

How Bear does analytics with CSS

01 Nov, 2023

Bear Blog has a few design constraints for speed, efficiency, and stability. There are many great open-source, privacy-focussed analytics platforms out there, but I wanted to build one native to Bear.

tldr;

One of my constraints for Bear is to not use client-side javascript. This applies to the analytics system as well. Client-side javascript can be tweaked to determine the authenticity of traffic to a page and determine (partially) whether it is bot traffic or not, which is very useful for analytics. The main downside, however, is that most adblockers block analytics scripts. And not just the bad ones, like Google Analytics. Even Fathom and Plausible analytics struggle with logging activity on adblocked browsers.

There's always the option of just parsing server logs, which gives a rough indication of the kinds of traffic accessing the server. Unfortunately all server traffic is generally seen as equal. Technically bots "should" have a user-agent that identifies them as a bot, but few identify that since they're trying to scrape information as a "person" using a browser. In essence, just using server logs for analytics gives a skewed perspective to traffic since a lot of it are search-engine crawlers and scrapers (and now GPT-based parsers).

So instead of using server logs, I trigger a read with CSS. Here's my slightly boutique analytics system.

When a person accesses the website the page is loaded. On each page I have the following CSS:

```
body:hover {
  border-image: url("/hit/{{ post.id }}/?ref={{ request.META.HTTP_REFERER }}");
}
```

The only info I need to actively re-add to this request is the referrer (yes, HTTP_REFERER is [spelt incorrectly](#)).

Now, when a person hovers their cursor over the page (or scrolls on mobile) it triggers `body:hover` which calls the URL for the post hit. I don't think any bots hover and instead just use JS to interact with the page, so I can, with reasonable certainty, assume that this is a human reader.

I then confirm the user-agent isn't a bot (which isn't perfect, but still something). I also extract the browser and platform from the user-agent string.

My second constraint is to not store any identifying information about the reader either in browser cookies, or on the server. In order to do this I use the IP address of the request to determine the country, then hash the IP address along with the date. All subsequent requests to the page are checked for matching IP address + date hashes and duplicates are discarded.

In this way each IP address per day constitutes one "read" of the page. No IP addresses are stored un-hashed and the IP-with-date-hash creates a convenient built-in expiry time.

Here's the code if you're interested:

```
user_agent = httpagentparser.detect(self.request.META.get('HTTP_USER_AGENT', None))
if user_agent.get('bot', False):
    print('Bot traffic')
    return

ip_hash = hashlib.md5(f"{client_ip(self.request)}-{timezone.now().date()}".encode('utf-8'))

country = get_user_location(client_ip(self.request)).get('country_name', '')
device = user_agent.get('platform', {}).get('name', '')
browser = user_agent.get('browser', {}).get('name', '')

referrer = self.request.GET.get('ref', '')
if referrer:
    referrer = urlparse(referrer)
    referrer = '{uri.scheme}://{uri.netloc}/'.format(uri=referrer)

Hit.objects.get_or_create(
    post_id=self.pk,
    ip_address=ip_hash,
    referrer=referrer,
    country=country,
    device=device,
    browser=browser)
```

The only downside to this method is if there are multiple reads from the same IP address but on separate devices, it will still only be seen as one read. And I'm okay with that since it constitutes such a minor fragment of traffic. This provides an accurate count of reads and I feel is more concise and simpler than many other forms of analytics capture.

tldr;

I use CSS to trigger a url analytics endpoint on `body:hover`, determine useful information from the IP address and user-agent, then hash the IP address with the date to create a unique "read" of a page.

Enjoyed the article? I write about 1-2 a month. Subscribe via [email](#) or [RSS feed](#).



96

Made with Bear ? The text "Made with Bear" is followed by a small blue icon of a bear's head, which is the logo for the Bear note-taking application.