Audit Report: Avaleht - estply

Website: https://estply.com/
Date: 28.10.2026
Audit Coverage: 100% — all sources returned data
Confidence: low

Pages Audited (5 of 5):

- https://estply.com/
- https://estply.com/en/2026/07/06/vineeri-kvaliteediklassid-selgeks
- https://estply.com/en/kupsiste-poliitika
- https://estply.com/kupsiste-poliitika
- https://estply.com/kontakt

Summary of results

Overall Score: 66 / 100
Status: Needs Improvement

Site overall 66 is the mean of 5 pages. Scores range 64 (https://estply.com/) → 70 (https://estply.com/kontakt). Weakest page: Mobile performance is the dominant drag: PSI mobile 68 vs desktop 94, with LCP 6.3 s (vs 1.2 s desktop) and Speed Index 5.71 s, driven almost entirely by a single 1.94 MB 3200×1800 hero WebP — images account for 2.70 MB of the 2.81 MB page weight. Accessibility is solid (axe 0 violations, PSI 91) but PSI still flags `button-name` and `target-size` failures, and 17 contrast nodes were left undetermined. Security basics FAILED (HSTS, X-Frame-Options, X-Content-Type-Options, CSP all missing), which caps the score below 90; the site does expose an upload surface, so CSP is treated as P1. W3C reports 9 errors (mostly WP Rocket `text/rocketlazyloadscript` + `defer` and malformed `srcset` width descriptors) and the document is served with `cache-control: max-age=0`, both secondary drags. The mobile/desktop gap (26 PSI points, 5.1 s LCP delta) is large enough to lower confidence in a single clean score.

Per-page scores

 Needs Improvement · https://estply.com/

Score	Performance	Accessibility	Best Practices	SEO	Security
64	68	91	100	92	FAILED

□ **Needs Improvement** · https://estply.com/en/2026/07/06/vineeri-kvaliteediklassid-selgeks

Score	Performance	Accessibility	Best Practices	SEO	Security
64	67	91	100	100	FAILED

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

Score	Performance	Accessibility	Best Practices	SEO	Security
66	76	91	100	85	FAILED

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

Score	Performance	Accessibility	Best Practices	SEO	Security
68	79	91	100	92	FAILED

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

Score	Performance	Accessibility	Best Practices	SEO	Security
70	76	92	100	92	FAILED

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/	**68** / 94	**6.30 s** / 1.24 s	**0.006** / 0.001
https://estply.com/en/2026/07/06/vineeri-kvaliteediklassid-selgeks	**67** / 94	**3.72 s** / 805 ms	**0.481** / 0.140
https://estply.com/en/kupsiste-poliitika	**76** / 98	**4.49 s** / 970 ms	0.000 / **0.001**
https://estply.com/kupsiste-poliitika	**79** / 98	**4.40 s** / 967 ms	0.000 / **0.001**

URL	Performance (M / D)	LCP (M / D)	CLS (M / D)
https://estply.com/kontakt	**76** / 95	**4.19 s** / 864 ms	**0.009** / 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. Right-size and re-encode the 1.94 MB hero image `Performance`

- **Impact:** LCP, Speed Index, total page weight
- **Problem:** The LCP image (kaasik-kaupo-kalda-3200x1800.webp) is 1.94 MB and images total 2.70 MB of the 2.81 MB page; mobile LCP is 6.3 s vs 1.2 s desktop, and PSI's image-delivery insight estimates 544 KiB savings.
- **Solution:**
  - Serve a mobile-appropriate source instead of the 3200×1800 original; cap the hero at ~1600 px wide and let `srcset` / `sizes` pick the rest.
  - Re-encode to AVIF with WebP fallback and target <200 KB for the hero.
  - Keep `fetchpriority="high"` and add a preload for the chosen hero source:

```
<link rel="preload" as="image" href="/hero-1200.avif"
      imagesrcset="/hero-800.avif 800w, /hero-1200.avif 1200w,
/hero-1600.avif 1600w"
      imagesizes="100vw" fetchpriority="high">
```

  - Verify the RocketCDN origin isn't bypassing the resized variants.

1B. Add the missing baseline security headers `Security`

- **Impact:** Transport security, clickjacking, MIME sniffing
- **Problem:** Security basics verdict is FAILED: Strict-Transport-Security, X-Frame-Options (or CSP frame-ancestors) and X-Content-Type-Options are all missing from the response.
- **Solution:** Send from the origin/Apache vhost:

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
Header always set X-Content-Type-Options "nosniff"
Header always set X-Frame-Options "SAMEORIGIN"
Header always set Referrer-Policy "strict-origin-when-cross-origin"
```

  Then submit the domain to hstspreload.org once the max-age is stable.

1C. Deploy a nonce-based Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth

- **Problem:** CSP is missing entirely, and the site exposes an upload surface (anchor href contains "upload"), so injected script could persist into user-visible content.
- **Solution:** Use nonce + `'strict-dynamic'` rather than a flat allowlist:

```
Content-Security-Policy: default-src 'self'; script-src 'nonce-
{random}' 'strict-dynamic'; object-src 'none'; base-uri 'none';
frame-ancestors 'self'
```

  - Roll out in Report-Only first to catch WP Rocket / jsDelivr / Google Fonts breakage.
  - `frame-ancestors 'self'` also closes the clickjacking gap.

1D. Eliminate the 0.481 footer layout shift `Performance`

- **Impact:** CLS, mobile Performance score
- **Problem:** CLS is 0.481 on mobile (desktop 0.140), with the shift attributed to div#page > div.main > div.main__footer > footer.footer — well past the 0.25 heavy-penalty threshold.
- **Solution:** Reserve space for the footer before it renders:
  - Give the footer container a `min-height` matching its rendered height, or render it server-side instead of injecting it late.
  - If the shift comes from a lazily-loaded element inside the footer (cookie banner, iframe manager, badge), wrap it in a fixed-size placeholder.
  - Audit any `position: sticky/fixed` footer element that is inserted after first paint. Verify with Lighthouse's Layout Shift Regions overlay.

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

- **Impact:** LCP (3.7 s mobile)
- **Problem:** The checklist flags the hero image (uploads/2026/08/6c041c12…png) as `loading="lazy"` with no `fetchpriority`, which defers the likely LCP element; mobile LCP is 3.7 s vs 0.8 s desktop.
- **Solution:**

```
<img src="…6c041c12….png"
     alt="…"
     width="251" height="134"
     loading="eager"
     fetchpriority="high"
     decoding="async">
```

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

1F. Add HSTS, X-Content-Type-Options and X-Frame-Options `Security`

- **Impact:** Transport security, clickjacking, MIME sniffing
- **Problem:** Security basics verdict is FAILED: Strict-Transport-Security, X-Frame-Options/frame-ancestors and X-Content-Type-Options are all missing from the response.
- **Solution:**

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
Header always set X-Content-Type-Options "nosniff"
Header always set X-Frame-Options "SAMEORIGIN"
```

  Prefer `Content-Security-Policy: frame-ancestors 'self'` once CSP is in place; keep X-Frame-Options as the fallback for older browsers.

1G. Add a real heading structure (currently zero headings) `Accessibility`

- **Impact:** Screen-reader navigation, document outline, SEO
- **Problem:** The heading outline reports h1 ×0, h2 ×0, h3 ×0, h4 ×0, h5 ×0, h6 ×0, and axe flags `page-has-heading-one` (moderate) on `html`.
- **Solution:**
  - Add exactly one `<h1>` containing the article title ("Vineeri kvaliteediklassid selgeks").
  - Mark up section titles as `<h2>` / `<h3>` in document order.
  - Do not use styled `<div>` / `<p>` elements in place of headings.
  - Re-run axe and confirm the outline is sequential with no skipped levels.

1H. Stop lazy-loading the hero image and prioritise it for LCP `Performance`

- **Impact:** LCP (4.5 s mobile), Speed Index, FCP
- **Problem:** The optimized-web checklist fails 'Hero image eagerly loaded': the first image (`6c041c12…png` , 251×134) has `loading="lazy"` and no `fetchpriority` , and mobile LCP is 4.5 s vs 1.0 s on desktop.
- **Solution:**
  - Remove `loading="lazy"` from the hero and add `fetchpriority="high"` :

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

  - Preload it in `<head>` so it is discovered before the CSS/JS chain:

```
<link rel="preload" as="image" href="/wp-
content/uploads/2026/08/6c041c12….png" fetchpriority="high">
```

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

1I. Cut render-blocking CSS and font work to fix FCP `Performance`

- **Impact:** FCP (3.63 s), Speed Index (3.63 s), LCP
- **Problem:** Mobile FCP is 3.63 s while TTFB is only 24 ms, so the delay is in the render path: 4 external stylesheets plus 3 inline blocks (9.6 KB), and 4 Google Fonts files totalling 96.2 KB from fonts.gstatic.com. PSI flags `document-latency-insight` (580 ms est. savings) and `font-display-insight` (40 ms).
- **Solution:**
  - Self-host Open Sans as WOFF2 and preload only the weights actually used; drop the `fonts.googleapis.com` / `fonts.gstatic.com` origins (2 + 2 requests).
  - Add `font-display: swap` (or `optional`) to every `@font-face`.
  - Inline the critical above-the-fold CSS and defer the rest:

```
<link rel="preload" href="/fonts/opensans-var.woff2" as="font"
type="font/woff2" crossorigin>
```

  - Reduce the 580 ms document-latency chain by removing unused stylesheet requests and trimming the 9.6 KB of inline CSS.

1J. Add the three missing baseline security headers `Security`

- **Impact:** Transport security, clickjacking, MIME sniffing
- **Problem:** Security basics verdict is FAILED: Strict-Transport-Security, X-Content-Type-Options and clickjacking protection (X-Frame-Options / CSP `frame-ancestors`) are all reported missing on the response.
- **Solution:** Send from the origin (Apache shown, matching the `Server: Apache / ZoneOS` header):

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
Header always set X-Content-Type-Options "nosniff"
Header always set X-Frame-Options "SAMEORIGIN"
Header always set Referrer-Policy "strict-origin-when-cross-origin"
```

  Prefer `Content-Security-Policy: frame-ancestors 'self'` over X-Frame-Options once CSP is in place, and submit the domain to hstspreload.org after the HSTS max-age is verified.

1K. Stop returning HTTP 429 and cache the document `Performance`

- **Impact:** TTFB, LCP, availability, crawlability
- **Problem:** The audit fetch of /kupsiste-poliitika returned status 429 with `retry-after: 10` and a 337 B error body; the response has no `cache-control` header and the checklist finds no page-cache plugin or CDN edge cache.
- **Solution:**
  - Whitelist normal page requests in the rate limiter (the 429 fired on a single first visit) and raise the burst threshold.
  - Enable a page cache / CDN edge cache for anonymous GETs and send explicit caching headers:

```
Cache-Control: public, max-age=600, s-maxage=86400, stale-while-revalidate=60
```

  - Ensure the cookie-policy page is served from cache, not regenerated per request.

1L. Cut mobile LCP (4.4 s) and FCP (3.05 s) `Performance`

- **Impact:** LCP, FCP, Speed Index, document latency
- **Problem:** PSI mobile LCP is 4.4 s vs 1.0 s on desktop and FCP 3.05 s vs 807 ms; `document-latency-insight` estimates 980 ms of savings and `font-display-insight` 110 ms.
- **Solution:**
  - Reduce document latency (see the caching fix) — it is the largest single contributor at ~980 ms.
  - Add `font-display: swap` and preload the primary font:

```
<link rel="preload" href="/fonts/main.woff2" as="font" type="font/woff2" crossorigin>
```

```
@font-face { font-family: Main; src: url(/fonts/main.woff2) format('woff2'); font-display: swap; }
```

  - Preconnect to the origin serving the LCP image and serve it as WebP/AVIF with explicit dimensions.
  - Flatten the critical request chain flagged by `network-dependency-tree-insight`.

1M. Make the LCP image eager and high-priority `Performance`

- **Impact:** LCP, Speed Index, FCP
- **Problem:** Mobile LCP is 4.2 s (desktop 0.9 s) and the optimized-web checklist flags the hero image as `loading="lazy"` with no `fetchpriority`. The heuristic picked a 100×100 thumbnail, so confirm the real LCP element in DevTools first.
- **Solution:**
  - Identify the true LCP element in DevTools → Performance → LCP.
  - Remove `loading="lazy"` from it and add `fetchpriority="high"`:

```
<img src="/hero.webp" alt="…" width="1600" height="900"
     fetchpriority="high" decoding="async">
```

  - Preload it so it is discovered in the initial HTML:

```
<link rel="preload" as="image" href="/hero.webp"
fetchpriority="high">
```

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

1N. Ship a nonce-based Content-Security-Policy `Security`

- **Impact:** XSS defense-in-depth on a page with a user-input form
- **Problem:** CSP is missing entirely, and the page hosts a contact form with a `<textarea>` (hasUserContent = yes), so injected script would persist into stored content.
- **Solution:** Use nonce + `'strict-dynamic'` rather than a flat allowlist:

```
Content-Security-Policy: default-src 'self'; script-src 'nonce-
{RANDOM}' 'strict-dynamic' https:; style-src 'self' 'unsafe-inline'
https://fonts.googleapis.com; font-src https://fonts.gstatic.com;
img-src 'self' data:; object-src 'none'; base-uri 'none'; frame-
ancestors 'self'; form-action 'self'
```

  - Roll out in `Content-Security-Policy-Report-Only` first and watch the reports.
  - Note the page loads `cdn.jsdelivr.net` (iframemanager) — allowlist it explicitly or self-host it.

1O. Name the unnamed buttons and enlarge touch targets `Accessibility`

- **Impact:** WCAG 4.1.2 name/role/value, 2.5.8 target size
- **Problem:** PSI fails `button-name` (score 0, weight 10) and `target-size` (score 0, weight 7) on mobile; axe found 0 violations, so these are likely inside the cookie-consent banner or a state axe did not capture.

- **Solution:**
  - Give every icon-only button an accessible name:

```
<button type="button" aria-label="Sulge küpsiste seaded">
  <svg aria-hidden="true">…</svg>
</button>
```

  - Ensure interactive targets are at least 44×44 px with ≥8 px spacing:

```
.icon-btn { min-width: 44px; min-height: 44px; }
```

  - Re-run axe after accepting/dismissing the cookie banner to cover that state.

Priority 2: Important

Essential for compliance, user reach, and search visibility.

2A. Fix unnamed buttons and undersized touch targets `Accessibility`

- **Impact:** WCAG 4.1.2 name/role/value, 2.5.8 target size
- **Problem:** PSI mobile accessibility is 91 with `button-name` (score 0.00, weight 10) and `target-size` (score 0.00, weight 7) failing, even though axe-core reported 0 violations.
- **Solution:**
  - Give every icon-only button an accessible name: `<button aria-label="Ava menüü">…</button>`.
  - Ensure interactive targets are at least 24×24 px (44×44 recommended) with adequate spacing:

```
.icon-btn { min-width: 44px; min-height: 44px; display: inline-
flex; align-items: center; justify-content: center; }
```

  - Re-run axe after the fix — the two tools disagree, so verify in-browser.

2B. Add a meta description `SEO`

- **Impact:** Search snippet CTR, PSI SEO score
- **Problem:** HTML inventory shows `Description: not set` and PSI's `meta-description` audit scores 0.00; SEO is 92 on both strategies.
- **Solution:** Add a unique 120–155 character description per page, localized for the `et` and `en` hreflang variants:

```
<meta name="description" content="ESTPLY toodab kvaliteetset
kasevineeri Jõgevamaal – kaasaegne vineeritehas ja usaldusväärne
```

```
tarnija.">
```

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

- **Impact:** TTFB on repeat visits, LCP
- **Problem:** The document response carries `cache-control: max-age=0` with an `expires` equal to the request time, so every navigation revalidates despite WP Rocket being active.
- **Solution:**
  - Set a short but non-zero document TTL with stale-while-revalidate:

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

  - Keep the WP Rocket page cache warm for the canonical URL.
  - Confirm logged-in / dynamic regions bypass the cache rather than forcing max-age=0 globally.

2D. Fix the 9 W3C validation errors `Best Practices`

- **Impact:** HTML validity, lazy-loading behavior, responsive image selection
- **Problem:** W3C reports 9 errors: 5× `text/rocketlazyloadscript` scripts carrying `defer`, an empty `href` on a dns-prefetch `<link>`, a `sizes="auto"` image without `loading="lazy"`, and 2 `srcset` candidates missing width descriptors.
- **Solution:**
  - Drop `defer` from the RocketLazyLoad data-block scripts, or switch them to `type="text/javascript"` with `defer`.
  - Remove the empty `<link data-rocket-prefetch href="" rel="dns-prefetch">` or give it a real hostname.
  - Add `loading="lazy"` to the `sizes="auto"` image.
  - Add width descriptors to the two PNG `srcset` entries:

```
<img src="...251x134.png" srcset="...251x134.png 251w,
...502x268.png 502w" sizes="251px" loading="lazy" width="251"
height="134" alt="">
```

2E. Give icon buttons an accessible name `Accessibility`

- **Impact:** WCAG 4.1.2, Lighthouse accessibility score
- **Problem:** Lighthouse's `button-name` audit scores 0.00 (weight 10) — buttons on the page have no accessible name.
- **Solution:**

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

Use visible text where possible; `aria-label` only when the control is icon-only.

2F. Enlarge touch targets to at least 44×44 px `Accessibility`

- **Impact:** WCAG 2.5.8, mobile usability
- **Problem:** Lighthouse's `target-size` audit scores 0.00 (weight 7) — touch targets lack sufficient size or spacing.
- **Solution:**
  - Ensure every interactive element is ≥44×44 CSS px (or ≥24×24 with adequate spacing per WCAG 2.5.8 AA).
  - Add padding rather than increasing font size:

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

2G. Stop sending max-age=0 on the HTML document `Performance`

- **Impact:** Repeat-visit TTFB, caching
- **Problem:** The document response carries `cache-control: max-age=0` with an `expires` equal to the request time, so every navigation revalidates despite WP Rocket being active.
- **Solution:**
  - In WP Rocket, enable browser caching for HTML with a short TTL (e.g. 600 s) and let the plugin purge on publish.
  - Or set at origin:

```
Header set Cache-Control "public, max-age=600, s-maxage=3600, stale-while-revalidate=60"
```

Keep `no-cache` only for logged-in or cart pages.

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

- **Impact:** W3C validity, responsive image selection
- **Problem:** W3C reports 2 errors: `srcset` candidates on the two uploads PNGs specify no width while a `sizes` attribute is present.
- **Solution:**

```
<img src="…6c041c12….png"
     srcset="…6c041c12….png 251w"
     sizes="(max-width: 600px) 100vw, 251px"
     width="251" height="134" alt="">
```

Every candidate in `srcset` must carry a `w` descriptor when `sizes` is used.

2I. Block the WordPress installer and move the login path `Security`

- **Impact:** Brute-force and takeover surface
- **Problem:** Security basics reports `/wp-admin/install.php` is reachable and the login form is served on the default `/wp-login.php` path (both warns).
- **Solution:**

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

  - Add rate limiting / 2FA on the login endpoint.
  - Optionally front the login with a plugin that renames the path, and keep xmlrpc.php blocked (already passing).

2J. Give buttons accessible names and enlarge touch targets `Accessibility`

- **Impact:** WCAG 4.1.2 name/role/value, 2.5.8 target size
- **Problem:** Lighthouse mobile accessibility is 91 and fails `button-name` (score 0.00, weight 10) and `target-size` (score 0.00, weight 7); axe-core reported 0 violations, so these were missed by its rule set and need manual confirmation.
- **Solution:**
  - Add a text or `aria-label` to every icon-only button:

```
<button type="button" aria-label="Close cookie banner"><svg aria-
hidden="true">…</svg></button>
```

  - Ensure interactive targets are at least 44×44 px with adequate spacing (WCAG 2.5.8), especially in the nav and cookie banner.
  - Re-run axe plus a keyboard-only pass, since axe left 7 color-contrast nodes undetermined.

2K. Fix duplicate h1 and vague link text `SEO`

- **Impact:** Document outline, Lighthouse `link-text`, screen-reader navigation
- **Problem:** The heading outline contains two `<h1>` elements ('Küpsiste poliitika' and 'Privacy Policy'), and the inventory flags vague repeated link text including

'here' ×2 plus 'about us', 'products', 'contact' repeated.

- **Solution:**
  - Keep exactly one `<h1>` per page; demote the second to `<h2>` (or remove it if it is a duplicated language label).
  - Replace 'here' links with descriptive text or add an `aria-label`:

```
<a href="/en/privacy-policy" aria-label="Read the full privacy policy">here</a>
```

2L. Harden the WordPress install paths `Security`

- **Impact:** Brute-force and takeover surface
- **Problem:** Security basics reports `/wp-admin/install.php` is reachable and the login form is served on the default `/wp-login.php` path (both 'warn').
- **Solution:**
  - Block the installer in `.htaccess`:

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

  - Move or front the login with a plugin/WAF rule, add rate limiting and 2FA, and keep `xmlrpc.php` blocked (already passing).

2M. Let the document be cached `Performance`

- **Impact:** Repeat-visit TTFB, LCP
- **Problem:** The document response sends `cache-control: max-age=0` with an `expires` equal to the request time, so every navigation revalidates the 78.4 KB HTML despite WP Rocket being active.
- **Solution:**
  - Raise the HTML cache lifetime and let WP Rocket serve the cached copy:

```
Cache-Control: public, max-age=600, s-maxage=86400, stale-while-revalidate=60
```

  - Keep `max-age=0` only for logged-in or personalised responses.

2N. Add baseline security headers (HSTS, nosniff, frame protection) `Security`

- **Impact:** Transport security, MIME sniffing, clickjacking

- **Problem:** Security basics FAILED: Strict-Transport-Security, X-Content-Type-Options and X-Frame-Options / CSP frame-ancestors are all missing on the HTTPS response.
- **Solution:** Send from the origin (Apache shown):

```
Header always set Strict-Transport-Security "max-age=63072000;
includeSubDomains; preload"
Header always set X-Content-Type-Options "nosniff"
Header always set X-Frame-Options "SAMEORIGIN"
```

  Then verify preload eligibility at hstspreload.org.

2O. Add meta description and structured data `SEO`

- **Impact:** Search snippet quality, rich results
- **Problem:** PSI SEO is 92 with `meta-description` and `structured-data` failing; the HTML inventory confirms no description, no Open Graph, no JSON-LD.
- **Solution:**

```
<meta name="description" content="Estply küpsiste poliitika –
milliseid küpsiseid me kasutame ja kuidas neid hallata.">
<script type="application/ld+json">
{"@context":"https://schema.org","@type":"WebPage","name":"Küpsiste
poliitika","inLanguage":"et"}
</script>
```

2P. Fix the 7 W3C validation errors `Best Practices`

- **Impact:** HTML validity, lazy-load reliability, responsive image selection
- **Problem:** W3C reports 7 errors: 5× `script type="text/rocketlazyloadscript"` carrying a `defer` attribute, and 2× `srcset` candidates with no width descriptor while `sizes` is present.
- **Solution:**
  - RocketLazyLoad rewrites `type` to a data block, which makes `defer` invalid. Exclude these scripts from lazy-load rewriting or drop `defer` on them.
  - Add width descriptors to every `srcset` candidate:

```
<img src="...-1024.png"
     srcset="...-480.png 480w, ...-1024.png 1024w"
     sizes="(max-width: 600px) 100vw, 1024px"
     width="1024" height="512" alt="">
```

2Q. Cut the 611 KB image payload `Performance`

- **Impact:** LCP, Speed Index, total page weight

- **Problem:** Images are 611.2 KB of the 732 KB total across 16 requests, and mobile Speed Index is 6.25 s.
- **Solution:**
  - Serve AVIF with WebP fallback for photographic content.
  - Right-size: the inventory shows 20 distinct srcset widths (100–2561 px) — trim the ladder to the 3–4 widths actually used.
  - Re-check the 100×100 thumbnails; if they render larger, the srcset `sizes` attribute is wrong.

2R. Stop sending cache-control: max-age=0 on the document `Performance`

- **Impact:** Repeat-visit TTFB, LCP
- **Problem:** The document response carries `cache-control: max-age=0` with `expires` equal to the request time, so nothing is cached by the browser despite WP Rocket being active.
- **Solution:**

```
# HTML: short browser TTL, long edge TTL
Cache-Control: public, max-age=600, s-maxage=86400, stale-while-
revalidate=86400

# Static assets: immutable
Cache-Control: public, max-age=31536000, immutable
```

  - Confirm WP Rocket's browser-cache setting is not set to "Disable".

2S. Self-host fonts and fix font-display `Performance`

- **Impact:** FCP, LCP, render-blocking CSS
- **Problem:** `font-display-insight` is a high-priority PSI finding (est. 80 ms) and 3 font files (47.2 KB) load from fonts.gstatic.com via a render-blocking Google Fonts CSS request.
- **Solution:**
  - Self-host Open Sans (400/600/700) as woff2 with `font-display: swap`.
  - Preload only the weight used above the fold:

```
<link rel="preload" as="font" type="font/woff2"
      href="/fonts/opensans-400.woff2" crossorigin>
```

  - Drop the `fonts.googleapis.com` / `fonts.gstatic.com` origins entirely.

2T. Fix the 18 W3C validation errors `Best Practices`

- **Impact:** Markup validity, parser robustness
- **Problem:** W3C reports 18 errors: 12× `script type="text/rocketlazyloadscript"` carrying `defer`, a `charset` meta after the first 1024 bytes, an empty `href` on a dns-prefetch link, `srcset` candidates without width descriptors, and `autocomplete` on hidden inputs.
- **Solution:**
  - Move `<meta charset="utf-8">` to the very first bytes of `<head>`.
  - Remove the empty `<link data-rocket-prefetch href="" rel="dns-prefetch">` or give it a real host.
  - Add width descriptors to every `srcset` candidate when `sizes` is present:

```
<img srcset="/a-400.webp 400w, /a-800.webp 800w" sizes="(max-width:
800px) 100vw, 800px" …>
```

  - Drop `autocomplete` from `<input type="hidden">` fields.
  - The 12 `text/rocketlazyloadscript` + `defer` errors come from WP Rocket's Delay JS — exclude those scripts from delay execution or update WP Rocket.

2U. Verify the 109 undetermined contrast nodes `Accessibility`

- **Impact:** WCAG 1.4.3 contrast
- **Problem:** axe could not determine contrast for 109 nodes (alpha, gradients, or overlapping backgrounds), so the 0-violation result is not proof of compliance.
- **Solution:**
  - Run axe DevTools in-browser and inspect the "needs review" contrast list.
  - Check text over images and gradient backgrounds manually with a contrast picker (target ≥4.5:1 for body, ≥3:1 for large text).

2V. Add the remaining hardening headers and SRI `Security`

- **Impact:** Referrer leakage, feature abuse, cross-origin isolation, supply chain
- **Problem:** Referrer-Policy, Permissions-Policy, COOP, CORP and X-Permitted-Cross-Domain-Policies are all missing; 1 of 1 cross-origin scripts lacks an `integrity` attribute; the `Server: Apache / ZoneOS` header discloses the stack.
- **Solution:**

```
Header always set Referrer-Policy "strict-origin-when-cross-origin"
Header always set Permissions-Policy "camera=(), microphone=(),
geolocation=()"
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 SRI to the jsDelivr script:

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

Priority 3: Best Practice

Recommended for long-term maintainability.

3A. Self-host fonts and set font-display `Performance`

- **Impact:** FCP, render-blocking CSS, third-party latency
- **Problem:** PSI flags `font-display-insight` (high, ~50 ms savings) and the runtime shows 3 font files plus a blocking `fonts.googleapis.com` CSS request (47.2 KB woff2 from fonts.gstatic.com).
- **Solution:**
  - Self-host Open Sans subsets as woff2 and preload the one weight used above the fold.
  - Add `font-display: swap` (or `optional`) to every `@font-face`.
  - Drop the unused cyrillic subset if the `et` / `en` locales don't need it.

3B. Manually verify the 17 undetermined contrast nodes `Accessibility`

- **Impact:** WCAG 1.4.3 contrast
- **Problem:** axe-core returned 0 violations but left 17 nodes as "incomplete" for color-contrast, so contrast is unverified rather than confirmed passing.
- **Solution:**
  - Check each flagged node in DevTools against a 4.5:1 (normal text) / 3:1 (large text) threshold.
  - Pay particular attention to text over the hero image and any gradient/alpha backgrounds axe skips.

3C. Tighten remaining hygiene items `Security`

- **Impact:** Information disclosure, WordPress hardening
- **Problem:** Server header discloses `Apache / ZoneOS`, 1 cross-origin script lacks SRI, `/wp-admin/install.php` is reachable, and login sits on the default `/wp-`

`login.php`; Permissions-Policy, COOP and CORP are also absent.

- **Solution:**
  - `Header always unset Server` / `Header always unset X-Powered-By`.
  - Add `integrity` + `crossorigin` to the jsDelivr iframemanager script, or self-host it.
  - Block the installer: `Redirect 404 /wp-admin/install.php`.
  - Move login behind a custom path or IP allowlist.
  - Add `Permissions-Policy: geolocation=(), camera=(), microphone=()` and `Cross-Origin-Opener-Policy: same-origin`.

3D. Add a nonce-based CSP and the remaining low-cost headers `Security`

- **Impact:** XSS defense-in-depth, referrer leakage, feature exposure
- **Problem:** CSP, Referrer-Policy, Permissions-Policy, COOP and CORP are all missing. Site signals show no auth, payments, user content or commerce, so XSS impact is limited — hence P3 rather than P1.
- **Solution:** Deploy CSP with nonces and `'strict-dynamic'` rather than a flat allowlist:

```
Content-Security-Policy: default-src 'self'; script-src 'nonce-
{random}' 'strict-dynamic'; object-src 'none'; base-uri 'none';
frame-ancestors 'self'
Referrer-Policy: strict-origin-when-cross-origin
Permissions-Policy: camera=(), microphone=(), geolocation=()
Cross-Origin-Opener-Policy: same-origin
Cross-Origin-Resource-Policy: same-origin
```

  Note WP Rocket's inline scripts will need nonce injection or hashes.

3E. Clean up WP Rocket's rocketlazyloadscript markup `Best Practices`

- **Impact:** W3C validity
- **Problem:** 5 of the 7 W3C errors are `script type="text/rocketlazyloadscript"` elements carrying a `defer` attribute, which is invalid on a non-JavaScript data block.
- **Solution:**
  - Update WP Rocket to a version that emits valid delay markup, or switch to the native `type="module"` / `defer` combination.
  - Alternatively exclude the affected scripts (cookieyes, iframemanager) from delay-JS execution.

3F. Add SRI to the jsDelivr script and hide the Server header `Security`

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

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

```
ServerSignature Off
ServerTokens Prod
```

3G. Add structured data and disambiguate repeated link text `SEO`

- **Impact:** Rich results, link-purpose clarity
- **Problem:** PSI's modeled `structuredData` audit fails and the JSON-LD block is typed "Unknown"; the inventory also lists vague repeated anchors ("about us" ×2, "products" ×2, "resources" ×2, "contact" ×2).
- **Solution:**
  - Emit a valid `Article` / `BlogPosting` JSON-LD block with headline, datePublished, author and image.
  - Disambiguate repeated nav links with `aria-label` or visually-hidden context, e.g. `<a href="/en/about/">About us<span class="sr-only"> — Estply</span></a>`.

3H. Clear the 7 W3C validation errors `Best Practices`

- **Impact:** Markup correctness, lazy-load reliability
- **Problem:** W3C reports 7 errors: 5× a `script type="text/rocketlazyloadscript"` data block carrying `defer` (first at line 21), and 2× `srcset` candidates missing width descriptors on the two `image__img` elements (lines 574 and 594).
- **Solution:**
  - WP Rocket's `text/rocketlazyloadscript` is a data block, so `defer` is invalid on it — let the plugin rewrite the type at runtime instead of adding `defer` in the markup.
  - Add width descriptors to the `srcset` values, e.g.:

```
<img src="…-251x134.png"
     srcset="…-251x134.png 251w, …-502x268.png 502w"
```

```
      sizes="(max-width: 600px) 100vw, 251px"
      width="251" height="134" alt="">
```

3I. Add a nonce-based CSP (deferred — no auth surface today) `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** Content-Security-Policy is missing and the security checklist marks both 'CSP present' and 'CSP is strict' as failed. Site signals show no auth, payments, user content or e-commerce, so this is P3 rather than P1.
- **Solution:** When a login or user content is added, deploy a nonce + `'strict-dynamic'` policy rather than a flat allowlist:

  ```
  Content-Security-Policy: default-src 'self'; script-src 'nonce-
  {random}' 'strict-dynamic'; object-src 'none'; base-uri 'none';
  frame-ancestors 'self'
  ```

  Also add `Permissions-Policy`, `Cross-Origin-Opener-Policy: same-origin` and `Cross-Origin-Resource-Policy: same-origin`, and add `integrity` + `crossorigin` to the `cdn.jsdelivr.net` iframemanager script (currently 1 of 1 cross-origin scripts lacks SRI).

3J. Serve Brotli instead of gzip `Performance`

- **Impact:** Transfer size, FCP
- **Problem:** The document is compressed with gzip at a 0.273 ratio (78.4 KB decoded → 21.9 KB); Brotli typically saves a further 15–20% on HTML/CSS/JS.
- **Solution:** Enable `brotli` in the server config and keep gzip as fallback:

  ```
  <IfModule mod_brotli.c>
    AddOutputFilterByType BROTLI_COMPRESS text/html text/css
  application/javascript application/json image/svg+xml
  </IfModule>
  ```

3K. Add a nonce-based CSP (defer until auth or user content exists) `Security`

- **Impact:** XSS defense-in-depth
- **Problem:** Content-Security-Policy is missing. Site signals show no auth, payments, user-generated content or e-commerce, so the exploitable surface is minimal today.
- **Solution:** When a login or user content is added, deploy a nonce + `'strict-dynamic'` policy rather than a flat allowlist:

  ```
  Content-Security-Policy: default-src 'self'; script-src 'nonce-
  {random}' 'strict-dynamic'; object-src 'none'; base-uri 'none';
  frame-ancestors 'self'
  ```

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

- **Impact:** Information leakage, feature abuse, origin isolation
- **Problem:** Referrer-Policy, Permissions-Policy, Cross-Origin-Opener-Policy and Cross-Origin-Resource-Policy are all missing; the response also discloses `Server: Apache / ZoneOS`.
- **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"
ServerTokens Prod
```

3M. Re-run the accessibility audit against the real page `Accessibility`

- **Impact:** Audit validity for WCAG findings
- **Problem:** axe-core ran on the 429 error document and reports `html-has-lang` (serious), `landmark-one-main` and `region`, while W3C sees `lang="et"` on the real page — the two results contradict each other.
- **Solution:**
  - Re-run axe-core once the rate limit is lifted and confirm whether `html-has-lang`, `landmark-one-main` and `region` still apply.
  - If they do, add `<main>` and wrap the cookie-policy text in it:

```
<main id="content"><h1>Küpsiste poliitika</h1>…</main>
```

  - Also confirm the `lang="et"` value is correct for the Estonian content (the validator's English heuristic is likely a false positive).

3N. Disambiguate repeated link text and add structured data `SEO`

- **Impact:** Anchor relevance, rich results
- **Problem:** `structuredData` fails and the JSON-LD block parses as type "Unknown"; 9 link texts repeat ("kontakt" ×3, "meist" ×2, "tooted" ×2, etc.).
- **Solution:**
  - Add `Organization` + `ContactPage` JSON-LD with name, url, telephone, address and `contactPoint`.
  - Disambiguate repeated anchors with `aria-label` or surrounding context, e.g. `<a href="/kontakt/" aria-label="Kontakt — Estply AS">kontakt</a>`.

3O. Harden the WordPress install `Best Practices`

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

```
# Block the installer
<Files "install.php">
  Require all denied
</Files>
```

  - Move the login to a custom path (e.g. via a security plugin) and add rate limiting / 2FA.