
│   └─ release list (index.json, signed by the release key, checked over its exact bytes)
│       └─ the package's SHA-256 and the file list's SHA-256
│           └─ every file's SHA-256
```

Nothing is installed unless every link checks out. SENTRA VIEW passes the signed proof down to each GUARD, which checks every file again, and each GUARD to its ATS and NTS boxes, which check again: no layer has to trust the one above it.

## The release server holds no secrets

updates.sentra-ai.com only serves signed files. Someone who controlled it, or the network path to it, could at most delay updates; they could not make any SENTRA system install code SENTRA did not approve. A second release server can therefore be added for availability without adding any trust.

## Customer control

Customer administrators decide when an offered update is installed (now, scheduled, or skipped), unless they choose automatic updates. Fleet operators can lock GUARDS and nodes to **signed releases only**.

# Identity and access

Two-factor sign-in, step-up confirmation, roles and least privilege, sessions and lockouts across the SENTRA consoles.

## Two-factor sign-in

| Console                   | Two-factor (TOTP)                                                                                    |
|---------------------------|------------------------------------------------------------------------------------------------------|
| Licensing Authority       | <b>Always required:</b> set up at the first sign-in, can't be skipped; eight one-time recovery codes |
| SENTRA VIEW, SENTRA GUARD | Available to every user; the security policy can <b>require</b> it for all staff                     |
| ATS and NTS               | Available to every user; a GUARD's policy can require it for the users it manages on its boxes       |

Codes follow RFC 6238 (30-second steps) and each code is accepted **once**.

## Step-up confirmation

In the Licensing Authority, revealing keys, downloading key bundles, approving licences and releases, managing keys, users and policy need a **fresh** two-factor code, valid for a few minutes, even inside a signed-in session.

## Roles and least privilege

- Every console has roles with explicit permissions; administrators can build their own roles in ATS, NTS, GUARD and VIEW. Built-in top-level administrator roles can't be removed or weakened, and the last top-level administrator can't be removed.
- A user who manages other users can't make anyone a top-level administrator or change a top-level administrator's account.
- In the Licensing Authority, per-user **limits** restrict which kinds of licence each person may issue, for how long, whether perpetual, and for how many systems; **four-eyes** approval separates requesting from approving.
- On boxes managed by a GUARD, the GUARD distributes its users (password hashes and 2FA secrets, only for the boxes each user may reach) and can switch local accounts off; one emergency **glass-break** account works only while the GUARD can't be reached.

---

## Sessions and sign-in protection

- Passwords are stored only as salted, deliberately slow hashes: scrypt in the Licensing Authority, PBKDF2-SHA256 with 200,000 iterations in ATS and NTS, PBKDF2 in GUARD and VIEW. Minimum lengths and complexity rules apply.
- Sign-in attempts are limited with lockouts and rate limits; in the Licensing Authority unknown users and wrong passwords get identical answers.
- Console session cookies in the Licensing Authority, ATS and NTS are `HttpOnly` and `SameSite=Strict`, and Secure over HTTPS. In the Licensing Authority every change also needs a per-session CSRF token, cross-site requests are refused, and idle and absolute session timeouts apply.
- The Licensing Authority keeps sessions only in memory and can restrict its console to listed networks.

# Audit ledgers

How SENTRA records administrative actions so that any later change to the record is visible.

| Where                        | What is recorded                                                                                                                                                                                                                   | Protection                                                                                                                                                                         |
|------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Licensing Authority          | Every sign-in (and refused attempt), key reveal, bundle download, approval, issue, revocation, authority enrolment and sync, import, key and user change, policy change, seal/unseal, release approval, publication and withdrawal | Each entry AES-256-GCM encrypted and hash-chained to the previous one (SHA-256), each chain value sealed with an HMAC under a vault key; one-click verification of the whole chain |
| SENTRA AI Threat Shield      | Security decisions and administrative changes                                                                                                                                                                                      | A tamper-evident ledger that can be verified and exported                                                                                                                          |
| SENTRA Network Threat Shield | Every administrative change from the console and the command line                                                                                                                                                                  | One JSON line per entry holding the SHA-256 of the previous entry; <code>sentra-nts audit verify</code> checks it; secrets are never written                                       |
| SENTRA VIEW                  | Staff actions across the fleet                                                                                                                                                                                                     | Entries chained by hash                                                                                                                                                            |

In a hash chain each entry includes a fingerprint of the one before it. Editing, deleting or reordering any past entry changes every fingerprint after it, so verification shows exactly where the record was altered.

---

## What is never written to a ledger

The Licensing Authority's trail names *what* was done (for example "revealed licence OTG-..."), *by whom*, *from where* and *when*; it never holds passwords, 2FA codes, recovery codes, licence keys or private keys. The NTS ledger likewise never writes passwords, tokens, 2FA codes or mail passwords.

# Data handling

What leaves a SENTRA box, where it goes, and what stays on the customer's systems.

SENTRA products run on the customer's own servers or appliances. Traffic is inspected there; it is not sent to SENTRA.

## What a box sends to SENTRA

| To                                                           | When                                                                   | What                                                                                                                          | Can be switched off                                                             |
|--------------------------------------------------------------|------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------|
| licensing.sentra-ai.com (licence check-in, ATS and NTS)      | ~2 minutes after start, then every 6 hours                             | Product, Installation ID, host name, software version, the IDs of the licences it holds                                       | Yes: SENTRA_LICENSING_CHECK=0 or "licensing_check": false                       |
| updates.sentra-ai.com (stand-alone ATS and NTS, SENTRA VIEW) | Every 30 minutes (VIEW: every 15 minutes when automatic, or on demand) | Downloads only: the signed release list and, when an update is installed, the package                                         | Updates are installed only when an administrator decides (or chooses automatic) |
| licensing.sentra-ai.com (VIEW or GUARD signing authority)    | Enrolment, and syncs                                                   | The authority's public key at enrolment; signed records of the licences it issued (licence keys, target system names, status) | An offline authority uses a signed ledger file instead                          |

No traffic content, request logs, visitor addresses, customer user accounts, passwords or configuration are sent to SENTRA by these mechanisms.

## Inside a managed fleet

Within a customer's or operator's own fleet, data flows only between their own systems:

- **ATS / NTS box → its GUARD:** health, statistics, licence state, security events for the GUARD's dashboards; the GUARD sends configuration, users allowed on that box, licences and updates.
- **GUARD → its VIEW:** a summary of customers, nodes, firewalls, billing records and alarms; VIEW sends changes and releases.

---

## Optional outside services

Some features contact outside services **only when an administrator switches them on**, for example public threat feeds or an AI provider for the AI analyst. The ATS console states this on every page: it only contacts outside services you switch on.

---

## Data at rest

- The Licensing Authority encrypts every record and every audit entry (see [Trust model and keys](#)).
- Secrets on boxes (keys, tokens, authority keys) are stored in files readable only by the service that needs them.

# Resilience and fail-open design

Why licensing, updates and component failures never cut a customer's traffic.

A security product that stops traffic when it has a problem becomes the outage. SENTRA is designed so that its own failures degrade inspection, not connectivity.

---

## SENTRA AI Threat Shield

- A separate **fail-open guard** service watches the shield. If the shield stops answering, the guard sends website traffic **straight to the web server** until the shield is healthy again.
- An emergency **bypass** switch (`sentra-shield-setup bypass on`) does the same on purpose, within about 5 seconds, and is undone with `bypass off`.
- Updates restart the shield for about 10 seconds while websites keep working.

---

## SENTRA Network Threat Shield

- Packet inspection (the IPS, DNS and HTTPS-name filtering) uses kernel queues bound **fail-open**, with rules that bypass the queue when no inspector listens: if an inspector stops or falls behind, packets pass rather than drop.
- Two boxes can run as an **active / standby pair**: the standby takes over within about 3 seconds, with the connection table copied across so open connections survive.

---

## Licensing and updates

- Licences are verified offline; a box never needs the Licensing Authority to keep working.
- If a licence lapses, protection keeps running exactly as configured (30-day grace, then only configuration changes for that module are blocked).
- If the Licensing Authority is sealed or unreachable, check-ins fail harmlessly and are retried.
- An update that fails any check is refused and nothing changes; an update that fails to install is recorded and the previous version keeps running. GUARD and VIEW can **roll back** to the previous release.

# Service hardening and network exposure

How SENTRA's own services run with the least privilege and the smallest network exposure.

## Least-privilege services

| Service                 | Hardening                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|-------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Licensing Authority     | Own system user without a login shell; code owned by root and read-only to the service; data folder readable only by the service; listens on 127.0.0.1 only; systemd sandbox: no capabilities, NoNewPrivileges, read-only system (ProtectSystem=strict), private /tmp and devices, kernel and control-group protection, restricted address families and namespaces, a system-call filter that forbids privileged calls, private umask, no core dumps |
| Release builder         | Its own unprivileged user; can write only its work folder; can't read the vault; runs product tests with a temporary home folder and a minimal environment; never signs                                                                                                                                                                                                                                                                              |
| Release server          | Static files only, on 127.0.0.1, behind SENTRA AI Threat Shield; only the release paths answer, everything else is 404                                                                                                                                                                                                                                                                                                                               |
| SENTRA AI Threat Shield | Runs as its own service user under systemd with NoNewPrivileges and ProtectSystem=strict, holding only the capability it needs to bind web ports                                                                                                                                                                                                                                                                                                     |

## Web hardening of the Licensing Authority

- Strict Content-Security-Policy (no inline scripts or styles), HSTS with preload, X-Frame-Options: DENY, nosniff, no referrer, cross-origin isolation headers, a restrictive permissions policy, noindex.
- API answers are never cached; request bodies are size-limited; the whole service is rate-limited per address.
- An optional allow-list restricts the console to listed networks (the system API stays reachable for check-ins).

## Network exposure

| Public name             | Served by                             | Behind                                                    |
|-------------------------|---------------------------------------|-----------------------------------------------------------|
| licensing.sentra-ai.com | Licensing Authority on 127.0.0.1:8640 | SENTRA AI Threat Shield (TLS, inspection, automatic bans) |
| updates.sentra-ai.com   | Release server on 127.0.0.1:8641      | SENTRA AI Threat Shield                                   |

Neither service is reachable directly from the internet. SENTRA's own public services are protected by the same product SENTRA sells.

# What customers should do

The practical steps that keep a SENTRA deployment as secure as it is designed to be.

---

## Accounts

- Turn on two-factor for every administrator, and set the policy to **require** it (GUARD and VIEW security policy).
- Give each person their own account with the least role they need; build custom roles rather than sharing an administrator account.
- Keep at least two top-level administrators, so one lost phone never locks you out.
- Store the glass-break account's password offline and use it only when the GUARD can't be reached.
- Review users and roles every quarter; remove leavers the same day.

---

## Licences

- Install keys only from SENTRA or your own VIEW or GUARD. Treat key bundles as confidential and delete local copies once installed.
- Bind stand-alone licences to the box's **Installation ID**.
- Allow outbound HTTPS to `licensing.sentra-ai.com` so revocations and renewals reach your boxes, unless your policy forbids any outside contact.

---

## Updates

- Keep systems on the **SENTRA release server** (never a developer source).
- Install offered updates promptly, at a time that suits you; read **What's new** and the expected downtime first.
- On fleets, lock GUARDs and nodes to **signed releases only** once every layer runs a version that supports it.
- Allow outbound HTTPS to `updates.sentra-ai.com` from the systems that update directly.

---

## Operations

- Back up GUARD, VIEW and box configurations regularly and keep copies off the server.
- Verify the audit ledgers periodically and investigate any break.
- Send security events to your own monitoring where the product supports it, and watch for alarms.
- Test the fail-open and bypass procedures in a maintenance window, so the team knows them before an emergency.

---

## Reporting a security issue

Write to [demo@sentra-ai.com](mailto:demo@sentra-ai.com). Please include the product, version and what you observed, and don't include passwords, keys or personal data.