Towards Building Better Mobile Web Browsers for Ad Blocking: The Energy Perspective (WiP Paper)

Nadja Heitmann[1], Benedikt Pirker[1], Sangyoung Park[2], Samarjit Chakraborty[3]
Technical University of Munich[1], Technical University of Berlin[2], University of North Carolina at Chapel Hill[3]
nadja.heitmann@tum.de,bene.pirker@gmail.com,sangyoung.park@tu-berlin.de,samarjit@cs.unc.edu

Abstract

Advertisements, or ads, are a major source of income for Internet-related service companies. Meanwhile, ads and trackers consume significant computing resources and power, which are crucial for mobile devices, and can drain a phone's battery. Moreover, most users find ads annoying, which led to the development of ad blockers. In this paper, we characterize the energy consumption of different ads. We find that ad blocking may either not affect the power consumption at all, or result in power savings of up to 50 % dependent on the web site, reducing the battery life of the mobile device. Our studies are based on the ad blocking engines provided by the Brave browser. We believe that our results might impact how such engines can be designed in the future.

ACM Reference Format:

Nadja Heitmann, Benedikt Pirker, Sangyoung Park, Samarjit Chakraborty. 2020. WIP: Towards Building Better Mobile Web Browsers for Ad Blocking: The Energy Perspective. In *Proceedings of the 21st ACM SIGPLAN/SIGBED Conference on Languages, Compilers, and Tools for Embedded Systems (LCTES '20), June 16, 2020, London, United Kingdom.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3372799.3394372

1 Introduction and Related Work

The majority of the world's most visited web sites today use ads as their primary source of income [1, 7]. Figure 1 shows a 2016 study where it was predicted that the global spending on ads would reach over 300 billion U.S. Dollars in 2019, and around 71 % of it will be attributable to mobile ads [9].

While mobile ads are an integral part of the online ecosystem, they are mostly perceived negatively by the users, even worse on mobile platforms - at best, ambivalence has been reported [15]. As a result, ad blockers are widely used. In 2018, Google released a built-in blocker for its Chrome browser [4]. The system filters *annoying* ads that, e.g., automatically play

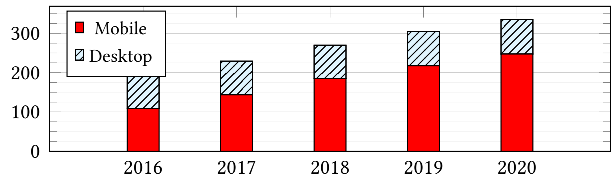

Figure 1. Past and predicted worldwide spending for digital advertisement in billion USD [8, 9].

videos, to force ad providers to adapt to the standards set by the "Coalition for Better Ads" (CfbA) [10]. In 2017, Mozilla also introduced a mobile version of its Firefox browser with a built-in ad blocker. However, their goal is not only a better ad experience, but to increase overall privacy. Mozilla justifies the complete blockage of all ads on every web site with the ads' aggressive tracking of user behavior [6].

However, ads cannot be completely blocked as most web sites cannot sustain without them. They ask the users to explicitly agree to their terms of service, especially in relation to the recent General Data Protection Regulation in the European Union, and sometimes detect the blockers and refuse to provide the contents. As complete blockage of ads is not sustainable from the perspective of the whole ecosystem, we believe alternative models will arise in the future.

An example of such an alternative business model is the one from the Brave browser [2]. It proposes to reward users who view ads using a cyber currency called Basic Attention Tokens. The users can voluntarily pass this benefit to the web sites and gain access to e.g., premium content. Such a model allows to selectively view ads rather than a complete on/off approach. The Brave browser blocks ads by default and provides various options for controlling privacy-related features such as tracking and fingerprint protection. The underlying philosophy is to give "real" control back to the users regarding how they want advertisements to be displayed and how their privacy is managed. However, in order to make choices *in an informed manner* users must be aware of the overhead of viewing ads, especially when browsing on mobile devices. This is only possible through a detailed analysis on the impacts of ads on the processing resources and a phone's battery – as we perform in this paper.

Ads are not only annoying to users and make use of private information, but also influence the user experience by reducing the battery life and increasing the web page loading time. The energy consumed to download an ad, and rendering it on the screen, could potentially be significant. This is to

LCTES '20, June 16, 2020, London, United Kingdom

ACM ISBN 978-1-4503-7094-3/20/06…$15.00
https://doi.org/10.1145/3372799.3394372

be seen in light of the fact that the battery lifetime of a phone is one of the major buying criteria for most users [23]. While mobile browsers are an essential part of every smartphone, they are not yet well optimized for energy efficiency [22].

There has been a large amount of research on application-specific power management for smartphones, especially for multimedia and interactive applications such as games [11–14, 19, 21]. At the same time, while there have been previous works on the power consumption of ads [16, 17, 24, 25, 27, 28], few give hints on how the underlying system. The browser and provided ad can be modified to save power and to prolong the battery life of the phone. In this paper, based on the anticipation that there will be more customization opportunities offered by browsers, we characterize the impacts of ads under more configurations compared to what was done in any previous work. We further propose methods to save power, while maintaining a functional ad infrastructure, since *zero-ads* will not be an option. Our work uses the Brave browser and makes the following contributions:

- We present an ad characterization framework that requires modifying the Brave browser for automated switching between multiple blocking features.
- We provide fine-grained analysis by separately controlling ad blocking and user tracking.
- We provide category-specific analysis of the power consumed by the CPU for different groups of web pages, such as *news*, and *social media*.
- We also investigate the impact of the ad blocking and tracking blocking engines on the loading time of the web pages, that is an essential factor in user experience.
- Based on our findings, we propose power management strategies that can be implemented within the browser and also within the operating system.

2 Web Browsing Advertisements

In this section, we briefly describe how ad blockers work and also introduce our experimental browser Brave.

Advertisements: Most users are willing to accept ads to keep web contents free, but there exist many web pages that embed obtrusive ads, that are perceived as disturbing [10]. Such ads contain automatically-played videos, pop-ups, etc. and are on the Blacklist of the CfbA [10]. They are targeted by ad blockers such as uBlock [26] and consume significantly more battery power compared to less obtrusive ads.

Ad blocking is implemented by checking the web page source code for certain keywords or IP addresses from so-called *filter lists*, such as EasyList [5]. Our goal is to identify the overhead in terms of energy consumption and web page loading time when allowing an ad versus when blocking it both for obtrusive and non-obtrusive ads.

The Brave Browser: The Brave browser [2] is a derivative of Google's open-source Chromium browser, which holds a market share of almost 60 % worldwide [20]. Brave offers a set of advanced integrated privacy and blocking features. At the time of this study, the underlying Chrome version of Brave was 64.0.3282.137. Our Brave version supports five shields that can be manually tuned: (1) for ad blocking and tracking protection, (2) to enable HTTPS for each web page and every connection that is triggered by the particular web page, (3) for third-party-cookie protection, (4) for blocking scripts, and (5) for fingerprinting protection. Brave's ad blocking engine applies the Easylist [5] and "uBlock Origin" filter lists [3] for blocking purposes. The tracking protection engine uses the "Disconnect.me" filter list [3].

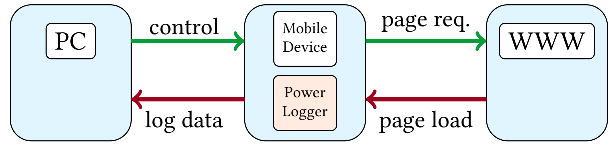

Figure 2. Experimental setup.

3 Ad Blocker Characterization Framework

This section describes the hardware and software framework for our ad blocker characterization. We explain the Odroid-XU3 development board and the software infrastructure we have built to perform the measurements on this board.

Hardware Setup: Our testbed is the Odroid-XU3 development board [18] containing a Samsung Exynos 5422 SoC (used in the Samsung Galaxy S5 smartphone). The chip uses the ARM big.LITTLE architecture and contains two quad-core CPUs, one optimized for performance (A15) and one for energy efficiency (A7). Moreover, the chip features a Mali-T628 MP6 GPU. Additionally, the board contains four sensors to measure the power of the A7, the A15, the GPU and the RAM. The Odroid board runs an Android Kitkat 4.4.4 OS.

Framework and methodology: We automated our measurement procedure by executing a script on a host computer that starts the browser and the power logger on the mobile device, as shown in Figure 2. Before each run, the browser cache is cleared and previously recorded data is removed to ensure unbiased results. The blocking configuration is then given to Brave via Chrome command-line parameters. We collected the power data for 40 s during each measurement. Hence, we not only recorded the loading time itself but also the time during which background scripts are executed. The test procedure was executed 10 times for 23 web pages with five different Brave configurations. We selected the web pages based on their popularity, but also attempted to maintain some diversity in terms of their characteristics. Hence, most pages were picked from the Alexa Top 500 list [1], the other pages are established news sites and blogs.

In addition, we modified the Brave code, such that we could separately control the ad blocking and tracking protection features. We also added command-line parameters for the Brave switches, including separate parameters for ad blocking and tracking protection, to facilitate the automated measurement procedure of our setup.

To get a baseline, we (1) disabled all Brave protection shields. Then, the behavior is similar to the Chrome browser. Configuration (2) is the Brave default configuration that enables the first three Brave shields (see Section 2 for details). The last three configurations are: (3) only ad blocker, (4) only tracking protection, and (5) both combined, without any of the other shields activated. The results are shown in Figure 4.

4 Characterization Results and Analysis

Figure 4 shows our measurement results. The energy values and loading times correspond to the total time taken to load the web page until it becomes ready for the user to interact with, and the energy consumption during the time. For both A15 and A7, the green line pattern shows the energy consumed during page loading, and the solid orange pattern corresponds to the energy consumption of loading and rendering. For example, the Wall Street Journal page over a measurement duration of 40 s incurs a total energy consumption of around 70 J, of which 60 J were consumed by the A15 and 10 J by the A7. Of these, web page loading consumed around 45 J (i.e., 71 % of the total energy consumption), where 40 J were consumed by A15 and 5 J by A7. Note that in Figure 4, the power consumption and loading times are almost twice as high for the web pages on the top row compared to those on the bottom row (maximum 80 J and 20 s on the top and maximum 40 J and 10 s below). These numbers are consistent with the number of trackers and ads on these web pages (see Figure 3).

Discussions: The above results show that the configuration where all blockers are disabled clearly consumes significantly more power than all other configurations. For some web pages, it is even twice as much power, e.g., for The Guardian or Le Monde. Figure 5 shows in detail how ads consume power. Ad blockers were enabled for the left two graphs. Hence, the workload of the CPU cores (bottom) and, consequently, the power consumption (top) drops immediately after page loading. On the right hand side of the figure – with the blockers disabled – the ads and trackers create a high workload even after the page loading has finished.

Note that, except for some of the news pages and the search engine Baidu, the loading times with and without blockers vary little. This also shows that most ads and trackers are actually triggered after the web page is fully loaded, not during loading itself. For search engines, the loading time and energy consumption with the Brave default mode are higher than the other configurations. This is due to an HTTPS feature that tries to encrypt insecure connections issued by the search engines while obtaining search results.

Implications of our results: Although there are many web pages for which the power consumption does not depend on ad blockers, there are others where ads heavily increase the devices' power consumption. Since web browsers are one of the most used applications on mobile phones, these results imply that decisions on ad blocking should also take power consumption into account. In particular, users get no feedback on how their decision to turn off an ad blocker impacts the battery life. These results show that browsers that allow a more fine-grained control on blocking ads, in the future, can implement what could be referred to as **battery charge level-aware advertisement control**. Here, the ad blocking decision would be based on the battery charge level and the energy requirement associated with an ad (that can be predetermined for different web pages, as in this study). Such presets would allow users a seamless browsing experience, while ensuring that the battery does not get drained in an unexpected manner.

These results also point to how ads could be designed in the future. As different versions of a web page are displayed depending on the device type anyways (e.g., mobile vs desktop), a browser could also fetch ads based on the charge level of the device's battery. For example, complex ads that play a video clip, will only be fetched if the battery has sufficient charge. Otherwise, less energy demanding static textual ads will be fetched. This feature can be easily implemented into existing browsers as the battery charge level can be read out from the Android API by any application. Finally, the results in Figure 5 imply that if the browser can inform the Android power manager about the web page being loaded and the workload to be incurred by its associated ads, then the power manager could actively reduce the processor's operating frequency in order to not allow the ads to refresh rapidly and consume too much power. What impact such policies will have on user perception needs to be studied.

5 Concluding Remarks

The increase in the usage of ad blockers and web sites rejecting to show contents in response to such blocking, mandates alternative ad models such as the one proposed by Brave. Such schemes give more control on which ads to view, back to the users, rather than a simple accept/reject, while still providing revenue to the advertisers. This paper suggests that web browsers for mobile devices now need to take an additional step: Given the constant usage of mobile phones today, battery-life is a major factor determining their usability. In addition, browsers are one of the most used applications on mobile phones and many web sites rely on advertisements as their sole revenue source. Hence, browsers should take into account the battery charge level of a device when determining which advertisements to fetch from a web site and how to display them. To support the design of such browsers, in this paper, we show detailed results on the energy cost of advertisements for a wide selection of web sites. These results show that ads versus no ads can result in almost 100 % increase in energy consumption for web sites, while for others there is no energy impact. We believe that in addition to browsers, how web sites and advertisements are designed and programmed might also be influenced by these results.

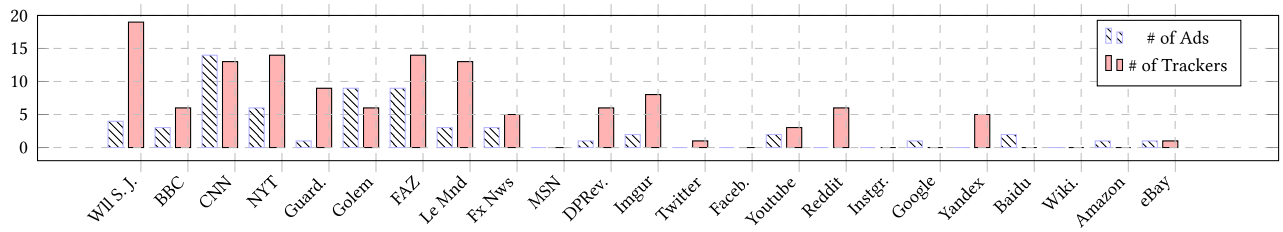

Figure 3. The number of trackers and ads found by Brave when loading each web page.

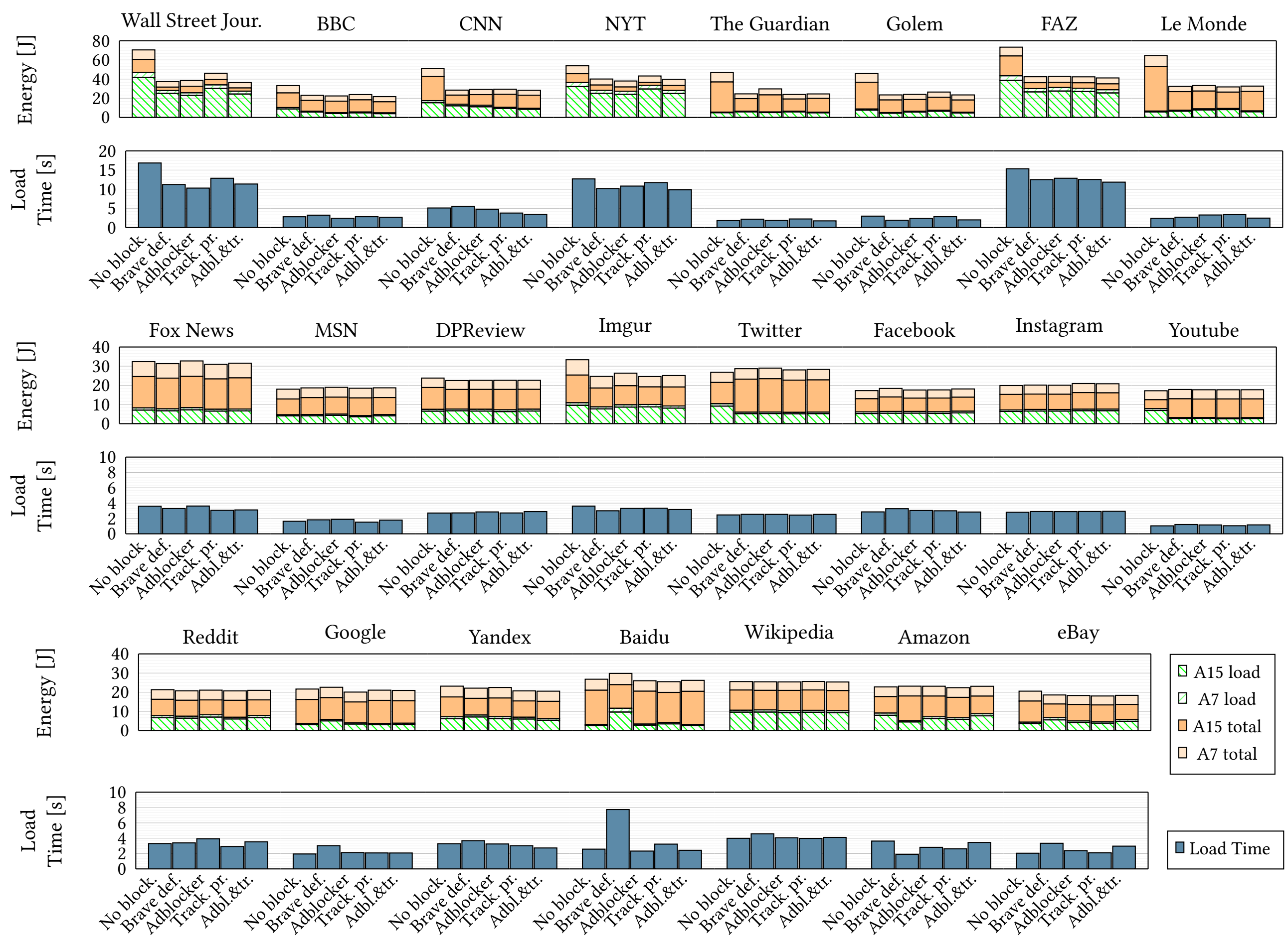

Figure 4. Energy consumption and loading times of web pages for different Brave configurations.

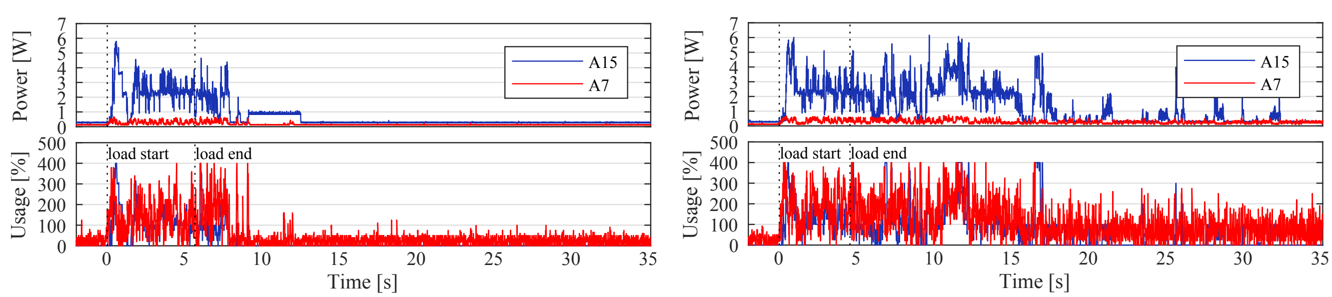

Figure 5. Power, CPU usage and frequency for loading the CNN page, blockers enabled (left) and disabled (right).

References

[1] Alexa top 500. https://www.alexa.com/topsites.

[2] Brave browser. https://brave.com/.

[3] Brave's filter lists. https://community.brave.com/t/all-about-ad-blocking/10004.

[4] Chrome's built-in adblocker. https://www.theverge.com/2017/12/19/16797522/google-chrome-ad-blocker-release-date-announced.

[5] Easylist. https://easylist.to/.

[6] Firefox focus. https://blog.mozilla.org/blog/2016/11/17/introducing-firefox-focus-a-free-fast-and-easy-to-use-private-browser-for-ios/.

[7] Internet device of choice in asia. https://spectrumfutures.org/smartphone-is-the-internet-access-device-in-most-of-asia/.

[8] Mobile advertising spending worldwide. https://www.statista.com/statistics/280640/mobile-advertising-spending-worldwide/.

[9] Online advertising spending worldwide. https://www.statista.com/statistics/237974/online-advertising-spending-worldwide/.

[10] Standards for better ads. https://www.betterads.org/standards.

[11] Dietrich, B., and Chakraborty, S. Lightweight graphics instrumentation for game state-specific power management in android. *Multimedia Syst. 20*, 5 (2014), 563–578.

[12] Dietrich, B., Goswami, D., Chakraborty, S., Guha, A., and Gries, M. Time series characterization of gaming workload for runtime power management. *IEEE Trans. Computers 64*, 1 (2015), 260–273.

[13] Dietrich, B., Nunna, S., Goswami, D., Chakraborty, S., and Gries, M. Lms-based low-complexity game workload prediction for DVFS. In *28th International Conference on Computer Design (ICCD)* (2010).

[14] Dietrich, B., Peters, N., Park, S., and Chakraborty, S. Estimating the limits of CPU power management for mobile games. In *IEEE International Conference on Computer Design (ICCD)* (2017).

[15] Fessenden, T. The most hated online advertising techniques. https://www.nngroup.com/articles/most-hated-advertising-techniques/, 2017.

[16] Gui, J., Li, D., Wan, M., and Halfond, W. G. J. Lightweight measurement and estimation of mobile ad energy consumption. In *International Workshop on Green and Sustainable Software (GREENS)* (2016).

[17] Gui, J., Mcilroy, S., Nagappan, M., and Halfond, W. G. J. Truth in advertising: The hidden cost of mobile ads for software developers. In *International Conference on Software Engineering (ICSE)* (2015).

[18] Hardkernel co., Ltd. Odroid-XU3. http://www.hardkernel.com, 2015.

[19] Huang, Y., Chakraborty, S., and Wang, Y. Using offline bitstream analysis for power-aware video decoding in portable devices. In *13th ACM International Conference on Multimedia (MM)* (2005).

[20] NetMarketShare. Browser Market Share. https://netmarketshare.com, 2018.

[21] Pasricha, S., Ayoub, R., Kishinevsky, M., Mandal, S. K., and Ogras, U. Y. A survey on energy management for mobile and IoT devices. *IEEE Design Test* (2020, online access).

[22] Peters, N., Park, S., Clifford, D., Kyostila, S., McIlroy, R., Meurer, B., Payer, H., and Chakraborty, S. Phase-aware web browser power management on hmp platforms. In *ACM International Conference of Supercomputing* (2018).

[23] Pilon, A. Smartphone battery survey: Battery life considered important. https://aytm.com/blogmarket-pulse-research/smartphone-battery-survey/, 2016.

[24] Rasmussen, K., Wilson, A., and Hindle, A. Green mining: Energy consumption of advertisement blocking methods. In *International Workshop on Green and Sustainable Software (GREENS)* (2014).

[25] Simons, R., and Pras, A. The hidden energy cost of web advertising. *CTIT Technical Reports* (2010).

[26] uBlock LLC. uBlock. https://www.ublock.org/, 2015.

[27] Vallina-Rodriguez, N., Shah, J., Finamore, A., Grunenberger, Y., Papagiannaki, K., Haddadi, H., and Crowcroft, J. Breaking for commercials: Characterizing mobile advertising. In *Internet Measurement Conference (IMC)* (2012).

[28] Visser, A. The effect of ad blockers on the energy consumption of mobile web browsing. In *Twente Student Conference on IT* (2016).