



SENTRA
DOCUMENTATION

PRODUCT SPECS

Product Specifications

Datasheets for every SENTRA product

One datasheet per product (Web App · ATS, Internet · NTS, GUARD, VIEW, Licensing Authority) with capabilities, deployment, requirements, ports, integrations, licensing modules and security features, plus a side-by-side comparison.

Version 2026.10 27 October 2026

Contents

01 Products side by side

02 Web App · ATS — SENTRA AI Threat Shield

03 Internet · NTS — SENTRA Network Threat Shield

04 SENTRA GUARD

05 SENTRA VIEW

06 SENTRA Licensing Authority

Products side by side

A one-page comparison of every SENTRA product — job, placement, management, licensing and requirements.

	Web App · ATS	Internet · NTS	SENTRA GUARD	SENTRA VIEW	Licensing Authority
Job	AI web application firewall for websites and APIs	Inline network firewall for a whole internet line	Multi-tenant orchestrator (control plane)	Fleet command above GUARDs	Root of trust: licences and release signing
Layer	Enforce (data plane)	Enforce (data plane)	Orchestrate	Govern	Trust & supply
In the traffic path?	Yes — reverse proxy	Yes — bridge or routed, fail-open	No	No	No
Runs on	Linux server or VM, same server as the site or a dedicated edge	Ubuntu Server 22.04 / 24.04, VM or bare metal	Linux server or VM	Linux server or VM	SENTRA's server, behind SENTRA ATS
Stand-alone	Yes	Yes	Yes (with or without VIEW)	—	—
Managed by	Its own console; or a GUARD (node agent)	Its own console; or a GUARD (node agent)	Its own console; optionally VIEW	Its own console	Its own console
Console	Web, port 9280 (local by default, can be published; optional HTTPS)	Web, HTTPS port 8443 on the management port	Web, HTTP 8000 / HTTPS 8443	Web, HTTP 8010 / HTTPS 9443	https://licensing.sentra-ai.com
Licence modules	sentra, sentra-ids, sentra-ai (or all)	sentra-nts (or all)	GUARD licence (seats, tier) + authority certificate	Authority certificate (products, limits, latest expiry)	—
Monitor / Enforce	Yes	Yes	Per customer firewall	Per firewall, fleet-wide	—

	Web App · ATS	Internet · NTS	SENTRA GUARD	SENTRA VIEW	Licensing Authority
Two-factor sign-in	Yes	Yes	Yes	Yes	Forced
Custom roles	Yes	Yes	Yes	Yes	Fixed roles + per-user limits
Signed updates	Yes (release server)	Yes (release server)	Yes (from VIEW)	Yes (release server)	Publishes them
Audit	Tamper-evident ledger	Audit log	Audit log	Tamper-evident audit trail	Encrypted, hash-chained audit

NOTE

Hardware sizing depends on traffic volume, the number of sites or customers and which modules are on. Where this document says *sized per deployment*, contact SENTRA (demo@sentra-ai.com) with your traffic figures.

Web App · ATS — SENTRA AI Threat Shield

Datasheet for the AI web application firewall that protects websites and APIs (version 2.26).

Version: 2.26 · **Layer:** Enforce · **Type:** AI web application firewall (inspecting reverse proxy)

An AI web application firewall that sits in front of a website and inspects every request before it reaches the server. Injection, credential stuffing, scrapers, AI agent swarms and floods are stopped at the edge, with no change to the application.

Key capabilities

Inspection

- **Five checks on every request** — network, fingerprint, behaviour, AI intent and policy — scored together into one verdict. Clear attacks are blocked, likely ones challenged (browser check), everyone else passes straight through. The challenge and block lines are adjustable, with Balanced, Strict and Paranoid strategies.
- **Monitor and Enforce modes** — Monitor records what *would* be stopped; Enforce blocks, challenges and bans.
- **AI agents and swarms** — the good-agent passport verifies genuine AI agents and crawlers by signature and network and catches impostors; swarms that spread one job across many addresses are grouped into one campaign and contained.
- **Agent maze** — bots that keep coming back after being blocked get believable decoy pages instead of an error.
- **Virtual patches** for specific, actively exploited CVEs, with coverage reporting against the CISA Known Exploited Vulnerabilities list.
- **Breached-password check** at sign-in (k-anonymity; the password never leaves the box).
- **Rate limits, tarpit and automatic bans**; allow and block lists by address, network, country and ASN.

Site protection

- **Page Shield** — watches the scripts your pages load and alerts on new or injected ones.
- **API Shield** — validates content types, sizes, tokens and JSON fields per API route.
- **Response hardening** — adds missing security headers, cookie flags and HSTS; strips version banners.
- **Websites and certificates** (edge) — add websites, automatic Let's Encrypt certificates, per-site inspect or pass-through, emergency **bypass**.
- **Fail-open guard** (edge) — if the shield stops answering, traffic goes straight to the web server within seconds and an alert is raised.

Intelligence (module sentra-ids)

- **IDS hub** — Suricata and Zeek network sensors in one real-time view.
- **Threat feeds** — open IDS signature sets and IP reputation lists, updated automatically.

AI analyst & Copilot (module sentra-ai)

- AI second opinions on traffic, AI assessments and the **Copilot** assistant.
- Works with OpenAI-compatible providers, including **private on-premises models** (Ollama, vLLM, LM Studio, NIM).
- Cross-model disagreement telemetry flags adversarial probing.

Operations

- **Overview** dashboard, customisable per user (widgets, order, size) with a default layout for everyone; **Live** view; **wall** mode for NOC screens.
- **Incidents and campaigns, events, who** (visitor investigation), **bans**.
- **Policy** with staged changes, deploy and roll back to earlier revisions (deterministic policy compiler).
- **Reports** — report builder with sections, periods and filters; schedules by e-mail; archive; print/PDF and CSV. **Value report** with a clearly labelled estimate of what was saved.
- **Tools** — passive security **Scanner**, **Bot Scorecard**, and **Active test** (optional assessment tools: nmap, nikto, gobuster, nuclei, whatweb) for sites you own.
- **Security drills** to rehearse attacks safely.
- **LAN storm guard** — detects broadcast, multicast and ARP storms and switching loops.
- **Alerts** in plain language by Telegram and WhatsApp (English, Bahasa Melayu or both), daily and monthly summaries; e-mail (SMTP) for scheduled reports.

Deployment

Option	Description
Same server	Beside the website, forwarding over loopback. Isolated service user, CPU and memory caps. For a single site and trials.
Dedicated edge (<code>install --edge</code>)	Its own server holds ports 80/443 and the certificates; web servers sit behind it on private addresses. Recommended for production.
GUARD node	Provisioned and managed from SENTRA GUARD through the node agent; each customer gets their own firewall.

Requirements

Item	Requirement
Operating system	Linux with systemd; Debian/Ubuntu (apt) for the edge install (certbot, nftables)

Item	Requirement
Python	3.8 or newer
CPU / RAM / disk	Sized per deployment — contact SENTRA. The service runs within a memory cap of 768 MB (same server, 60% CPU cap) or 1.5 GB (edge).
Network	Edge: public ports 80/443. Outbound HTTPS for licence check-in, release checks and any feeds or AI providers you enable.

Ports and protocols

Port	Use
80, 443 (TCP)	Public websites (edge install) — HTTP redirect and HTTPS
9280 (TCP)	Dashboard — bound to 127.0.0.1 by default; publish opens it on the private IP; optional HTTPS (dashtls on)
9281, 9283 (TCP)	Demo site and SENTRA website lanes on a non-edge install (local by default)
443 (outbound)	Licensing check-in, release server, feeds, AI providers, alert channels — only those you enable

Integrations

Telegram, WhatsApp (Meta templates), SMTP e-mail; Suricata, Zeek; OpenAI-compatible AI providers and private models; S3-compatible storage, SFTP or a folder for off-box backups; Let's Encrypt; SENTRA GUARD (node agent and central identity).

Licensing

Module	Covers
sentra	The shield: inspection, Policy Fabric, bans, incidents, reports, dashboard
sentra-ids	Network sensors & threat feeds: Suricata, Zeek, threat feeds and signature updates
sentra-ai	AI analyst & Copilot

No user or site count. 30 days of grace for an unlicensed module, then *restricted*: protection continues, changes for that module pause.

Management and security

- Users with **fully customisable roles** (15 permissions in four groups; built-in Master Admin), **two-factor sign-in**, sign-in rules, and a **glass-break** account for when the managing GUARD is unreachable.

- **Central identity** from SENTRA GUARD when managed: GUARD users sign in with their GUARD accounts, mapped to local roles.
- **Tamper-evident audit ledger** of security decisions and admin actions.
- **Backups** nightly, with optional encrypted off-box destinations; restore from the console or command line.
- **Signed updates** from the release server, installed only when an admin chooses.

Internet · NTS — SENTRA Network Threat Shield

Datasheet for the inline network firewall that keeps a whole internet line clean (version 0.18).

Version: 0.18 · **Layer:** Enforce · **Type:** inline network firewall (layers 2, 3 and 4)

Protects a whole internet line, not just one website. NTS sits between the internet and the network and cleans all traffic for the organisation's IP addresses before it arrives — so mail servers, VPNs, remote access and every other service behind the line are protected.

How it sits in the line

Every physical port gets a role: **WAN**, **LAN**, **management** or **unused**. A WAN and a LAN port form a **segment**; one box can run several segments of both kinds:

Segment mode	Behaviour
Bridge	Bump in the wire: no IP addresses, no change on the customer side; default "accept" with stateful rules on top
Routed	NTS is the gateway with its own routing per segment, NAT and port forwards; default inbound "drop"

Fail-open: a bridge keeps forwarding even if SENTRA software stops; on bare metal a bypass network card can be driven by NTS.

Key capabilities

Protection layers

Layer	What it does
Edge protection	First thing after the network card on every WAN port: allow list, martians, block list, threat feeds, bans, per-source SYN / UDP / ICMP flood limits, customer prefixes with their own counters and limits
Policy	Ordered stateful rules per segment and direction (first match wins) with objects, services (36 built in), schedules, expiry, per-source connection limits and hit counts; rule analysis finds shadowed, redundant and unused rules
Box protection	Only the management port answers (SSH and the console, from allowed networks)
NAT	Masquerade, port forwards with source restriction and hairpin, 1:1 NAT, source NAT

Layer	What it does
AI daemon	Learns normal traffic per WAN port and per customer, flags floods, bans the top sources for a set time, refreshes feeds, alerts
Inspection	DNS and HTTPS-name (SNI) filtering (fail-open), firewall log to JSON and syslog, NetFlow v9 export
Shaping	Line rates, priority classes and caps by application or category
IPS	Suricata intrusion detection and prevention with protocol-based application control

Networking

- **VPN** — WireGuard for remote users and branches (client config and QR code shown once), IPsec IKEv2 for other firewalls; full or split tunnel.
- **Multi-WAN** — failover or weighted balance across several internet links, with health checks and alerts.
- **BGP** — announce your own space, take default or full tables, local preference, AS-path prepending, BFD, max-prefix, ECMP, and **remotely triggered blackhole** to ask carriers to drop attack traffic.
- **High availability** — two identical boxes with VRRP and synchronised sessions; failover in about 3 seconds; every deploy on the primary is sent to the secondary.
- **Interfaces** — bonds, VLANs, bridges, per-port MTU (jumbo frames), built from drop-downs.

Operations

- **Monitor and Enforce** for the protection layers; explicit policy rules always apply.
- **Commit-confirm** — every change goes live at once and rolls itself back after 60 seconds unless confirmed.
- **Staged changes and revisions** — see exactly what a deploy would change; every kept deploy is a numbered revision that can be compared and restored.
- **Dashboard** customisable per user, with a default layout for everyone; **wall** mode; light and dark themes.
- **Reports** — report builder with sections, schedules and e-mail; daily, weekly and monthly reports kept on the box.
- **Policy templates** — export a box's policy and import it on another; GUARD pushes templates to chosen NTS nodes.
- **Alerts** — Telegram, e-mail and webhook (Slack, Teams or any JSON endpoint), with a daily digest.
- **LAN storm guard** — broadcast, multicast and ARP limits per device on every LAN port.

Deployment

Option	Description
Virtual machine	Deployed by SENTRA GUARD or with the stand-alone installer
Bare metal / appliance	Stand-alone installer or deployed by GUARD; supports bypass network cards

Option	Description
HA pair	Two identical boxes plus a direct sync cable

Requirements

Item	Requirement
Operating system	Ubuntu Server 22.04 or 24.04
Packages	Installed automatically: nftables, iproute2, Python 3, ethtool, conntrack, netplan; optional Suricata (IPS), linux-modules-extra for DNS/HTTPS filtering; WireGuard, strongSwan, FRRouting, keepalived and conntrackd for the features that use them
Network ports	At least one WAN and one LAN port per segment, plus a management port; a dedicated sync port for HA
CPU / RAM / disk	Sized per deployment — contact SENTRA. For SME and home appliances see the hardware appliance guide.

Ports and protocols

Port / protocol	Use
8443 (TCP, HTTPS)	Console, on the management port only
22 (TCP)	SSH, on the management port only
UDP (configurable)	WireGuard VPN
UDP 500 / 4500	IPsec IKEv2
TCP 179	BGP to carriers
VRRP (unicast) and conntrackd	HA pair, over the sync link
UDP / TCP (configurable)	Syslog (RFC 5424) to a SIEM
UDP (configurable)	NetFlow v9 export
443 (outbound)	Licensing check-in, release server, feeds and alert channels you enable

Licensing

Module `sentra-nts` (or `all`), bound to the box's Installation ID. 30 days of grace, then changes, AI bans and feed updates pause — protection itself never stops.

Management and security

- Console with **two-factor sign-in** and **fully customisable roles** (16 permissions in 11 groups; built-in Admin).
- Managed by **SENTRA GUARD** through the node agent when part of a fleet, with central identity and a configurable mapping of GUARD roles to local roles; a **glass-break** admin works only while the GUARD is unreachable.
- Audit log of every change; **signed updates** from the release server, installed only when an admin chooses.

SENTRA GUARD

Datasheet for the multi-tenant orchestrator that runs SENTRA as a service (version 1.25).

Version: 1.25 · **Layer:** Orchestrate · **Type:** multi-tenant control plane

SENTRA GUARD runs SENTRA as a service. It onboards customers, provisions a firewall for each on the operator's nodes, watches node health and bandwidth live, moves customers between nodes, and handles licences, billing, alerts and access. GUARDS are named **OTG-OR-01, OTG-OR-02, ...**

Key capabilities

Area	Capabilities
Dashboard	Live radar cockpit with the hierarchy of nodes and firewalls; hover bandwidth for a bandwidth and latency graph; uplink, live threats and fleet health
Customers	Onboard customers with plan, contacts and KYC; one firewall per customer — Web App · ATS (website or API) or Internet · NTS (internet line / IP block); suspend, resume, close; customer self-service portal (<i>My firewall</i> , support tickets)
Instances	Every provisioned engine with state, mode (Monitor / Enforce) and node; move a firewall between nodes
Nodes	Register servers and VMs (one-time install command and agent token), live health, CPU, memory, uplink bandwidth and latency, capacity, drain, remove; per-node update policy and roll back
Security	Threats stopped across every firewall
NTS policies	Ready policies and templates with revisions, pushed to chosen NTS nodes
Licensing	This GUARD's licence and every box licence; acts as a signing authority certified by the SENTRA Licensing Authority, licensing its boxes automatically within its limits
Approvals	Maker-checker: sensitive actions need a second master admin; no self-approval
Billing	Plans and prices (or the price book pushed from VIEW), monthly billing run, invoices, payments, void; currency; <i>Billing on this GUARD</i> or <i>Billing from VIEW</i>
Reports	Build, save and schedule reports; print, PDF or CSV
Console	Run commands on a node (master admin only)
Users & roles	Fully customisable roles (35 permissions in 12 groups; built-in Master Admin), two-factor sign-in, password policy; who may sign in to each ATS / NTS box and with which role
Settings	Organisation, notifications and alerts (webhook, risk threshold, node-down alerts, daily digest), security policy, firewalls (dashboard addresses, GUARD public address, automatic firewall updates), integrations, billing defaults, SENTRA VIEW link, software updates, backups, system

Working with and without VIEW

- **On its own** a GUARD is a complete system: customers, nodes, billing, users and licensing.
- **Linked to VIEW**, it sends VIEW a snapshot every 2 seconds and carries out VIEW's changes through its own rules. If VIEW is unavailable the GUARD keeps working and catches VIEW up when it returns.
- Prices become read-only once VIEW pushes a fleet price book; with *Billing from VIEW* only VIEW bills its customers.

Deployment

Option	Description
From SENTRA VIEW	Deploy new orchestrator prepares the GUARD, with the billing mode chosen at deployment
Stand-alone	Installed on its own server, licensed by the SENTRA Licensing Authority

Requirements

Item	Requirement
Operating system	Linux with systemd; Debian/Ubuntu (apt) for the installer
Runtime	Python 3.8 or newer with venv (installed by the installer)
Database	SQLite by default; PostgreSQL for production (SENTRA_PANEL_DB)
CPU / RAM / disk	Sized per deployment — contact SENTRA
Network	Reachable by its nodes at its public address; outbound to its VIEW when linked

Ports and protocols

Port	Use
8000 (TCP, HTTP)	Console and API; node agents (configurable, 0TG_PORT)
8443 (TCP, HTTPS)	Built-in HTTPS for the console, self-signed or your own certificate (0TG_HTTPS_PORT)
Outbound to VIEW	Link to SENTRA VIEW (GUARD connects out)

Node agents connect **to** the GUARD (polling every 10 seconds by default); the GUARD never needs to reach into a node.

Licensing

A GUARD licence carries **seats** and a **tier**. With an authority certificate from the Licensing Authority (or from its VIEW), the GUARD issues chained keys for its ATS and NTS boxes within its limits and reports them back.

Management and security

Two-factor sign-in, customisable roles, maker-checker approvals, audit log of every action, nightly backups of the GUARD and every node under it (sent to VIEW and/or an extra S3, SFTP or folder destination, sealed with the GUARD's backup key), signed releases from VIEW (can be locked to signed releases only), and per-node control over updates.

SENTRA VIEW

Datasheet for fleet command above every SENTRA GUARD (version 1.25).

Version: 1.25 · **Layer:** Govern · **Type:** fleet command centre

SENTRA VIEW is the cockpit above every SENTRA GUARD: orchestrators, nodes and customers at once, with fleet traffic, alarms, tickets, central billing, the fleet price book, access across the fleet, licensing and software updates.

Key capabilities

Area	Capabilities
Fleet cockpit	Every orchestrator live; constellation of GUARDS; interactive hierarchy map VIEW → GUARDS → nodes → ATS / NTS with hover and click details
Orchestrators	Deploy, link and manage GUARDS; health, versions and traffic; jump in to a GUARD's console with a single-use pass, capped at a maximum role per GUARD
Customers, services, nodes	Every customer, firewall and node on every GUARD — and create or change them on any GUARD (customers, firewalls, moves, nodes); changes for an offline GUARD wait and run when it reconnects; conflicts are refused, never overwritten
Tickets	Customer faults and alarms with SLA, assignment, internal notes and replies shown in the customer portal (ticketing lives only in VIEW)
Billing	Central billing across all orchestrators: monthly runs, invoices, payments, voids, revenue and receivables
Plans & pricing	One fleet price book (plans, prices, currency) pushed to every GUARD, with per-GUARD status (in step, waiting, differs)
Access across the fleet	Users, roles, two-factor and password policy on every GUARD and down to each ATS / NTS box
Licensing	VIEW is a signing authority certified by the SENTRA Licensing Authority; licenses GUARDS and boxes automatically from the deployment plan, works disconnected, syncs its ledger; exceptions and stand-alone installs by Installation ID
Reports	Print, PDF or CSV
Team & access	VIEW's own staff, with fully customisable roles (only the Super Admin changes roles)
Audit trail	Every action, tamper-evident
Logs	Requests and security events
Backup & restore	Encrypted copies of VIEW's database
Software updates	VIEW itself, its GUARDS and their nodes (ATS, NTS, agent) from the signed release server

Built-in roles (all editable by the Super Admin)

Super Admin (fixed), OMM Admin, NOC Engineer, Billing Officer, Account Manager, Licence Requester, Licence Approver, Security Auditor, Viewer — plus any custom role. Each grant can be fleet-wide or scoped to an orchestrator or a customer.

Deployment

Installed on its own Linux server or VM with the VIEW installer, which also places the ATS and NTS code beside it so VIEW can hand releases to its GUARDs.

Requirements

Item	Requirement
Operating system	Linux with systemd; Debian/Ubuntu (apt) for the installer
Runtime	Python 3 with venv, git (installed by the installer)
Database	SQLite by default; PostgreSQL supported
CPU / RAM / disk	Sized per deployment — contact SENTRA
Network	Reachable by its GUARDs; outbound HTTPS to the release server and the Licensing Authority

Ports and protocols

Port	Use
8010 (TCP, HTTP)	Console, API and the GUARD link (configurable with <code>--port</code>)
9443 (TCP, HTTPS)	Built-in HTTPS for the console, self-signed or your own certificate (<code>--https-port</code>)
443 (outbound)	Release server and Licensing Authority

GUARDs connect to VIEW; VIEW never needs to reach into a GUARD.

Licensing

VIEW holds an **authority certificate** from the Licensing Authority listing the products, modules, limits and latest expiry it may grant. Keys it issues are chained (OTG2) and reported back.

Management and security

Two-factor sign-in, customisable roles with per-orchestrator and per-customer scope, maker-checker licence approvals (licence requester / approver), tamper-evident audit trail, encrypted

backups, and signed updates.

SENTRA Licensing Authority

Datasheet for the root of trust that issues every SENTRA licence and signs every release (version 1.2).

Version: 1.2 · **Layer:** Trust & supply · **Address:** <https://licensing.sentra-ai.com>

The root of trust for every SENTRA system. It holds the SENTRA root signing key, designs and issues every licence, certifies VIEWS and GUARDs as signing authorities with limits, keeps a record of every key ever issued, and approves and signs software releases. It is operated by SENTRA.

Key capabilities

Area	Capabilities
Design & issue	A wizard that designs the hierarchy for a customer (stand-alone ATS, stand-alone NTS, GUARD with boxes, VIEW fleet), then what each level needs: products, modules, term or perpetual, seats and limits
Approvals	Requests and approvals with per-user limits; the Super Admin's approval is final; optional <i>four eyes</i>
Registry	Deployments, every licence (reveal key with 2FA, revoke, CSV export), customers (archive or delete), signing authorities, systems seen online
Signing authorities	Enrolment codes; authority certificates with products, limits and latest expiry; ledger upload for offline authorities
Import	Bring existing licence files into the register
Releases	Candidates from the release builder with tests and what's new; Approve & publish with a fresh 2FA code; Withdraw
Security	Root signing keys, users and access, policy and vault, audit trail with one-click verification

Security design

Control	Implementation
Vault	Every record and audit entry encrypted with AES-256-GCM; master key wrapped with a key derived from the vault passphrase (scrypt, $N = 2^{17}$, 128 MiB); sub-keys by HKDF-SHA256; blind indexes so nothing — not even a customer name — is stored in the clear
Sealing	Sealed after every restart until the passphrase is entered; no way to recover a lost passphrase
Sign-in	scrypt password hashes; TOTP two-factor forced at first sign-in, with recovery codes; codes work once; lockout and rate limits

Control	Implementation
Sessions	In memory only; __Host- cookie (Secure, HttpOnly, SameSite=Strict); idle and absolute timeouts; CSRF token on every change
Step-up	Revealing keys, bundles, approvals, signing keys, users and policy need a fresh ZFA code
Roles	Super Admin, Administrator, Approver, Issuer, Auditor, Viewer, plus per-user limits (levels, longest term, perpetual, most boxes)
Audit	Encrypted entries chained with SHA-256 and sealed with an HMAC; verifiable in one click
Web hardening	Strict CSP, HSTS, frame deny, COOP/COEP/CORP, no-store, noindex, optional IP allow list, body size limits; locked-down systemd service behind SENTRA AI Threat Shield

Release signing

```

SENTRA root key (in the vault)
├─ certifies the release key (0TGRK1)
│   └─ signs each product's release list (index.json + index.sig)
│       └─ pins the checksum of each package and of every file

```

The release builder runs as a separate service user, tests each product and prepares candidates — it never signs. The release server (`updates.sentra-ai.com`) serves only public, signed files, so a second server for 1+1 availability needs only a copy of the published folder.

Interfaces used by SENTRA systems

Endpoint	Purpose
POST <code>/api/v1/licence/check</code>	Check-in from ATS and NTS: status of each key, revocations and renewals
POST <code>/api/v1/authority/enrol</code>	A VIEW or GUARD enrolls as a signing authority
POST <code>/api/v1/authority/sync</code>	An authority reports the keys it issued (signed with its own key)
GET <code>/api/v1/health</code>	Health check

Deployment

Runs on SENTRA's server on `127.0.0.1:8640` (release server `127.0.0.1:8641`), published through SENTRA AI Threat Shield, which terminates TLS and inspects every request.