Old Dominion University

ODU Digital Commons

6-2019

Impact of HTTP Cookie Violations in Web Archives

Sawood Alam

Michele C. Weigle

Michael L. Nelson

Impact of HTTP Cookie Violations in Web Archives

Sawood Alam, Michele C. Weigle, and Michael L. Nelson

Department of Computer Science Old Dominion University, Norfolk, Virginia – 23529 (USA)

{salam,mweigle,mln}@cs.odu.edu

ABSTRACT

Certain *HTTP Cookies* on certain sites can be a source of content bias in archival crawls. Accommodating *Cookies* at crawl time, but not utilizing them at replay time may cause cookie violations, resulting in defaced composite mementos that never existed on the live web. To address these issues, we propose that crawlers store *Cookies* with short expiration time and archival replay systems account for values in the Vary header along with URIs.

1 INTRODUCTION AND BACKGROUND

For a long time we have been observing a strange behavior of various web archives when accessing mementos [11] of Twitter pages, some of the mementos would be replayed in non-English languages. This happens even if those Twitter timelines belong to English-speaking personalities, archived using crawlers in North America, and were not requested in any specific language explicitly as shown in Figure 1. After a thorough investigation we figured it out that it is happening due to the use of *HTTP Cookies* for content negotiation by Twitter [4, 9]. We found that almost half of the mementos of Barack Obama's Twitter timeline out of over 9,000 properly archived mementos in five different web archives were in non-English languages, of which, almost half were in Kannada (a regional Indian language) alone, and remaining in 45 other languages (as shown in Figure 2). While language diversity in web archives is generally a good thing, this non-uniform bias is disconcerting when a page is archived in a language not anticipated.

One day we were looking at a Twitter timeline's memento which should have been in English, but was primarily in Portuguese (for the reason described above), after a while we noticed that a notification appeared in Urdu, suggesting that there were 20 new tweets (as shown in Figure 3). On further inspection found that the page contained a sidebar block in English too. Apparently, we were seeing a defaced composite memento of a page that perhaps never existed on the live web. We knew that live-leakage (also known as Zombies) [6] and temporal violations [1] can cause such malformed memento reconstruction and we also knew their potential prevention techniques [3, 8]. However, this mixed-language Twitter timeline issue cannot be explained by zombies nor temporal violations. After a thorough investigation we found that *Cookies* were again the reason behind this replay issue [2, 10].

HTTP is stateless, but often applications need to maintain state information between a client and a server. This is often done with the help of *Cookies* [5]. Servers can send one or more Set-Cookie headers containing strings of name-value pairs along with scope (domain and path) and expiration information. Clients store them and send them back with each request in the scope using Cookie header until expired or removed. *Cookies* are used for session management, personalization, content-negotiation, tracking, and client-side key-value store. The latter is less common now after wide adoption of *LocalStorage* and other similar techniques in web browsers.

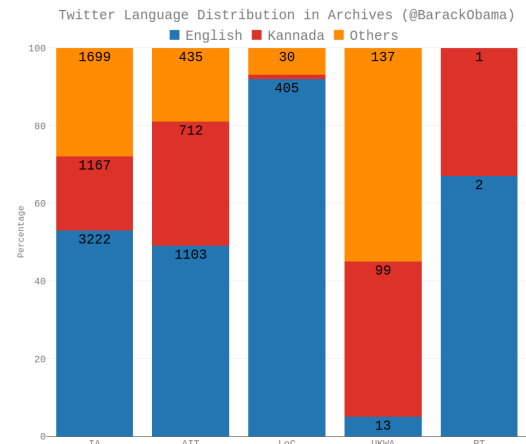Figure 1: Barack Obama's Twitter Timeline is Archived in Urdu, Which Should be in English

Figure 2: Language Distribution of Mementos of Barack Obama's Twitter Timeline in Different Web Archives
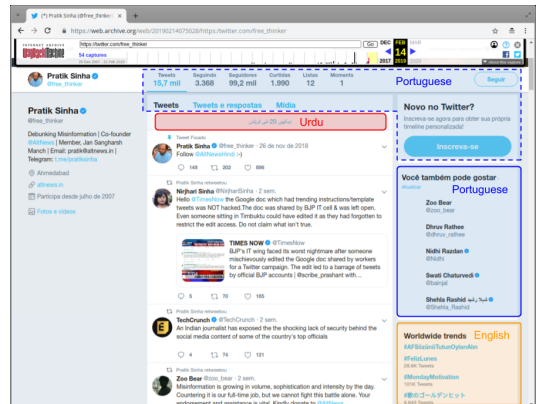

Figure 3: A Memento of a Twitter Timeline Simultaneously in Multiple Languages (Portuguese, English, and Urdu)

```
1  <link rel="alternate" hreflang="x-default"
2                  href="https://twitter.com/">
3  <link rel="alternate" hreflang="fr"
4                  href="https://twitter.com/?lang=fr">
5  ... [45 LINKS TRUNCATED] ...
6  <link rel="alternate" hreflang="kn"
7                  href="https://twitter.com/?lang=kn">
```

Figure 4: 47 Alternate Language Links in Twitter

```
1  $ curl -s -c /tmp/tt.cook https://twitter.com/?lang=ar \
2  >          | grep "<html"
3  <html lang="ar" data-scribe-reduced-action-queue="true">
4  $ grep lang /tmp/tt.cook
5  twitter.com FALSE / FALSE 0 lang ar
6  $ curl -s https://twitter.com/ | grep "<html"
7  <html lang="en" data-scribe-reduced-action-queue="true">
8  $ curl -s -H "Accept-Language: ur" https://twitter.com/ \
9  >          | grep "<html"
10 <html lang="ur" data-scribe-reduced-action-queue="true">
11 $ curl -s -b /tmp/tt.cook https://twitter.com/ | grep "<html"
12 <html lang="ar" data-scribe-reduced-action-queue="true">
```

Figure 5: Language Content Negotiation in Twitter Using Query Parameters, Accept-Language, and Cookies

2 INTERNATIONALIZATION IN TWITTER

Twitter uses standard method of internationalization in its publicly accessible pages by including `alternate` links in 47 supported languages and the `x-default` landing language (as illustrated in Figure 4) to help search engines point users with different locales to the correct language. This technique is utilized by many other multilingual sites such as Facebook and Instagram. However, unlike other popular multi-lingual sites, when accessing a language-specific URI (that contains a `lang` query parameter), Twitter sets a `lang` *Cookie* with the corresponding language (as illustrated in Figure 5). This *Cookie* sticks throughout the session and forces all subsequent pages to be served in that language until another language-specific URI overwrites the *Cookie*. This essentially means Twitter performs language content negotiation using *Cookie* header, though it does not acknowledge it in a `Vary` header.

3 COOKIE VIOLATIONS

Some websites insist that certain *Cookies* are present in a request before they return desired content otherwise they issue redirects and attempt to set those *Cookies*. Failure to send their desired *Cookies* in subsequent requests may turn such sites into crawler traps without any useful content. Web archiving crawlers such as Heritrix[1] have built-in support for cookies. However, the web surfing pattern of crawlers is generally breadth-first-style and comprehensive (not necessarily how human surf the web) for which they use frontier queue of URIs to be crawled. In case of the Twitter's example above, when one of the language-specific `alternate` link is crawled, it impacts all the subsequent non-language-specific URIs due to the `lang` sticky *Cookie*. Kannada being the last language in the list (in Figure 4) gets more exposure before it gets overwritten by another language, resulting in the disproportionate language bias.

Popular archival replay systems (such as OpenWayback[2] and PyWB[3]) utilize only the canonicalized *URI-R* and the datetime of the capture to select a memento to replay. Other request headers that might have been used for content negotiation (such as `Accept-Language` or `Geolocation` etc.) are ignored at replay. Traditional crawlers did not execute JavaScript, so the likelihood of a custom request header being utilized during crawling was minimal,

---

[1] https://github.com/internetarchive/heritrix3

[2] https://github.com/iipc/openwayback

[3] https://github.com/webrecorder/pywb

but it is changing with headless browser-based crawlers. *Cookies*, however, have been supported even in traditional crawlers that are used by some sites for content negotiation (as is the case with Twitter). Moreover, aggregating private archives [7] containing authenticated resources without isolating them based on session *Cookies* has some privacy implications.

Based on our assessment we propose that *Cookies* in crawlers are kept short-lived and pruned frequently to minimize the impact of sticky *Cookies*. Accommodating *Cookies* (or other headers that affect the response) at capture/crawl time, but not utilizing them at replay time has this consequence of cookie violations, resulting in defaced composite mementos. On the contrary, blindly utilizing every *Cookie* as a filter at replay would result in many false negatives. Unfortunately, *Cookie* names are opaque strings and carry no agreed upon semantics to identify ones that affect the payload.

4 CONCLUSIONS AND FUTURE WORK

We identified that certain *Cookies* on certain sites can be a source of content bias in archival crawls. To address this issue we propose that crawlers store *Cookies* with short expiration time explicitly, irrespective of the original value. We also identified that *Cookie Violations* at replay time have the potential to deface composite mementos and reconstruct pages from web archives that never existed on the live web. Archival replay systems need to behave like HTTP proxies or cache servers that accommodate values in the `Vary` header along with URIs. Not every *Cookie* is created equal, those impacting the content need to be identified and accounted for at replay. This is a difficult problem which opens up the possibility for a more extensive research to fully address the issue.

5 ACKNOWLEDGEMENTS

REFERENCES

[1] Scott G. Ainsworth, Michael L. Nelson, and Herbert Van de Sompel. 2015. Only One Out of Five Archived Web Pages Existed as Presented. In *Proceedings of the 26th ACM Conference on Hypertext & Social Media*. 257–266.

[2] Sawood Alam. 2019. Cookie Violations Cause Archived Twitter Pages to Simultaneously Replay in Multiple Languages. https://ws-dl.blogspot.com/2019/03/2019-03-18-cookie-violations-cause.html.

[3] Sawood Alam, Mat Kelly, Michele Weigle, and Michael L. Nelson. 2017. Client-side Reconstruction of Composite Mementos Using ServiceWorker. In *Proceedings of the 17th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL '17)*. ACM, New York, NY, USA, 237–240. https://doi.org/10.1109/JCDL.2017.7991579

[4] Sawood Alam and Plinio Vargas. 2018. Cookies Are Why Your Archived Twitter Page Is Not in English. https://ws-dl.blogspot.com/2018/03/2018-03-21-cookies-are-why-your.html.

[5] Adam Barth. 2011. HTTP State Management Mechanism, Internet RFC 6265. https://tools.ietf.org/html/rfc6265.

[6] Justin F. Brunelle. 2012. Zombies in the Archives. https://ws-dl.blogspot.com/2012/10/2012-10-10-zombies-in-archives.html.

[7] Mat Kelly, Michael L. Nelson, and Michele C. Weigle. 2018. A Framework for Aggregating Private and Public Web Archives. In *Proceedings of the 18th ACM/IEEE-CS Joint Conference on Digital Libraries (JCDL '18)*. 273–282. https://doi.org/10.1145/3197026.3197045

[8] Ada Lerner, Tadayoshi Kohno, and Franziska Roesner. 2017. Rewriting History: Changing the Archived Web from the Present. In *Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security*. 1741–1755.

[9] David S. H. Rosenthal. 2018. All Your Tweets Are Belong To Kannada. https://blog.dshr.org/2018/04/all-your-tweets-are-belong-to-kannada.html.

[10] David S. H. Rosenthal. 2019. The 47 Links Mystery. https://blog.dshr.org/2019/03/the-47-links-mystery.html.

[11] Herbert Van de Sompel, Michael L. Nelson, and Robert Sanderson. 2013. HTTP Framework for Time-Based Access to Resource States – Memento, Internet RFC 7089. https://tools.ietf.org/html/rfc7089.