Even as I visit www.nsfgrfp.org, information about my visit is being sent to Google Analytics. That isn't an isolated incident. As users visit websites, they are constantly being tracked using data through things like web requests, cookies, and browser fingerprinting [4]. This information is used to create detailed profiles of user behavior across websites and can be used to infer additional information about the user. This information allows for ads to be tailored to specific users, which has been shown to enable discrimination on the basis of race, age, sexual orientation, and other categories that users may find sensitive [3,7]. Tools have been developed to protect users from tracking in the form of browser extensions (e.g. Privacy Badger) and built directly into browsers (e.g. Enhanced Tracking Protection in Firefox), but these tracking protection tools can have the unintended consequence of causing websites to fail to function properly, which I term "breakage". This breakage can manifest as things like a website looking messy, not being able to add products to a cart, buttons being disabled, preventing videos from playing, or causing major portions of the websites to be completely missing. For example, with the "strict" level of Enhanced Tracking Protection in Firefox enabled, the Georgia COVID-19 daily status report will not display the interactive data that shows up when tracking protection is disabled, as shown in the figure below. This in turn presents users with a lose-lose situation of dealing with the broken websites or allowing themselves to be tracked by disabling the protection they have in place. **I believe that people shouldn't have to tolerate being tracked just to use the internet and access needed content.**

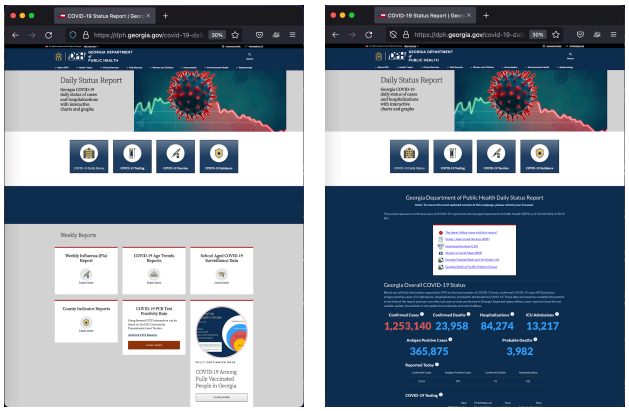

Fig. 1: Georgia COVID-19 status report with (left) and without (right) strict Firefox Enhanced Tracking Protection enabled

Shockingly, this breakage, while a deciding factor when deciding what forms of tracking protection are rolled out in browsers [2], is often only a side note in academic papers or even entirely ignored. Even when breakage is considered, it is usually to see if new tools cause breakage, rather than understanding or fixing the existing issues. The ultimate goal of the proposed project is to allow users to be protected from tracking without having to encounter broken sites.

Research Proposal: To solve this issue and fill the gap in the literature, I propose a three part research plan. The first stage will be to understand how breakage occurs and perform a crawl of websites to assess the prevalence of these issues under different forms of tracking protection. The second phase will be to develop methods to fix the different forms of breakage. The final step will be to conduct a user study to understand how users perceive and handle the breakage that they encounter. Collectively, these steps will provide a framework for reasoning about and fixing breakage in a way that benefits users by optimizing privacy and browser functionality.

Intellectual Merit: Since the discussion of breakage is fairly rare in academic papers, there is no formal method for classifying how breakage manifests from a user perspective. Papers currently mainly rely on manual analysis, checking only if breakage exists and is circumventable [6]. Having a better understanding of how breakage manifests as an issue for a user (e.g. missing content or inoperable buttons) and what specifically causes the breakage (e.g. content not being loaded or code waiting for the return value of a request) is an important first step. I will accomplish this by investigating examples of breakage that were reported through programs like Bugzilla and issue trackers for other tracking protection tools (e.g. Privacy Badger). To augment these reported breakage examples I will cause my own breakage with an extension that I have already built that can be used to mimic these tools that block network requests, strip URL parameters, and/or block cookies. Causing the breakage by simulating these existing tools, rather than directly using them, gives me more granular control over how network requests are modified. To understand the impact and cause of tracking protection, I will analyze breakage using PageGraph [1] which is an instrumented version of the Brave browser. PageGraph records far more information than just looking at the HTML of the webpage or what browser automation tools can currently provide. Creating a method for detecting breakage opens up the possibility of running a measurement study to look at the prevalence of the types of breakage under different forms of tracking

protection. Creating this framework would also allow researchers that are designing new tools to assess the breakage their tools may cause and allow browser vendors to get a sense of the impact that shipping new tracking protection tools would have on their users.

What makes detecting breakage so difficult is differentiating unintended breakage from beneficial side effects of tracking blocking (such as ads not appearing on a page) and the dynamic nature of many websites. To address this and detect breakage, I will make comparisons of the graphs produced by PageGraph over multiple reloads of the page and with programmatic interactions with the site itself both with and without various forms of tracking protection enabled.

After breakage is more meaningfully understood, potential remediation steps need to be created for the different types of breakage. Smith et al. propose modifying JavaScript scripts to prevent them from accessing sensitive data [5]. Building on that idea it could be possible to prevent certain scripts from being accessed to begin with by preemptively returning expected values or by loading common resources from a trusted site rather than from the original link that may be tracking the user. This method would be particularly helpful in the case where code waits for a response and will hang if the request is blocked. These are only a couple of possible ways to mitigate breakage and with a better understanding of breakage, it will be possible to develop more ways of fixing breakage.

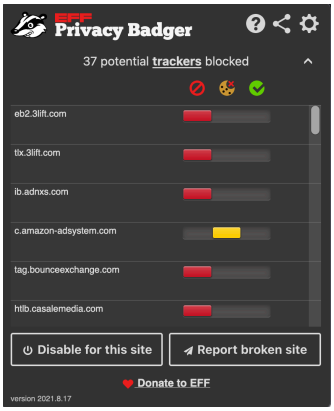

Fig. 2: Privacy Badger interface for users to unblock individual domains when breakage occurs

After understanding the types of breakage that exist, it is also important to understand the way in which the different types of breakage are experienced by users and how they navigate the breakage when encountered. Since people can use different browsers, security settings, extensions, etc., the individual's experience of breakage may be quite particular. For example when visiting a site like TMZ, if breakage occurs and a user is faced with breakage while using Privacy Badger, they are faced with an incredibly long list of domains that they have to consider. In my ongoing work on encrypted DNS settings, I found that many users regularly use more than one browser. Understanding the reasons for that usage of multiple browsers and how that impacts and relates to the breakage that they encounter is an interesting place for investigation. To do this, I propose an online survey of users using Qualtrics and Prolific, both of which I have extensive experience with through other research projects and classes. This survey can also serve as an evaluation of the user facing impacts of the breakage I uncover and provide as a possible metric for how critical the different types of breakage are to address from a user perspective and this could be compared to the prevalence of the different types of breakage reported from the crawl.

Broader Impacts This project has the opportunity to make a material impact on the privacy and user experience of browsers. By fixing the breakage, users will no longer have to make the choice between functionality and privacy. It also creates the opportunity for meaningful collaboration between industry researchers and academia since not only is this a gap in the academic literature, but it is an area that is relevant towards the usage of browsers. Within this project, there are many opportunities to involve people that are just starting out in usable security and privacy research and with the flexibility that the Graduate Research Fellowship would provide with respect to funding for my graduate studies, I would be able to dedicate time towards making that a meaningful mentoring experience, as well.

References

[1] Brave Software. PageGraph '20. https://github.com/brave/brave-browser/wiki/
[2] Browser Privacy: Opportunities and Tradeoffs (Panel). USENIX Enigma '20
[3] Datta et al. Automated Experiments on Ad Privacy Settings. PoPETS '15.
[4] Englehardt et al. Online Tracking: A 1-million-site Measurement and Analysis. CCS '16.
[5] Smith et al. SugarCoat: Programmatically Generating Privacy-Preserving, Web-Compatible Resource Replacements for Content Blocking. CCS '21.
[6] Snyder et al. Most websites don't need to vibrate: A cost-benefit approach to improving browser security. CCS '17.
[7] Sweeney. Discrimination in online ad delivery. CACM '13.