

ForteBase Licensing

How licensing works across the DenseDefense platform: one signed, offline, fail-closed model that unlocks paid editions and features without ever calling home. Scan for free, license to remediate and prove. This guide is both the fast path (Section 2, licensed in about five minutes) and the model behind it.

● Ed25519-signed | verified offline, no phone-home

● Fail-closed: no valid license -> free baseline

● Fingerprint-bound or portable -- minted either way

● Editions & add-ons are per product

The 30-second version

FREE BY DEFAULT

Every install runs the free baseline -- recon, scan, and visible findings -- with no license at all.

LICENSE TO UNLOCK

Drop your `license.key` beside the product; it grants the paid features its `features` list names (remediation, signed evidence, offense, active-protect, and included capabilities like the free-for-life DDWitness attestation -- DDVault, the cloud evidence store, is the optional SaaS half).

OFFLINE, ALWAYS

Verification happens on your host against an embedded public key. No internet is required to license, ever -- the airgap is a first-class path.

CONTENTS

- | | |
|---|--|
| 1 What ForteBase is | 5 The per-product feature map |
| 2 Quick start -- licensed in about five minutes | 6 Machine binding -- bound or portable |
| 3 The license, in one paragraph | 7 Offline & air-gap -- no phone-home |
| 4 Editions & entitlements -- free unioned with paid | 8 The ForteBase scheduler |
| | 9 Fail-closed behavior & FAQ |

1 | What ForteBase is

ForteBase is the platform the DenseDefense products run on, and its licensing model is the one contract they all share. ForteFide handles compliance; ForteStrike and ForteLock handle offense and active-protect; new capabilities (DenseSense among them) arrive under the same roof. What ties them together for an administrator is licensing: *one signed, offline, fail-closed scheme* that decides which paid features each product will perform.

The design is deliberately boring in the way security should be: a license is a small signed statement -- your organization, an edition, a list of features, an optional host binding, an optional expiry. The product reads it, verifies the signature against a public key baked into the binary, and grants exactly what it names. Nothing calls home to check.

Tip -- Same model, per product. ForteStrike verifies through `fortebase_license`; ForteFide verifies through its own entitlement layer. They are parallel implementations of one design, not a single shared service -- so a license is scoped to the product family it was minted for, and its `features` list is what actually grants capability.

2 | Quick start -- licensed in about five minutes

Licensing is a file drop plus a read-back. The scan is free; the license turns on the doing. The example below is ForteFide (dashboard on `127.0.0.1:5000`, token as `X-Auth-Token`); ForteStrike follows the same shape at its own console.

1. See what you have now. A fresh install reads edition `free` (or `scanner`) and `licensed` false:

```
TOKEN=$(sudo cat /var/lib/fortefide/dashboard.token)
curl -s http://127.0.0.1:5000/api/capabilities \
  -H "X-Auth-Token: $TOKEN" | grep -E 'edition|licensed'
```

2. Install the license -- drop the key where the product reads it:

```
# ForteFide (or drag license.key onto the LICENSE panel in the dashboard)
sudo install -m600 license.key /var/lib/fortefide/license.key

# ForteStrike
sudo install -m600 license.key /opt/fortestrike/license.key
```

3. Verify it took. Entitlement re-reads live -- no restart:

```
curl -s http://127.0.0.1:5000/api/capabilities \
  -H "X-Auth-Token: $TOKEN" | grep -E 'edition|licensed'
```

The `edition` flips to your paid tier and the granted features report on. If it still reads `free`, the license did not verify -- a bad signature, a host-bound license on the wrong machine (Section 6), or an expired date. All three fall back to the baseline; Section 9 walks the fix.

Note -- Back up the existing `license.key` before any reinstall or purge -- a fresh install lands on the free baseline until you restore it. Verification is offline, so none of this needs the network.

3 | The license, in one paragraph

A ForteBase license is a JSON payload signed with **Ed25519**. The signing private key lives only on the DenseDefense minter (held in Infisical, never in a shipped binary and never in the repo); every client carries only the *public* half and uses it to verify. That is the whole trust model -- asymmetric signature, public verifier, no shared secret to leak.

FIELD	WHAT IT DOES
<code>org</code>	The licensed organization -- shown in the product and stamped into evidence.
<code>edition</code>	The tier name (e.g. <code>free</code> , <code>arsenal</code> , <code>full</code>).
<code>features</code>	The array that actually grants capability -- the product unions it onto the free baseline.
Host binding	Optional. A bound license names a machine fingerprint (Section 6); an unbound license runs anywhere.
Signature	Ed25519 over the payload. If it does not verify against the embedded public key, the license is ignored and the product stays on the free baseline.

Note -- Replacing the embedded public key with a placeholder is what once left a build *free-only*: with an all-zeros key no license could ever verify, and every install silently fell back to the baseline. The shipped key is real; a build that cannot verify a known-good license is misbuilt, not unlicensed.

4 | Editions & entitlements -- free unioned with paid

Entitlement is computed the same way everywhere: start from the **free baseline**, then union in whatever a valid license's `features` array grants. No license, an expired license, or a bad signature all resolve to the same safe place -- the baseline.

ALWAYS ON (FREE BASELINE)	GRANTED BY A LICENSE
Recon / discovery	Remediation & signed evidence (ForteFide)
Full, honest scan	Enrichment, exploit validation, lockdown, report export (ForteStrike)
Findings visible	Active-protect / containment (ForteLock)
--	DDWitness attestation -- free for the life of your license; DDVault, the optional cloud evidence store, holds the signed evidence; DenseSense regulated-data discovery is its own licensed product that can also attach as a ForteFide add-on

Tip -- The scan is free on purpose -- you get the honest posture number before you pay for anything. Licensing turns on the *doing*: fixing the host, proving it with signed evidence, running offense, standing active watch, and the added capabilities a license carries. DDWitness -- the free-for-life witness that attests a package is unaltered -- comes with any licensed edition; DDVault is the optional cloud evidence store that actually holds it. DenseSense is a standalone product you can license on its own or add on. If you cancel, DDVault keeps your evidence for 30 days so you can still retrieve it before it is removed.

5 | The per-product feature map

Each product names its own paid features. The license's `features` array is matched against these; anything not granted stays off, fail-closed.

PRODUCT	FREE BASELINE	PAID FEATURES
ForteFide (compliance)	Accounts for all 110 controls (78 scanned automatically, 32 attestation), see the score	Remediation, endpoint prepare, Ed25519-signed evidence package; DenseSense (regulated-data discovery) is a standalone licensed product that can also attach here as an add-on
ForteStrike (offense) -- <i>Arsenal</i> edition	Recon, scan, findings	<code>enrichment</code> , <code>exploit_check</code> , <code>lockdown</code> , <code>report_export</code>
ForteLock (active-protect, inside ForteStrike)	--	Firewall containment and, where authorized, source-remediation -- gated by the ROE and the license
DDWitness (suite-wide)	--	The free-for-life witness/attestation that a package is unaltered -- included with <i>any</i> licensed edition, not ForteFide alone; DDVault , the optional cloud evidence store, holds the signed evidence

Note -- Edition gating comes before license gating. A capability like DenseSense -- a standalone product that can also run as a ForteFide add-on -- is only present as a *ForteFide add-on* in a paid-edition build; on the free scanner edition the module is not shipped at all, so no license can flip it on there. The license decides *within* an edition that includes the capability.

6 | Machine binding -- bound or portable

A license can be minted **bound** to a host or **portable**. Binding uses a stable machine fingerprint - `/etc/machine-id` on Linux, the `MachineGuid` on Windows -- hashed to a hex digest. The fingerprint never leaves the host and is never phoned anywhere; it is only compared, locally, against the value in a bound license.

LICENSE KIND	BEHAVIOR
Portable (unbound)	Runs on any host. Convenient for a partner-delivered engagement or a short-lived VM.
Bound	Verifies only on the host whose fingerprint it names. Move it and it falls back to the free baseline -- the signature is still valid, but the binding is not.

Tip -- Fingerprinting never raises: if nothing readable is present it yields an empty value, which means an *unbound* license still works while a *bound* one will not verify. That is the fail-closed choice -- a missing fingerprint denies a bound license rather than accidentally honoring it.

7 | Offline & air-gap -- no phone-home

Verification is entirely local. The public key is embedded in the binary; the license sits on disk; the product checks the signature and the binding in-process. **There is no license server to reach and no internet requirement to license a product** -- the air-gapped enclave is a first-class path, not a degraded one.

This is the same posture the products hold everywhere else: ForteFide reaches your scoped targets over SSH / WinRM from the one host you run it on, and ForteStrike runs on-host with no telemetry. Licensing does not open a new outbound path; it closes the question locally.

High-impact -- Because there is no callback, a license cannot be revoked mid-flight over the network. Control is exercised at issuance -- scope with binding and expiry -- not by a kill switch. Mint bound, short-dated licenses where that control matters.

8 | The ForteBase scheduler

ForteBase also carries the platform's scheduler -- recurring, authorized runs on a fixed cadence. ForteFide's *ScanScheduler* re-scans a fleet on a schedule to prove controls are still met; ForteStrike's *Engagement Scheduler* is the offensive sibling, re-attacking on a schedule to prove the exposure is still closed.

Note -- The cadence is validated, not trusted. Only a known interval from `VALID_INTERVALS` is accepted; an unknown cadence is **rejected outright**, never silently coerced. This exists because a bogus interval once fell through a branch and ran daily -- and on a scheduler whose output feeds an FCA-grade attestation, "ran more often than you asked" is a defect, so the scheduler refuses the input rather than guessing.

9 | Fail-closed behavior & FAQ

I dropped in my license and nothing unlocked

Check `GET /api/capabilities` first. If the edition still reads `free`, the license did not verify: a bad or truncated signature, a host-bound license on the wrong machine (Section 6), or an expired date. All three resolve to the same safe place -- the free baseline -- by design.

Can one license unlock every product?

No. Licensing is per product family, and a license grants exactly the `features` its payload names. ForteFide and ForteStrike verify independently; buy the editions you need.

Does licensing need internet?

Never. Verification is offline against the embedded public key. If a workflow ever seems to wait on the network to license, that is not ForteBase licensing -- it is something else in the environment.

A capability is greyed out even though I am licensed

Edition gating precedes license gating. If the build is the free scanner edition, an added capability like DenseSense is not present to enable -- no license can flip it. Confirm the edition in `/api/capabilities`; you may need the paid-edition build, not a different license.

What happens to my evidence if I let a license lapse?

Already-generated, signed evidence stays valid -- the signature does not expire with the license. A lapsed license stops new paid actions (remediation, new signed packages, offense), returning the product to the free baseline until renewed.

ForteBase licensing governs paid editions across DenseDefense: ForteFide v26.09.06.0122 (compliance) and ForteStrike v0.8.3 with ForteLock (offense & active-protect), plus standalone products such as DenseSense licensed under the same model. The signing private key lives only on the DenseDefense minter; the client carries the public half and verifies offline. This guide describes shipped behavior; features a build does not include cannot be unlocked by any license.

DenseDefense | densedefense.com. Licensed use is per your order; a license binds to your organization and, where issued bound, to the host you registered.