TLS & Edge Security

Certificate management, reverse proxy configuration, hardening headers

Chapter 5 — TLS & Edge Security

5.1 Edge Model

Every public surface is fronted by an **nginx reverse proxy terminating TLS on the droplet that serves it**. No CDN or external proxy layer is in scope at this fleet size; origin concealment (TLS-04) is addressed in §5.6 within that constraint.

Surface	Droplet	Proxy → Upstream	State
`monitor.sbsdash.com`	`monitor-servers`	nginx 1.28.3 → Grafana 127.0.0.1:3000 (websockets on `/api/live/`)	**Live, verified 2026-07-17**
`wiki.sbsdash.com`	`sbs-wiki`	nginx → BookStack (localhost)	Live; re-verified at NYC2 rebuild
`sbsdash.com`, `www`	`sbsdash-server-prod`	nginx → Compartment frontend containers (Docker Compose, Chapter 6)	Pending — gates the apex repoint (§4.3)
`admin.sbsdash.com`	`sbsdash-server-prod`	nginx → admin console container	Pending
`clients.sbsdash.com`	`sbsdash-server-prod`	nginx → client surface container	Pending

Host-header routing on `sbsdash-server-prod` separates the three tenant surfaces behind one IP. A **default server block** returns `444` (connection close, no response) for any request whose `Host` /SNI matches no configured surface — direct-to-IP scans and unrecognized hostnames get nothing, including no certificate hint (see §5.2 default cert note).

5.2 Certificate Management (TLS-02)

- **CA and protocol:** Let's Encrypt via ACME. Issuance and renewal by certbot with the nginx plugin, auto-renewal by `certbot.timer` (systemd) — the pattern already live and verified on the monitor is the fleet standard.
- **One certificate per hostname.** No wildcard certificates (consistent with §4.6 CAA — `issuewild` not authorized) and no multi-SAN certificates spanning surfaces: a SAN list enumerating `admin.` and `clients.` on the apex cert would leak the surface inventory to any TLS client, defeating the FE-family enumeration controls. Per-hostname certs keep CT disclosure to exactly the hostname being served.

- **Challenge type:** HTTP-01 on port 80 per surface (matches the §3.5 firewall admits). DNS-01 challenges become available post zone cutover (§4.2) but are not required — HTTP-01 avoids granting the ACME client DNS-write credentials, a smaller blast radius.
- **Renewal monitoring:** certificate expiry is a Prometheus-alerted metric (blackbox exporter probe per surface, Chapter 12). A cert inside 14 days of expiry with renewals failing is a paged alert — `certbot.timer` failure is otherwise silent.
- **Default/fallback certificate:** the `444` default block carries a self-signed placeholder cert (nginx requires one to complete the handshake before it can close); it discloses no real hostname.
- **Key hygiene:** private keys remain on the serving droplet at `/etc/letsencrypt/live/<host>/`, root-owned 0600, never copied off-host, never committed. Keys are not vaulted — reissuance via ACME is cheaper and safer than key escrow.

5.3 Protocol Floor (TLS-01)

Fleet-standard nginx TLS parameters, IaC-templated (Chapter 10), identical on every surface:

- `ssl_protocols TLSv1.2 TLSv1.3;` — nothing below 1.2, ever. 1.3 preferred by client negotiation.
- TLS 1.2 cipher set restricted to AEAD/ECDHE (`ECDHE-ECDSA-AES128-GCM-SHA256`, `ECDHE-RSA-AES128-GCM-SHA256`, `ECDHE-*-AES256-GCM-*`, `CHACHA20-POLY1305`); no CBC, no RSA key exchange, no 3DES/RC4.
- `ssl_prefer_server_ciphers off;` (correct posture for a modern AEAD-only set — client picks its fastest).
- ECDSA P-256 leaf keys preferred (smaller handshakes); RSA-2048 acceptable where tooling requires.
- OCSP stapling on (`ssl_stapling on; ssl_stapling_verify on;`).
- Session tickets off (`ssl_session_tickets off;`) — avoids ticket-key rotation burden; session cache local, 10 m.
- Verification: `testssl.sh` run per surface at each gate; target grade A on SSL Labs semantics with no protocol/cipher findings. The monitor's live config is brought to this template at the next maintenance window (certbot's default bundle admits TLS 1.0/1.1 on some distro snippets — to be confirmed and corrected; Build Register).

5.4 HSTS (TLS-03)

- Port 80 exists solely for ACME HTTP-01 (`/.well-known/acme-challenge/`) and a 301 to HTTPS; no content is ever served over HTTP.
- Rollout in two steps: (1) at surface go-live, `Strict-Transport-Security: max-age=86400` — one day, recoverable if anything mis-serves; (2) after seven clean days, raise to `max-age=31536000; includeSubDomains; preload` fleet-wide and submit `sbsdash.com` to the Chrome preload list.
- `includeSubDomains` is safe **only because** the null-mail/no-other-services posture (§4.4) guarantees no plaintext-HTTP subdomain will ever exist. Preload submission is one-way in practice (removal takes months); it is executed as a named Build Register gate after Alexandra Del Rey's sign-off, since it permanently commits every future `*.sbsdash.com`

service to TLS.

5.5 Security Headers (APP-02) and Frontend Exposure Controls (FE-01…FE-06)

Fleet-standard header set, applied at nginx (`add_header ... always;`), IaC-templated; per-surface CSP tightening happens at the application layer:

Header	Value (baseline)	Control
`Strict-Transport-Security`	per §5.4	TLS-03
`X-Frame-Options`	`DENY`	APP-02 (clickjacking)
`X-Content-Type-Options`	`nosniff`	APP-02
`Referrer-Policy`	`no-referrer`	**FE-04** — no sbsdash.com URL ever appears in an outbound `Referer`
`Content-Security-Policy`	`default-src 'self'; frame-ancestors 'none'; base-uri 'self'; form-action 'self'` — tightened per app	APP-02 / **FE-05** (`'self'`-only enforces first-party assets: no CDN fonts, no third-party scripts, no analytics beacons)
`X-Robots-Tag`	`noindex, nofollow, noarchive, nosnippet`	**FE-01** — anti-indexing at the header layer, covers non-HTML responses too
`Permissions-Policy`	`camera=(), microphone=(), geolocation=(), interest-cohort=()`	APP-02
`Cross-Origin-Opener-Policy`	`same-origin`	APP-02

- **robots.txt (FE-02):** every surface serves `User-agent: * / Disallow: /`. Defense-in-depth with FE-01 — robots.txt alone is advisory; the header is the enforcement.
- **No social cards (FE-03):** no OpenGraph/Twitter meta tags in any served HTML; link unfurlers (Slack, iMessage, crawlers) get no title, description, or preview image. Grafana and BookStack template overrides to strip their defaults are Build Register items.
- **Neutral metadata (FE-06):** `server_tokens off;` (no nginx version); HTML `<title>` values are neutral ("Sign in" — not "SBS …"); no `meta description`; error pages are generic nginx defaults with no branding. The platform's public fingerprint is deliberately information-free: resolvable domain, TLS cert, sign-in page, nothing else.
- Grafana behind the proxy requires `proxy_set_header Host` + websocket upgrade headers on `/api/live/` (already configured); BookStack requires correct `X-Forwarded-Proto` to generate https URLs.

5.6 Origin Concealment (TLS-04)

Scoped honestly for a no-CDN architecture: the A records necessarily disclose droplet IPs, so concealment here means **the origin exposes nothing but the intended service**:

- Direct-to-IP requests hit the `444` default block — no content, no redirect, no certificate naming a real host.

- No service banner, version string, or debug endpoint on any public port (§5.5 neutral metadata; `ss -tlnp` gate confirms only 80/443/22 public).
- Non-public services (Grafana upstream 3000, Prometheus 9090, exporters 9100, future data services) bind 127.0.0.1 or VPC-private only — verified as-built on the monitor.
- If DDoS or scraping pressure later justifies it, fronting with Cloudflare proxy (post zone cutover) upgrades TLS-04 to true IP concealment; that decision is deferred and would arrive as a Change Order touching §4.2, §5.2 (cert model), and §3.5 (firewall admits narrowed to Cloudflare ranges).

5.7 Admin Surface Isolation (APP-03, edge layer)

`admin.sbsdash.com` receives edge-layer restrictions beyond the tenant surfaces, ahead of the application-layer controls in Chapter 9:

- Separate nginx server block, separate upstream container, separate access log.
- IP allowlist at the edge (`allow`/`deny` from the IaC admin-allowlist variable) — the admin console is not globally reachable even pre-authentication, unlike `sbsdash.com`/`clients.`.
- Stricter rate limit (`limit_req` zone: 5 r/s vs 20 r/s tenant baseline) and `client_max_body_size` scoped to actual admin upload needs.
- All four proxies (three tenant + admin) log in a common structured format shipped to the observability pipeline (Chapter 12) — access logs are a MON-01/MON-02 input.

5.8 Verification

Per-surface gate set (LH-SBS-INST-001): `testssl.sh` clean, protocol floor confirmed (no 1.0/1.1 handshake accepted); cert issued/renewing with expiry probe green; HTTP→HTTPS 301 with ACME path excluded; full header set present (curl -sI diff against template); robots.txt served; direct-IP request returns 444; no OG/Twitter tags in HTML; `server_tokens` absent; admin surface unreachable from a non-allowlisted IP; upstream ports absent from public `ss -tlnp`. The monitor surface — already live — is retro-fitted to the full template and re-verified as the reference implementation before the prod surfaces build against it.