Audit Report: Avaleht - estply

Website: https://estply.com/
Date: 09.11.2026
Audit Coverage: 100% — all sources returned data
Confidence: low

Pages Audited (4 of 5):

- https://estply.com/
- https://estply.com/en/privacy-policy/
- https://estply.com/privaatsuspoliitika/
- https://estply.com/kupsiste-poliitika/

Summary of results

Overall Score: 90 / 100
Status: □ **Excellent**

What is pulling the score down

Page score = weighted PageSpeed categories (Performance 40%, Accessibility 30%, Best Practices 15%, SEO 15%) minus at most 5 deduction points, and never below 90 when every category is ≥90 and Security basics PASS. Points = largest deduction on any page.

Signal	Points (max)	Pages	Evidence (weakest page)
Field TTFB	−2	4/4	CrUX mobile TTFB p75 914 ms
W3C errors	−1	4/4	8 validator errors (5× defer on data-block scripts, 2× srcset without width)
Advanced security headers	−1	4/4	Failing: CSP is strict, Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy, Cross-Origin-Resource-Policy

Site overall 90 is the mean of 4 pages; scores range 88 (https://estply.com/en/privacy-policy/) → 92 (https://estply.com/kupsiste-poliitika/).

Per-page scores

□ **Excellent** · https://estply.com/

Score	Performance	Accessibility	Best Practices	SEO	Security
90	93	91	100	92	PASS

 Needs Improvement · https://estply.com/en/privacy-policy/

Score	Performance	Accessibility	Best Practices	SEO	Security
88	93	91	100	85	PASS

 Excellent · https://estply.com/privaatsuspoliitika/

Score	Performance	Accessibility	Best Practices	SEO	Security
91	97	91	100	92	PASS

 Excellent · https://estply.com/kupsiste-poliitika/

Score	Performance	Accessibility	Best Practices	SEO	Security
92	100	91	100	92	PASS

PageSpeed Insights — Mobile vs Desktop

*Lower is worse for Performance; higher is worse for LCP and CLS. Worse value is **bolded**.*

URL	Performance (M / D)	LCP (M / D)	CLS (M / D)
https://estply.com/	93 / **92**	**1.05 s** / 281 ms	0.000 / 0.000
https://estply.com/en/privacy-policy/	93 / **92**	**1.05 s** / 282 ms	**0.050** / 0.000
https://estply.com/privaatsuspoliitika/	**97** / 100	**1.05 s** / 281 ms	**0.000** / 0.000
https://estply.com/kupsiste-poliitika/	100 / 100	**1.05 s** / 281 ms	0.000 / 0.000

Optimization Checklist

0 of 2 passing — 0 pass · 0 warn · 2 fail · 6 n/a

Item	Status	Detail
Page caching plugin / CDN active	**Fail**	No WordPress cache plugin marker or CDN edge cache detected on the document response.
Response compressed (gzip / brotli)	N/A	Response is not compressed but only 337 bytes — too small to matter.
Images lazy-loaded	N/A	No raster elements found.
Hero image eagerly loaded	**Fail**	Hero image has loading="lazy", which delays LCP (inferred from DOM order/size — Lighthouse LCP element unavailable). Use loading="eager" (or omit loading) and add fetchpriority="high".
Hero is a real (not a CSS background-image)	N/A	No CSS background-images detected on raster-image-eligible elements.
Responsive images (srcset /)	N/A	Only 2 raster images on the page — responsive-image rule does not apply.
Reasonable number of image sizes	N/A	Too few raster images to evaluate srcset width variety.
JS scripts not blocking in	N/A	No external scripts on the page.

Fixes

Priority 1: Critical

Immediate action — impacts user experience, search rankings, or site safety.

1A. Reduce server response time (TTFB) `Performance`

- **Impact:** TTFB, FCP, LCP, Speed Index
- **Problem:** CrUX mobile TTFB p75 is 914 ms (desktop 709 ms) while lab TTFB is only 3 ms, and the document request took 4.42 s in the runtime capture — real users hit an uncached origin.
- **Solution:**
  - Verify WP Rocket page cache is actually serving the homepage (the `cache-control: max-age=0` response suggests a cache MISS).
  - Warm the cache after deploy and after content edits.
  - Put a CDN / edge cache in front of HTML so EU visitors are served from a nearby PoP.
  - Check PHP/DB time for the cold path (query monitor, slow query log) and enable object caching (Redis/Memcached).

1B. Remove render-blocking and unused CSS `Performance`

- **Impact:** Speed Index, FCP, LCP
- **Problem:** `render-blocking-insight` estimates 710 ms of savings and mobile Speed Index is 6.92 s versus 3.49 s on desktop, with 2 external stylesheets plus 3 inline blocks (9.6 KB).
- **Solution:**
  - Inline the critical above-the-fold CSS and load the rest with `media="print" onload="this.media='all'"` or `rel="preload"`.
  - Defer the two external stylesheets (`core.*.css` and the theme stylesheet) behind the critical CSS.
  - Audit the 3 inline style blocks for rules not used on the homepage and drop them.
  - Re-measure Speed Index after the change; target < 3.4 s on mobile.

1C. Add accessible names to buttons `Accessibility`

- **Impact:** Screen reader and voice-control access to controls
- **Problem:** Lighthouse `button-name` scores 0.00 with weight 10 — buttons on the page have no accessible name, which is the single largest accessibility failure.
- **Solution:**
  - Give every `<button>` visible text or an `aria-label`:

```
<button type="button" aria-label="Avaa haku"><svg aria-
hidden="true">…</svg></button>
```

- For icon-only toggles (menu, search, close), keep the label in sync with state (`aria-expanded`, `aria-label="Sulje valikko"`).
- Re-run Lighthouse Accessibility and confirm `button-name` passes.

1D. Harden the Content-Security-Policy `Security`

- **Impact:** XSS mitigation, injected-script containment
- **Problem:** The CSP is only `frame-ancestors 'self'` — it has no `default-src` and no `object-src 'none'`, so it restricts nothing about scripts or resources. The site exposes an upload-related anchor, so injected script would have somewhere to persist.
- **Solution:** Start from a report-only policy, then enforce:

```
Content-Security-Policy: default-src 'self'; object-src 'none';
base-uri 'none';
  frame-ancestors 'self'; script-src 'nonce-{random}' 'strict-
dynamic' https:;
  style-src 'self' 'unsafe-inline'; img-src 'self' data:
https://estply.com;
  report-uri /csp-report
```

  - Use nonces + `'strict-dynamic'` rather than a flat host allowlist.
  - Roll out with `Content-Security-Policy-Report-Only` first and fix violations before enforcing.

1E. Enlarge touch targets `Accessibility`

- **Impact:** WCAG 2.5.8 target size
- **Problem:** Lighthouse `target-size` scores 0.00 with weight 7 — tap targets are smaller or closer than the 24×24 px minimum.
- **Solution:**
  - Give icon links and buttons at least 44×44 px hit area via padding or `min-width` / `min-height`.
  - Add spacing between adjacent inline links so their hit areas do not overlap.

1F. Add a meta description `SEO`

- **Impact:** Search snippet quality, CTR
- **Problem:** Lighthouse `meta-description` scores 0.00 and the captured document has no description tag.
- **Solution:** Add a unique, ~150-character description per page:

```
<meta name="description" content="Estply küpsiste poliitika:
```

```
milliseid küpsiseid me kasutame ja kuidas neid hallata.">
```

1G. Load the hero image eagerly `Performance`

- **Impact:** LCP
- **Problem:** The first/largest image (`6c041c12…png` , 251×134) carries `loading="lazy"` and no `fetchpriority` , delaying the LCP candidate.
- **Solution:**

  ```
  <img src="…" width="251" height="134" loading="eager"
  fetchpriority="high" alt="">
  ```

  Keep `loading="lazy"` only on images below the fold.

1H. Stop rate-limiting legitimate page requests `Best Practices`

- **Impact:** Crawlability, availability, all audit signals
- **Problem:** The audit fetch returned HTTP 429 with `retry-after: 10` , and the captured document title was "429 Too Many Requests" (337 B body) — the WAF (ZoneOS) is throttling ordinary page views.
- **Solution:**
  - Allowlist verified search-engine crawlers and monitoring user agents at the WAF/CDN.
  - Raise the per-IP request threshold and serve cached HTML instead of a 429 for anonymous GETs.
  - Return a real page (with `lang` , title, meta description) on any error response.

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Serve images in WebP/AVIF at the right size `Performance`

- **Impact:** Page weight, LCP, Speed Index
- **Problem:** `image-delivery-insight` estimates 231 KiB of savings, and the runtime capture shows 8 image requests totalling 2.66 MB of a 2.72 MB page.
- **Solution:**
  - Re-encode the remaining JPEG/PNG assets (e.g. the 324×216 and 520×140 thumbnails) to WebP/AVIF.
  - Tighten the 28 generated srcset widths — the theme emits sizes up to 2560 px for thumbnails displayed at 324 px.
  - Cap WordPress intermediate sizes to the widths actually used in the layout.

2B. Fix W3C validation errors `Best Practices`

- **Impact:** Markup correctness, image loading behaviour, crawler parsing
- **Problem:** The validator reports 11 errors, including 5× `defer` on `type="text/rocketlazyloadscript"` data blocks, an empty `href` on a `dns-prefetch` link, and `srcset` width descriptors without matching `sizes`.
- **Solution:**
  - Remove `defer` / `async` from WP Rocket's `text/rocketlazyloadscript` data blocks (they are not executable scripts).
  - Drop the empty `<link data-rocket-prefetch href="" rel="dns-prefetch">` or give it a real host.
  - Add `sizes` wherever `srcset` uses width descriptors, and add `loading="lazy"` to images using `sizes="auto"`.
  - Re-validate at validator.w3.org after the theme/WP Rocket update.

2C. Enable HSTS preload `Security`

- **Impact:** First-visit downgrade protection
- **Problem:** HSTS is set to `max-age=31536000; includeSubDomains` but lacks the `preload` directive, so the domain cannot be submitted to the preload list.
- **Solution:**
  - Add `preload` and confirm the max-age is at least 31536000:

```
Strict-Transport-Security: max-age=63072000; includeSubDomains;
preload
```

  - Submit the domain at hstspreload.org and keep all subdomains HTTPS-only before doing so.

2D. Make link text descriptive `SEO`

- **Impact:** WCAG 2.4.4, crawlability
- **Problem:** Lighthouse `link-text` fails with 2 links, and the inventory shows "here" repeated twice plus other vague anchors.
- **Solution:**
  - Replace "here" / "read more" with the destination, e.g. "read our cookie policy".
  - Where the visual design requires short text, add `aria-label` with the full context.

2E. Cache the HTML document `Performance`

- **Impact:** TTFB on repeat views

- **Problem:** The document is served with `cache-control: max-age=0` and an `Expires` equal to the request time, forcing full revalidation on every visit while field TTFB sits at 914 ms.
- **Solution:**
  - Let WP Rocket serve a short but non-zero HTML TTL, e.g. `Cache-Control: public, max-age=600, stale-while-revalidate=60`.
  - Keep `Vary: Accept-Encoding` but avoid per-cookie cache keys for anonymous visitors.

2F. Set the document language and landmark structure `Accessibility`

- **Impact:** WCAG 3.1.1 Language of Page, 1.3.1 Info and Relationships
- **Problem:** axe-core reports `html-has-lang` (serious) plus `landmark-one-main` and `region` on the captured document, which had no `lang`, no `<main>` and no `<nav>`.
- **Solution:**
  - Add `<html lang="et">` (or the correct language) to the theme header.
  - Wrap the policy content in `<main>` and the site navigation in `<nav>`.
  - Re-check the real page after the 429 is resolved, since the capture may reflect the error template only.

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. Add Referrer-Policy, Permissions-Policy, COOP and CORP `Security`

- **Impact:** Referrer leakage, cross-origin isolation, feature abuse
- **Problem:** Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are all missing from the response.
- **Solution:**

```
Referrer-Policy: strict-origin-when-cross-origin
Permissions-Policy: geolocation=(), microphone=(), camera=(),
payment=()
Cross-Origin-Opener-Policy: same-origin
Cross-Origin-Resource-Policy: same-site
X-Permitted-Cross-Domain-Policies: none
```

3B. Add Subresource Integrity to third-party scripts `Security`

- **Impact:** Supply-chain tampering of third-party JS

- **Problem:** 1 of 1 cross-origin scripts — `iframemanager.js` from cdn.jsdelivr.net — lacks an `integrity` attribute.
- **Solution:**

```
<script
src="https://cdn.jsdelivr.net/gh/orestbida/iframemanager@1.3.0/dist
/iframemanager.js"
        integrity="sha384-…" crossorigin="anonymous" defer>
</script>
```

  - Generate the hash with `openssl dgst -sha384 -binary file.js | openssl base64 -A`.
  - Pin the exact version (already `@1.3.0`) so the hash stays valid.

3C. Hide server version headers `Security`

- **Impact:** Attack-surface reconnaissance
- **Problem:** The response discloses `Server: Apache / ZoneOS`, which tells an attacker the stack and narrows exploit selection.
- **Solution:**
  - Apache: `ServerTokens Prod` and `ServerSignature Off`.
  - At the CDN/proxy layer, strip or rewrite the `Server` header.

3D. Add structured data `SEO`

- **Impact:** Rich results eligibility
- **Problem:** The `structuredData` audit fails; the single JSON-LD block is reported with type `Unknown`, so no Organization/Product entity is recognised.
- **Solution:**
  - Emit `Organization` + `WebSite` JSON-LD on the homepage and `Product` / `BreadcrumbList` on product pages.

```
{"@context":"https://schema.org","@type":"Organization","name":"EST
PLY","url":"https://estply.com/","logo":"https://estply.com/…"}
```

  - Validate with Google's Rich Results Test.

3E. Verify color contrast manually `Accessibility`

- **Impact:** WCAG 1.4.3 contrast
- **Problem:** axe-core returned 0 violations but flagged 9 nodes as incomplete for color contrast, so contrast on those elements is unverified.
- **Solution:**

    - Check the 9 flagged nodes in DevTools with the contrast picker, including any text over gradients or with alpha transparency.
    - Target ≥4.5:1 for body text and ≥3:1 for large text.
    - Re-test with a browser extension after any brand-colour change.

3F. Protect the WordPress login and installer `Security`

- **Impact:** Brute-force and enumeration surface
- **Problem:** The login form is still served on the default `/wp-login.php` path (other WordPress hygiene checks pass).
- **Solution:**
    - Put `/wp-login.php` behind a WAF rate limit or IP allowlist.
    - Add a login-path plugin or basic-auth on the file, and enforce 2FA for admin accounts.
    - Keep `xmlrpc.php` blocked (already passing).