Audit Report: Avaleht - estply

Website: https://estply.com/
Date: 29.10.2026
Audit Coverage: 100% — all sources returned data
Confidence: high

Pages Audited (5 of 5):

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

Summary of results

Overall Score: 78 / 100
Status: **Needs Improvement**

Site overall 78 is the mean of 5 pages. Scores range 66 (https://estply.com/) → 88 (https://estply.com/privaatsuspoliitika/). Weakest page: Mobile performance is the dominant drag: PSI mobile 74 vs desktop 79, but the underlying metrics diverge sharply — LCP 4.7 s mobile vs 2.7 s desktop and Speed Index 11.11 s vs 2.78 s, both far past the heavy-penalty thresholds, while TBT (0 ms), TTFB (3 ms), FCP (1.2 s) and CLS (0.000) are all clean. The runtime capture shows 2.70 MB of the 2.76 MB total is images across just 8 requests, and PSI's image-delivery insight estimates 276 KiB of savings, so image payload is the single biggest lever. Accessibility is split: axe-core reports 0 violations across 40 passes, but PSI accessibility lands at 91/96 with `button-name` and `target-size` scoring 0.00, and axe flagged 9 color-contrast nodes as incomplete. SEO sits at 85 with no meta description, an invalid robots.txt and no usable structured data, and W3C reports 12 errors (mostly WP Rocket `text/rocketlazyloadscript` blocks and malformed `srcset` / `sizes`). Security basics PASS — quoted disclaimer applies: "These are high-level hygiene checks on HTTP headers, TLS, cookies and a few well-known exposed paths. PASS means the basics are in place. It does not mean the site is secure: application logic, authentication, plugins, server configuration and dependencies are not tested here. Treat FAILED as a must-fix and PASS as a starting point, not a certificate." — but the advanced tier fails on CSP strictness, HSTS preload, Referrer-Policy, Permissions-Policy, COOP/CORP and SRI, which keeps the SEO/Security budget partial.

Per-page scores

 Needs Improvement · https://estply.com/

Score	Performance	Accessibility	Best Practices	SEO	Security
66	74	91	81	85	PASS

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

Score	Performance	Accessibility	Best Practices	SEO	Security
73	86	91	100	77	PASS

 Needs Improvement · https://estply.com/privaatsuspoliitika/

Score	Performance	Accessibility	Best Practices	SEO	Security
88	96	91	100	92	PASS

 Needs Improvement · https://estply.com/kupsiste-poliitika/

Score	Performance	Accessibility	Best Practices	SEO	Security
84	91	91	100	92	PASS

 Needs Improvement · https://estply.com/kontakt/

Score	Performance	Accessibility	Best Practices	SEO	Security
79	89	92	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/	**74** / 79	**4.65 s** / 2.68 s	0.000 / **0.000**

URL	Performance (M / D)	LCP (M / D)	CLS (M / D)
https://estply.com/en/privacy-policy/	**86** / 99	**3.30 s** / 721 ms	0.000 / 0.000
https://estply.com/privaatsuspoliitika/	**96** / 99	**2.55 s** / 721 ms	**0.000** / 0.000
https://estply.com/kupsiste-poliitika/	**91** / 100	**3.45 s** / 721 ms	0.000 / 0.000
https://estply.com/kontakt/	**89** / 100	**3.45 s** / 721 ms	0.000 / 0.000

Optimization Checklist

4 of 5 passing — 4 pass · 0 warn · 1 fail · 3 n/a

Item	Status	Detail
Page caching plugin / CDN active	**Pass**	Caching plugin detected (WP Rocket)
Response compressed (gzip / brotli)	**Pass**	Document response is compressed with gzip.
Images lazy-loaded	**Pass**	All non-hero raster images use loading="lazy".
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.

Item	Status	Detail
Reasonable number of image sizes	N/A	Too few raster images to evaluate srcset width variety.
JS scripts not blocking in	**Pass**	No render-blocking scripts in .

Fixes

Priority 1: Critical

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

1A. Cut mobile LCP (4.7 s) and Speed Index (11.1 s) `Performance`

- **Impact:** LCP, Speed Index, mobile Performance score
- **Problem:** Mobile LCP is 4.7 s against 2.7 s on desktop, and Speed Index is 11.11 s against 2.78 s — both far beyond the heavy-penalty thresholds, while TBT is 0 ms and TTFB is 3 ms, so the bottleneck is render/asset delivery, not server or main-thread work.
- **Solution:**
  - Preload the actual LCP image and serve it at the right size:

```
<link rel="preload" as="image" href="/wp-
content/uploads/2026/10/kaasik-kaupo-kalda-360x640.webp"
      imagesrcset="..." imagesizes="..." fetchpriority="high">
```

  - Work through PSI's `lcp-breakdown-insight` and `network-dependency-tree-insight` to find the render-blocking chain ahead of the hero.
  - Inline the critical CSS for the above-the-fold section and defer the two external stylesheets.
  - Re-test on throttled mobile after the image work below — Speed Index at 11 s usually collapses once hero bytes drop.

1B. Reduce the 2.70 MB image payload `Performance`

- **Impact:** LCP, Speed Index, total page weight
- **Problem:** Browser runtime shows 8 image requests totalling 2.70 MB out of 2.76 MB total page weight, and PSI's image-delivery insight estimates 276 KiB of savings.
- **Solution:**

- Convert the remaining JPG/PNG assets (e.g. `kaupokalda-com-dji_...jpg`, `estply-cta-1-520x140.jpg`, the two 251×134 PNGs) to WebP/AVIF; only the hero is WebP today.
- Tighten the srcset ladder — 27 distinct widths up to 2560 px is excessive for a 360–520 px display slot; cap at the largest rendered size ×2.
- Re-encode with quality ~75 and verify the 276 KiB Lighthouse saving lands.

1C. Name all buttons and enlarge touch targets `Accessibility`

- **Impact:** PSI Accessibility (91 mobile / 96 desktop), WCAG 4.1.2, 2.5.8
- **Problem:** PSI accessibility audits `button-name` and `target-size` both score 0.00 (weights 10 and 7) on mobile, even though axe-core reported 0 violations — the two tools are inspecting different DOM states.
- **Solution:**
  - Give every `<button>` visible text or an `aria-label`:

```
<button type="button" aria-label="Ava menüü"><svg aria-
hidden="true">…</svg></button>
```

  - Ensure interactive targets are at least 44×44 px with 8 px spacing (WCAG 2.5.8); pad icon-only controls rather than relying on the glyph box.
  - Re-run axe after JS hydration — the 0-violation axe pass likely ran before the menu/carousel buttons rendered.

1D. Load the hero image eagerly with high fetchpriority `Performance`

- **Impact:** LCP, Speed Index
- **Problem:** The optimized-web checklist fails 'Hero image eagerly loaded': the first raster image (`6c041c12…png`, 251×134) has `loading="lazy"` and no `fetchpriority`, and mobile LCP is 3.3 s vs 0.7 s on desktop.
- **Solution:**

```
<img src="/wp-content/uploads/2026/08/6c041c12….png"
     alt="…"
     width="251" height="134"
     loading="eager"
     fetchpriority="high"
     decoding="async">
```

  - Remove `loading="lazy"` from the LCP element only; keep it on below-the-fold images.
  - If the real LCP element is a heading or background, preload it instead:

```
<link rel="preload" as="image" href="…" fetchpriority="high">
```

1E. Eliminate render-blocking resources `Performance`

- **Impact:** Speed Index, FCP, LCP
- **Problem:** PSI flags `render-blocking-insight` with an estimated 690 ms savings and `network-dependency-tree-insight`; Speed Index is 6.83 s on mobile against 1.33 s on desktop.
- **Solution:**
  - Inline critical above-the-fold CSS and defer the two external stylesheets (34.6 KB total) via `media="print" onload="this.media='all'"` or WP Rocket's 'Optimize CSS delivery'.
  - Add `preconnect` / `preload` for the font files (`open-sans-regular.woff2`) and the jsdelivr origin.
  - Flatten the dependency chain: the document took 3.96 s in the runtime capture while TTFB was 3 ms, so the delay is client-side chain depth, not server response.

1F. Give every button an accessible name `Accessibility`

- **Impact:** WCAG 4.1.2, Lighthouse accessibility score
- **Problem:** Lighthouse `button-name` scores 0.00 with weight 10 — buttons on the page have no accessible name, which is the single largest accessibility deduction (mobile 91).
- **Solution:**
  - Add visible text or `aria-label` to each `<button>`:

```
<button type="button" aria-label="Open cookie settings">
  <svg aria-hidden="true" focusable="false">…</svg>
</button>
```

  - Mark decorative icons inside buttons `aria-hidden="true"` so the label is the only announced text.
  - Re-run axe/Lighthouse to confirm `button-name` reaches 1.00.

1G. Load the hero image eagerly with high fetch priority `Performance`

- **Impact:** LCP (mobile 3.5 s vs desktop 0.7 s)
- **Problem:** The optimized-web checklist flags the hero image (uploads/2026/08/6c041c12fb0c531933532c08c2749a1a836d595a.png) as

loading="lazy" with no fetchpriority, and mobile LCP is 3.5 s against a 0.7 s desktop value.

- **Solution:**
  - Set `loading="eager"` (or omit `loading`) and `fetchpriority="high"` on the LCP image.
  - Add a preload hint in `<head>`:

```
<link rel="preload" as="image" href="/wp-content/uploads/2026/08/6c041c12fb0c531933532c08c2749a1a836d595a.png" fetchpriority="high">
```

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

1H. Load the LCP image eagerly with high fetchpriority `Performance`

- **Impact:** LCP (mobile 3.5 s), Speed Index (3.83 s)
- **Problem:** Mobile LCP is 3.5 s against 0.7 s on desktop, and the optimization checklist fails "Hero image eagerly loaded" — the flagged hero (cz5i9178-100x100.webp) carries `loading="lazy"` and no `fetchpriority`.
- **Solution:** Identify the real LCP element in DevTools (Performance → LCP) and make it eager:

```
<img src="/wp-content/uploads/.../hero.webp"
     srcset="... 800w, ... 1600w" sizes="100vw"
     width="1600" height="900"
     fetchpriority="high"
     alt="...">
```

  - Remove `loading="lazy"` from that one image only; keep it on everything below the fold.
  - If WP Rocket is rewriting the tag, exclude the hero from its lazyload list.

1I. Complete the Content-Security-Policy (default-src, object-src 'none', nonce + strict-dynamic) `Security`

- **Impact:** XSS mitigation, plugin/object injection
- **Problem:** The CSP is only `frame-ancestors 'self'` — no `default-src`, no `object-src 'none'`, and no script/resource restrictions. The page contains a form with a `<textarea>` (hasUserContent signal), so injected script would run against real user input.
- **Solution:** Deploy a nonce-based policy rather than a flat allowlist:

```
Content-Security-Policy: default-src 'self'; object-src 'none';
base-uri 'none';
```

```
  script-src 'nonce-{RANDOM}' 'strict-dynamic' https:;
  style-src 'self' 'unsafe-inline'; img-src 'self' data:;
  frame-ancestors 'self'; form-action 'self'; upgrade-insecure-
requests
```

- Roll out in `Content-Security-Policy-Report-Only` first and watch reports for a week before enforcing.
- WordPress + WP Rocket inject inline scripts; the nonce must be generated per request and passed to the inline `<script>` tags.

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Add meta description, fix robots.txt, add structured data `SEO`

- **Impact:** SEO score (85), SERP snippet, rich results eligibility
- **Problem:** PSI SEO audits `meta-description`, `robots-txt` and `structured-data` all fail; the HTML inventory confirms `Description: not set` and a single JSON-LD block of type "Unknown".
- **Solution:**
  - Add a unique description per page:

```
<meta name="description" content="ESTPLY toodab kvaliteetset
kasevineeri Jõgevamaal – ...">
```

  - Serve a valid `robots.txt` at the root (currently reported invalid) and reference the sitemap.
  - Replace the untyped JSON-LD with a real `Organization` / `Manufacturer` graph including `name`, `url`, `logo`, `address` and `sameAs`.

2B. Harden the Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth, Lighthouse Best Practices
- **Problem:** The CSP is `frame-ancestors 'self'` only — no `default-src`, no `object-src 'none'`, and no script or resource restrictions, so it provides clickjacking protection but no XSS mitigation.
- **Solution:** The site has no auth, no payments and no forms on this page, so this is Priority 2 rather than 1 — but the `upload` anchor in the nav means user-supplied files may exist elsewhere, so tighten it before that surface grows. Use a nonce + `'strict-dynamic'` policy rather than a flat allowlist:

```
Content-Security-Policy: default-src 'self'; object-src 'none';
base-uri 'none';
```

```
  script-src 'nonce-{random}' 'strict-dynamic'
https://cdn.jsdelivr.net;
  style-src 'self' 'unsafe-inline'; img-src 'self' data:; frame-
ancestors 'self'
```

Roll it out in report-only mode first — WP Rocket's lazyload rewriting and the CookieYes banner will need nonces.

2C. Fix the 12 W3C validation errors `Best Practices`

- **Impact:** HTML validity, image loading behavior, Best Practices
- **Problem:** W3C reports 12 errors, led by 5× `defer` on `type="text/rocketlazyloadscript"` data blocks, an empty `href` on a `dns-prefetch` link, `src` on `<source>`, and two `srcset` values missing width descriptors.
- **Solution:**
  - The `rocketlazyloadscript` errors are WP Rocket's lazyload rewriting; exclude the affected scripts from delay or switch to native `loading="lazy"` where possible.
  - Remove the empty prefetch: `<link data-rocket-prefetch href="" rel="dns-prefetch">` → delete or give it a real origin.
  - Add width descriptors to the two offending `srcset` values:

```
<img srcset="...6c041c12....png 251w" sizes="251px" loading="lazy"
width="251" height="134" alt="">
```

  - Add `loading="lazy"` to the `<img>` using `sizes="auto"` at line 310, or drop the `auto` prefix.

2D. Stop serving the HTML document with max-age=0 `Performance`

- **Impact:** Repeat-visit TTFB, FCP, LCP
- **Problem:** The document response carries `cache-control: max-age=0` with `expires` equal to the request time, so every navigation revalidates the 87 KB HTML even though WP Rocket is installed.
- **Solution:**
  - Let WP Rocket manage HTML caching and emit a short but non-zero TTL for anonymous visitors, e.g. `Cache-Control: public, max-age=600, s-maxage=3600` at the CDN edge.
  - Keep `max-age=0, no-cache` only for logged-in or cart sessions.
  - Verify the canonical URL is warmed after deploy so the first visitor does not pay the cold-start cost.

2E. Enlarge touch targets to 44×44 px `Accessibility`

- **Impact:** WCAG 2.5.8, mobile usability
- **Problem:** Lighthouse `target-size` scores 0.00 (weight 7) and the SEO `tapTargets` audit also fails — interactive elements are too small or too close together on mobile.
- **Solution:**

```
.icon-link, .menu a, .footer a {
  min-width: 44px;
  min-height: 44px;
  display: inline-flex;
  align-items: center;
  justify-content: center;
}
```

  - Add spacing (`gap` / `margin`) so adjacent targets are at least 8 px apart.

2F. Add a meta description `SEO`

- **Impact:** Search snippet CTR, Lighthouse SEO (77 mobile)
- **Problem:** HTML inventory shows `Description: not set` and Lighthouse `meta-description` scores 0.00.
- **Solution:**

```
<meta name="description" content="How estply collects, uses and
retains personal data, and how to exercise your GDPR rights.">
```

  - Keep it 120–155 characters and unique per page; in WordPress this is usually the Yoast/RankMath field.

2G. Fix the invalid robots.txt `SEO`

- **Impact:** Crawlability, Lighthouse SEO
- **Problem:** Lighthouse `robots-txt` scores 0.00 — 'robots.txt is not valid' — which can block or confuse crawlers.
- **Solution:**
  - Fetch `https://estply.com/robots.txt` and validate it.
  - Ensure it is plain text, starts with `User-agent:` , and contains no HTML (a WordPress 404 page served at that path is the usual cause).
  - Reference the sitemap explicitly:

```
User-agent: *
Allow: /
```

```
Sitemap: https://estply.com/sitemap_index.xml
```

2H. Fix srcset candidates missing width descriptors `Best Practices`

- **Impact:** W3C validity, responsive image selection
- **Problem:** W3C reports 2 errors: `srcset` values on the two `<img>` elements at lines 536 and 556 specify no width, which is invalid when `sizes` is present.
- **Solution:**

```
<!-- before -->
<img srcset="/…/6c041c12….png" sizes="(max-width: 600px) 100vw,
251px">
<!-- after -->
<img srcset="/…/6c041c12….png 251w" sizes="(max-width: 600px)
100vw, 251px">
```

  - Every candidate in a `srcset` must carry a `w` or `x` descriptor.

2I. Remove defer/async from WP Rocket lazyload data blocks `Best Practices`

- **Impact:** W3C validity (6 of 8 errors)
- **Problem:** W3C reports 5 errors for `defer` and 1 for `async` on `<script type="text/rocketlazyloadscript">` — a non-JavaScript MIME type cannot carry those attributes.
- **Solution:**
  - In WP Rocket → File Optimization, disable 'Delay JavaScript execution' for these snippets, or
  - Exclude the affected scripts (cookieyes, the Google tag) from delay so they emit as real `type="text/javascript"` with `defer`.
  - Alternatively move the attributes into the `data-rocket-` namespace the plugin rewrites at runtime.

2J. Give every button an accessible name `Accessibility`

- **Impact:** WCAG 4.1.2 Name/Role/Value; Lighthouse accessibility score
- **Problem:** Lighthouse `button-name` scores 0.00 with weight 10 on mobile, yet axe-core reported 0 violations — the unnamed buttons are likely inside the cookie-consent / iframe-manager UI that axe did not exercise.
- **Solution:**
  - Add visible text or an `aria-label` to each `<button>`:

```
<button type="button" class="cookie-accept" aria-label="Nõustu
kõigi küpsistega">OK</button>
```

- For icon-only buttons use a visually hidden span:

```
<button type="button"><span class="sr-only">Sulge</span><svg aria-hidden="true">…</svg></button>
```

- Re-run axe after interacting with the consent banner so the hidden UI is actually scanned.

2K. Prioritise the LCP element instead of lazy-loading it `Performance`

- **Impact:** LCP (2.6 s mobile)
- **Problem:** The Optimized-Web checklist fails "Hero image eagerly loaded": the first image (`…6c041c12….png` , 251×134) carries `loading="lazy"` and no `fetchpriority` , while mobile LCP is 2.6 s.
- **Solution:**
  - Identify the real LCP element in DevTools → Performance → Timings (the checklist heuristic is inferred and may point at the wrong node).
  - If it is an image, load it eagerly and hint it:

```
<img src="…" width="251" height="134" alt="…" fetchpriority="high" decoding="async">
```

  - If the LCP element is the `h1.hero-simple__title` , preload the font it uses and avoid `font-display: block` .

2L. Remove render-blocking CSS and trim unused rules `Performance`

- **Impact:** FCP, LCP, Speed Index (2.85 s mobile)
- **Problem:** PSI's `render-blocking-insight` estimates 720 ms of savings and `unused-css-rules` 32 KiB; the page loads 2 external stylesheets plus 3 inline blocks (9.6 KB).
- **Solution:**
  - Inline the critical above-the-fold CSS and defer the rest:

```
<link rel="preload" href="/wp-content/themes/…/style.css" as="style" onload="this.onload=null;this.rel='stylesheet'">
```

  - Enable WP Rocket's "Remove Unused CSS" (Optimize CSS Delivery) and verify the generated used-CSS file covers the privacy-policy template.
  - Re-test with Lighthouse to confirm the 720 ms render-blocking estimate drops.

2M. Block /wp-admin/install.php `Security`

- **Impact:** WordPress takeover surface
- **Problem:** The Security basics checklist warns that the installer at `/wp-admin/install.php` is reachable — harmless while installed, but an open takeover path if the database ever becomes unreachable.
- **Solution:**

  ```
  <Files "install.php">
    Require all denied
  </Files>
  ```

  Or via the ZoneOS/WAF rule set. Verify with `curl -I https://estply.com/wp-admin/install.php` returning 403.

2N. Enlarge touch targets to at least 24×24 px (44×44 recommended) `Accessibility`

- **Impact:** WCAG 2.5.8 Target Size; mobile usability
- **Problem:** Lighthouse fails `target-size` with score 0.00 (weight 7) — "Touch targets do not have sufficient size or spacing."
- **Solution:**
  - Increase padding/min-width/min-height on small links and buttons (cookie banner, footer links) to ≥44×44 px.
  - Add spacing between adjacent targets so their hit areas do not overlap.

2O. Cut render-blocking CSS and unused CSS `Performance`

- **Impact:** FCP, Speed Index, LCP
- **Problem:** `render-blocking-insight` estimates 580 ms of savings and `unused-css-rules` estimates 32 KiB unused; mobile Speed Index is 2.59 s versus 479 ms on desktop.
- **Solution:**
  - Inline critical above-the-fold CSS and defer the two external stylesheets.
  - Strip unused rules from the theme CSS (the page is a short text document — most component CSS is unused).
  - WP Rocket: enable "Remove Unused CSS" / Optimize CSS delivery and verify the cookie-banner styling still renders.

2P. Fix the 8 W3C validation errors `Best Practices`

- **Impact:** HTML validity, image delivery correctness
- **Problem:** W3C reports 8 errors: 5× `defer` and 1× `async` on `type="text/rocketlazyloadscript"` data-block scripts, plus 2× `srcset`

candidates with no width descriptor while `sizes` is present.

- **Solution:**
  - WP Rocket lazyload: stop emitting `defer` / `async` on `type="text/rocketlazyloadscript"` tags — move the delay into `data-rocket-defer` instead.
  - Add width descriptors to every `srcset` candidate:

```
<img src="…251x134.png"
     srcset="…-251x134.png 251w, …-502x268.png 502w"
     sizes="(max-width: 600px) 251px, 251px"
     width="251" height="134" alt="">
```

2Q. Cut image weight — 168 KiB of savings identified `Performance`

- **Impact:** LCP, Speed Index, total page weight
- **Problem:** Images account for 673.9 KB of the 745.9 KB page (17 requests), and PSI `image-delivery-insight` estimates 168 KiB of avoidable savings.
- **Solution:**
  - Audit the 19 distinct srcset widths (100–2561 px) — the 100×100 thumbnails are being served at sizes far above their display box.
  - Tighten `sizes` so mobile does not download desktop-width candidates:

```
<img srcset="...400w, ...800w, ...1600w" sizes="(max-width: 600px)
100vw, 50vw">
```

  - Re-encode any remaining oversized WebP at quality 75–80 and verify with `cwebp -q 80`.
  - Consider AVIF for the largest hero/photo assets.

2R. Eliminate forced reflow in theme JavaScript `Performance`

- **Impact:** Main-thread work, Speed Index, INP
- **Problem:** PSI flags `forced-reflow-insight` at high priority and records a 52 ms long task on the document; Speed Index is 3.83 s on mobile.
- **Solution:**
  - Batch DOM reads before writes; never interleave `offsetWidth` / `getBoundingClientRect()` with style mutations in a loop.
  - Use `requestAnimationFrame` for layout reads and `ResizeObserver` instead of scroll/resize handlers that force layout.

- Profile with DevTools Performance → "Layout" track to find the forced synchronous layout call sites in `core.7c21cc2eb9f1534d.js`.

2S. Add Referrer-Policy, Permissions-Policy, COOP and CORP; enable HSTS preload `Security`

- **Impact:** Information leakage, cross-origin isolation, HSTS coverage
- **Problem:** Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are all missing, and HSTS (`max-age=31536000; includeSubDomains`) lacks the `preload` directive.
- **Solution:**

```
Header always set Referrer-Policy "strict-origin-when-cross-origin"
Header always set Permissions-Policy "geolocation=(), microphone=
(), camera=(), payment=()"
Header always set Cross-Origin-Opener-Policy "same-origin"
Header always set Cross-Origin-Resource-Policy "same-site"
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
```

  - Only add `preload` once you are confident every subdomain is HTTPS-only, then submit at hstspreload.org.

2T. Fix the 19 W3C validation errors `Best Practices`

- **Impact:** HTML validity, encoding correctness, image rendering
- **Problem:** W3C reports 19 errors: 12× `script type="text/rocketlazyloadscript"` with `defer`, a `charset` meta placed after the first 1024 bytes, an empty `href` on a `dns-prefetch` link, 2 hidden inputs with `autocomplete`, and 2 `srcset` values without widths.
- **Solution:**
  - Move `<meta charset="utf-8">` to be the very first element inside `<head>` (currently at line 6, after the 1024-byte mark).
  - Remove the empty prefetch link: `<link data-rocket-prefetch href="" rel="dns-prefetch">`.
  - Add widths to the two offending `srcset` candidates:

```
<img srcset=".../img-800.png 800w, .../img-1600.png 1600w"
sizes="...">
```

  - Drop `autocomplete` from the hidden Gravity Forms inputs, and stop WP Rocket from adding `defer` / `async` to `type="text/rocketlazyloadscript"` blocks.

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. Add the remaining low-cost security headers `Security`

- **Impact:** Referrer leakage, feature abuse, cross-origin isolation, tech disclosure
- **Problem:** Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are all missing, HSTS lacks `preload`, the `Server: Apache / ZoneOS` header discloses the stack, and `/wp-admin/install.php` is reachable.
- **Solution:**

```
Header always set Referrer-Policy "strict-origin-when-cross-origin"
Header always set Permissions-Policy "geolocation=(), microphone=
(), camera=()"
Header always set Cross-Origin-Opener-Policy "same-origin"
Header always set Cross-Origin-Resource-Policy "same-origin"
Header always set X-Permitted-Cross-Domain-Policies "none"
Header unset Server
```

  - Add `preload` to HSTS once you are confident in the subdomain set, then submit at hstspreload.org.
  - Block the installer: `<Files "install.php">\n Require all denied\n</Files>`

3B. Add SRI to the jsDelivr script and clear the deprecation warning `Best Practices`

- **Impact:** Supply-chain integrity, Best Practices score (81)
- **Problem:** 1 of 1 cross-origin scripts (`iframemanager.js` from cdn.jsdelivr.net) lacks an `integrity` attribute, and PSI's `deprecations` audit scores 0.00 with 1 warning.
- **Solution:**
  - Pin and hash the third-party script:

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

  - Better: self-host the 3 KB file to remove the third-party origin entirely.
  - Open DevTools console with "Verbose" to identify the deprecated API flagged by Lighthouse and replace it.

3C. Verify the 9 unresolved color-contrast nodes and disambiguate repeated link text `Accessibility`

- **Impact:** WCAG 1.4.3, 2.4.4
- **Problem:** axe-core returned 9 `color-contrast` nodes as "incomplete" (gradients/alpha backgrounds it cannot compute), and the inventory lists vague repeated anchors such as "meist" ×2, "tooted" ×2 and "kontakt" ×2.
- **Solution:**
  - Manually check the 9 flagged nodes with DevTools' contrast picker against the actual composited background; target ≥4.5:1 for body text and ≥3:1 for large text.
  - Disambiguate repeated links with `aria-label` or visually hidden text:

```
<a href="/meist/">meist<span class="sr-only"> — ESTPLY
ettevõttest</span></a>
```

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

- **Impact:** Privacy leakage, cross-origin isolation
- **Problem:** All four headers are missing per the security headers scan (each flagged low/advanced-fail).
- **Solution:**

```
Referrer-Policy: strict-origin-when-cross-origin
Permissions-Policy: camera=(), microphone=(), geolocation=(),
interest-cohort=()
Cross-Origin-Opener-Policy: same-origin
Cross-Origin-Resource-Policy: same-origin
```

3E. Add SRI to the jsdelivr script and trim info disclosure `Security`

- **Impact:** Supply-chain integrity, fingerprinting
- **Problem:** 1 of 1 cross-origin scripts (`iframemanager.js` from cdn.jsdelivr.net) lacks an `integrity` attribute, and the `Server: Apache / ZoneOS` header discloses the stack.
- **Solution:**

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

- Better: self-host the 3 KB file and drop the third-party origin entirely.
- Strip the version from the `Server` header (`ServerTokens Prod` in Apache).

3F. Block the WordPress installer and move the login path `Security`

- **Impact:** Attack surface
- **Problem:** Security basics flags `/wp-admin/install.php` as reachable and the login form served on the default `/wp-login.php` path.
- **Solution:**

```
<Files "install.php">
  Require all denied
</Files>
```

  - Put `/wp-login.php` behind a WAF rule, IP allowlist, or a plugin that moves it to a custom slug.
  - Add `preload` to HSTS once the domain is submitted at hstspreload.org.

3G. Disambiguate repeated vague link text `SEO`

- **Impact:** Anchor text quality, screen-reader navigation
- **Problem:** HTML inventory finds 'here' ×2 plus repeated 'about us', 'products', 'resources', 'contact' anchors, and Lighthouse `link-text` scores 0.00 for 2 links.
- **Solution:**
  - Replace 'here' with descriptive text: 'read our cookie policy', 'view the GDPR disclosures'.
  - Where the visible text must stay short, add `aria-label` :

```
<a href="/en/contact/" aria-label="Contact estply customer
support">contact</a>
```

3H. Fix document cache-control `Performance`

- **Impact:** Repeat-visit TTFB, bfcache
- **Problem:** The document response sends `cache-control: max-age=0` with no ETag, so repeat visits revalidate the full 77.5 KB HTML despite WP Rocket being active.
- **Solution:**
  - Let WP Rocket serve a cached HTML response with a short `max-age` plus `stale-while-revalidate` :

```
Cache-Control: public, max-age=600, stale-while-revalidate=86400
```

- Add an `ETag` so conditional requests can return 304 instead of a full body.

3I. Harden the Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** The CSP is only `frame-ancestors 'self'` — the security-headers tool flags missing `default-src` and missing `object-src 'none'`, so no script or resource restriction is in place. Site signals show no auth, payments, UGC or commerce, so this is a hardening item rather than an urgent one.
- **Solution:** Extend the existing header rather than replacing it:

  ```
  Content-Security-Policy: default-src 'self'; object-src 'none';
  base-uri 'self'; frame-ancestors 'self'
  ```

  If you later add a login or user content, move to a nonce + `'strict-dynamic'` policy instead of a flat allowlist.

3J. Add Referrer-Policy, Permissions-Policy, COOP and CORP; enable HSTS preload `Security`

- **Impact:** Privacy leakage, cross-origin isolation, HSTS coverage
- **Problem:** Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are all missing, and HSTS (`max-age=31536000; includeSubDomains`) lacks the `preload` directive.
- **Solution:**

  ```
  Header always set Referrer-Policy "strict-origin-when-cross-origin"
  Header always set Permissions-Policy "geolocation=(), microphone=
  (), camera=()"
  Header always set Cross-Origin-Opener-Policy "same-origin"
  Header always set Cross-Origin-Resource-Policy "same-origin"
  Header always set Strict-Transport-Security "max-age=63072000;
  includeSubDomains; preload"
  ```

  Submit the domain at hstspreload.org once the header is live.

3K. Fix the 8 W3C validation errors `Best Practices`

- **Impact:** HTML validity, parser robustness
- **Problem:** W3C reports 8 errors: 5× `text/rocketlazyloadscript` with `defer`, 1× with `async`, and 2× `srcset` candidates missing width descriptors (lines 545 and 565).
- **Solution:**
  - WP Rocket's `text/rocketlazyloadscript` is a data block, so `defer` / `async` are invalid on it. Either drop the attributes or switch to a real

deferred script:

```
<script type="text/javascript" defer src="…"></script>
```

- Add width descriptors to the two `srcset` values:

```
<img src="…6c041c12….png" srcset="…6c041c12….png 251w"
sizes="251px" width="251" height="134" alt="" loading="lazy">
```

3L. Set a usable Cache-Control for the HTML document `Performance`

- **Impact:** Repeat-visit TTFB, back/forward cache
- **Problem:** The document response is served with `cache-control: max-age=0` and an `expires` timestamp equal to the request time, so every navigation revalidates the HTML despite WP Rocket being active.
- **Solution:**
  - Let WP Rocket manage the HTML cache and emit a short but non-zero TTL for anonymous visitors:

```
Cache-Control: public, max-age=600, stale-while-revalidate=86400
```

  - Keep `max-age=0, must-revalidate` only for logged-in or cart-bearing sessions.

3M. Serve the two PNGs as WebP/AVIF `Performance`

- **Impact:** Image bytes, LCP
- **Problem:** The HTML inventory shows 2 raster images, both PNG, with 0 WebP — the only remaining modern-format gap on an otherwise lean 73.8 KB page.
- **Solution:**
  - Convert both uploads to WebP and serve via `<picture>` with a PNG fallback, or enable WP Rocket's / your image CDN's automatic WebP conversion.
  - Keep the existing `width` / `height` and `loading="lazy"` attributes intact.

3N. Add SRI to the jsDelivr script and reduce fingerprinting `Security`

- **Impact:** Supply-chain integrity, information disclosure
- **Problem:** 1 of 1 cross-origin scripts lacks an `integrity` attribute (`cdn.jsdelivr.net/…/iframemanager.js`), and the response advertises `Server: Apache / ZoneOS` while the login form sits on the default `/wp-login.php` .
- **Solution:**

- Pin the CDN asset with a hash:

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

- Strip or genericise the `Server` header at the proxy.
- Move the login form behind a non-default path or add IP/2FA gating in front of `/wp-login.php`.

3O. Disambiguate vague link text and type the JSON-LD block `SEO`

- **Impact:** Anchor-text relevance, rich-result eligibility
- **Problem:** The HTML inventory flags repeated vague anchors (`siin` ×2, `meist` ×2, `tooted` ×2, `kontakt` ×2) and PSI's `structuredData` audit fails while the single JSON-LD block reports type `Unknown`.
- **Solution:**
  - Replace bare `siin` / `meist` anchors with descriptive text, e.g. `siin` → `Estply privaatsuspoliitika kontaktandmed`.
  - Give the JSON-LD block an explicit `@type` (e.g. `WebPage` or `PrivacyPolicy`) and validate it in the Rich Results Test.

3P. Add Referrer-Policy, Permissions-Policy, COOP/CORP and HSTS preload `Security`

- **Impact:** Privacy leakage, cross-origin isolation, HSTS coverage
- **Problem:** Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are all missing, and HSTS (`max-age=31536000; includeSubDomains`) lacks the `preload` directive.
- **Solution:**

```
Header always set Referrer-Policy "strict-origin-when-cross-origin"
Header always set Permissions-Policy "geolocation=(), microphone=
(), camera=()"
Header always set Cross-Origin-Opener-Policy "same-origin"
Header always set Cross-Origin-Resource-Policy "same-origin"
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
```

  Submit the domain at hstspreload.org once the header is live.

3Q. Add SRI to the jsDelivr script and trim WordPress exposure `Security`

- **Impact:** Supply-chain integrity, takeover surface

- **Problem:** 1 of 1 cross-origin scripts (cdn.jsdelivr.net iframemanager) lacks an `integrity` attribute; `/wp-admin/install.php` is reachable and login is on the default `/wp-login.php` path; the `Server: Apache / ZoneOS` header discloses the stack.
- **Solution:**
  - Add `integrity="sha384-…" crossorigin="anonymous"` to the jsDelivr `<script>`, or self-host the file.
  - Block the installer in `.htaccess`: `RewriteRule ^wp-admin/install\.php$ - [F,L]`.
  - Move login behind a custom path or IP allowlist and add rate limiting.
  - Strip the `Server` header: `Header unset Server`.

3R. Raise document cache lifetime `Performance`

- **Impact:** Repeat-visit TTFB, cache hit rate
- **Problem:** The document is served with `cache-control: max-age=0`, forcing revalidation on every navigation despite WP Rocket being active.
- **Solution:**
  - In WP Rocket, enable browser caching for HTML with a short but non-zero TTL (e.g. `max-age=600`) and rely on cache purging on publish.
  - Verify the cookie-banner and consent state are not baked into the cached HTML.

3S. Publish valid structured data `SEO`

- **Impact:** Rich results eligibility, Lighthouse `structuredData`
- **Problem:** Lighthouse `structuredData` fails and the single JSON-LD block parses with type "Unknown" — no recognized schema.org entity.
- **Solution:** Emit a `ContactPage` wrapping an `Organization` with `address`, `telephone` and `email` matching the visible contact details:

```
{"@context":"https://schema.org","@type":"ContactPage","mainEntity"
:
{"@type":"Organization","name":"Estply","telephone":"+372...","addr
ess":{"@type":"PostalAddress","addressCountry":"EE"}}}
```

3T. Add SRI to the jsDelivr script and stop disclosing the server `Best Practices`

- **Impact:** Supply-chain integrity, information disclosure
- **Problem:** 1 of 1 cross-origin scripts (`cdn.jsdelivr.net/.../iframemanager.js`) lacks an `integrity` attribute, and the `Server: Apache / ZoneOS` header discloses the stack.

- **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 iframemanager.js | openssl base64 -A`.
  - Strip the server banner: `Header unset Server` and `ServerTokens Prod`.

3U. Harden the WordPress install surface `Best Practices`

- **Impact:** Attack surface, brute-force exposure
- **Problem:** `/wp-admin/install.php` is reachable and the login form is served on the default `/wp-login.php` path — both flagged as warnings by the security checklist.
- **Solution:**
  - Block the installer in `.htaccess`:

```
<Files "install.php">
  Require all denied
</Files>
```

  - Move the login to a custom path (WPS Hide Login or a rewrite rule) and add rate limiting / 2FA.
  - Note: the Security basics verdict is PASS, but PASS only means the basics are in place — it does **not** mean the site is secure: application logic, authentication, plugins, server configuration and dependencies are not tested here.

3V. Allow browser caching of the HTML document `Performance`

- **Impact:** Repeat-visit TTFB, back/forward navigation
- **Problem:** The document response carries `cache-control: max-age=0` despite WP Rocket being active, so every repeat visit re-downloads the 174 KB HTML.
- **Solution:**
  - Let WP Rocket manage the document cache and emit a short but non-zero TTL, e.g. `Cache-Control: public, max-age=600, stale-while-revalidate=60`.
  - Keep `Vary: Accept-Encoding` (already present) and add an `ETag` so conditional requests return 304.

3W. Fix the unused preload for rings.svg `Best Practices`

- **Impact:** Console noise, wasted preload bandwidth
- **Problem:** The browser console warns that the preload for `/wp-content/themes/wordpress-base/inc/theme/img/rings.svg` is unused because the request credentials mode does not match.
- **Solution:**
  - Add `crossorigin` to the preload link so it matches the consuming request:

```
<link rel="preload" href="/wp-content/themes/wordpress-base/inc/theme/img/rings.svg" as="image" type="image/svg+xml" crossorigin>
```

  - Or drop the preload entirely if the SVG is not on the critical path.

3X. Disambiguate repeated link text `SEO`

- **Impact:** Anchor-text quality, screen-reader link lists
- **Problem:** The page repeats vague anchors — "kontakt" ×3, "meist" ×2, "tooted" ×2, "info@estply.com" ×2 — across 73 anchors.
- **Solution:**
  - Make anchor text describe the destination, e.g. "Kontakt — Estply müügiosakond" instead of a bare "kontakt".
  - Where the visible text must stay short, add `aria-label` with the full destination context.