Signed URL Hardening Checklist

12 items to verify before a paid-streaming launch goes live.

Use this checklist when scoping a new signed-URL deployment, auditing an existing one, or migrating between CDNs. Every item maps to a concrete artefact: a config field, a cache-key policy, a dashboard panel, an origin firewall rule. If you cannot point at the artefact, the item is not done.

- [ ] **1. Choose a signing algorithm and document key rotation**
  HMAC-SHA256 for Akamai/Google/Fastly/NGINX; RSA-SHA256 (RS256 JWT) for CloudFront/Mux/Cloudflare Stream.
  RSA lets you rotate the verifier independently; HMAC is cheaper but rotating the shared secret needs every signer updated at once.

- [ ] **2. Decide signed URL vs signed cookie per surface**
  Master playlist: signed URL bootstrap. Variants + segments + keys: signed cookie covering the protected path.
  Hybrid is the production default. Pure URL signing on every fetch is the cache-economy killer.

- [ ] **3. Strip signature parameters from the cache key**
  Verify Policy, Signature, Key-Pair-Id (CloudFront) and hdnts (Akamai) are excluded from the cache key.
  If a signature ends up in the cache key, hit ratio collapses to near zero and origin egress spikes.

- [ ] **4. Sign a path prefix, not every URL**
  Use an ACL or wildcard (e.g. /customer42/title-9000/*). One signature, all segments, stable cache key.
  This is also what fixes the hls.js bitrate-switch token-expiry failure mode (issue #1450).

- [ ] **5. Token expiry matches the longest plausible session**
  VOD ≤ runtime + 30 min. Live ≥ event duration + 30 min. Sessions long enough to break should refresh mid-stream.
  Plan token-renewal via app server + xhrSetup in hls.js, or a Set-Cookie refresh endpoint.

- [ ] **6. Pin the token to user identity, not just expiry**
  Bind the policy to IP CIDR, user ID, device fingerprint, or session ID — at least one beyond the timestamp.
  Pure timestamp-only tokens enable replay attacks within the validity window.

- [ ] **7. Watch the IPv6 trap (CloudFront)**
  CloudFront custom-policy IP restriction is IPv4-only. Disable IPv6 on the distribution, or omit the IP claim.
  iOS rollouts on IPv6-only carriers expose this silently in production.

- [ ] **8. Origin allows only CDN traffic**
  Origin Access Control (AWS S3), Authenticated Origin Pulls (Cloudflare), Site Shield (Akamai), or a custom shared-secret header.
  Without origin lockdown, the origin URL is one whois lookup away from being scraped.

- [ ] **9. Mutual TLS between CDN and origin for paid tier**
  CDN presents a client certificate; origin validates it. Forces every legitimate request through the CDN pipeline.
  Enterprise tier on every major CDN supports this; it is the default for OTT paid stacks.

- [ ] **10. Cache max-age ≤ signature validity for paid content**
  If max-age > expiry, the CDN serves a cached 200 after the original token has expired.
  Either keep max-age short or rely on cookie-bound sessions where the cookie carries the time bound.

- [ ] **11. Signing secret never leaves the server**
  Generate signatures in a backend service, edge function, or KMS. Never in front-end JavaScript or mobile bundle.
  Front-end secrets show up in view-source, app decompilation, or proxy traces within hours of launch.

- [ ] **12. Observability covers the failure modes**
  Dashboards: 403 rate by URL pattern, token-expiry rebuffer events, cache hit ratio for signed paths, origin egress by referrer.
  If a player sees more than 0.5% 403s on segments, a token or cache-key bug is in flight.