LinCa: A Page Loading Time Optimization Approach for Users Subject to Internet Access Restriction

Chen Ling, Lei Wang✉, Jun Lang, Qiufen Xia, Guoxuan Chang, Kun Wang, Peng Zhao
Dalian University of Technology
lei.wang@dlut.edu.cn

ABSTRACT

Internet access restriction in various areas unceasingly poses inconvenience for users. Unblocked websites or web pages always accompany couple fate-to-fail links from blocked servers due to such restrictions, while users wait for a long time to see contents of these blocked/*invalid* links. Therefore, it is better to directly show timeout links to users for better experiences rather than making users excessively wait.

In this paper, we present LinCa (Links Catcher), a novel approach that fully considers the Internet access restriction and reduces page loading time on client-side by parsing all HTTP requests, and intercepting all invalid links when a web navigation starts. Thus, we first create and maintain a *Rule Base* to store invalid links under given access restriction rules. We then update the Rule Base periodically to cover as many invalid links as possible and remove links that become valid. We finally demonstrate the effectiveness of LinCa through experiments by building and deploying a Chrome extension. Experimental results show that LinCa can reduce page loading time with average 28.12% of original page loading time for our data sets.

KEYWORDS

Page Loading Time; Internet Access Restriction; User Experience; Ad-Blockers

ACM Reference Format:

Chen Ling, Lei Wang✉, Jun Lang, Qiufen Xia, Guoxuan Chang, Kun Wang, Peng Zhao. 2018. LinCa: A Page Loading Time Optimization Approach for Users Subject to Internet Access Restriction. In *WWW '18 Companion: The 2018 Web Conference Companion, April 23–27, 2018, Lyon, France*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3184558.3186933>

1 INTRODUCTION

For reasons such as cyber security and work efficiency, we may get restrictions to access some specific resources, especially in schools and public facilities. And increasingly complex network environment only drives institutions and organizations to deploy more access restrictions to secure their environments and to protect their data from malware and hackers. One consequential inconvenience caused by Internet access restriction is the latency on page loading. Amongst requested resources, some may locate on blocked servers

although majority of them are not. While no mechanism to validate the accessibility of servers before browser requests, a browser has to call unreachable servers for resources and to wait for responses till timeout. Then requests will be terminated and failed status will be confirmed. This pattern repeats for same situations, with CPU resources wasted and users' patience consumed.

Previous works that analyzed page loading time in different perspectives [2], did not consider Internet access restriction. Although some have considered latency problems [1], most of them gave solutions for web developers instead of users [4]. Whereas no work, to the best of our knowledge, addresses it in formality, we find approaches to a similar problem on online advertisements – research on studying the mechanism of online ad-blockers where scholars found the approach on javascript-injected advertisements is blocking them from being loaded [3].

We address the problem of Internet access restrictions: namely how browsers inefficiently load pages containing third-party resources which are blocked at the network level. Therefore, we propose LinCa and identify the major factors on page loading latency subject to Internet access restrictions. To the best of our knowledge, our paper is one of the first attempts to formally address this from the client side.

2 METHOD AND ARCHITECTURE

First, we give an analysis of LinCa. A link is described as *invalid* if it is sent to a server whose returned HTTP status message is either a failed status with undefined response type or an error response. Otherwise, requests that result in responses excluding client and server error responses are coined *non-invalid*.

LinCa blocks invalid links by setting up a user-oriented *Rule Base*, which composes of one regular expression table and a hash table. The scenario is under an Internet access restriction where users are denied access to a given set of hosts or servers by ISP (Internet Service Provider). We argue that size of the set is rather dynamic instead of static, with new servers or hosts can be appended or removed. So we use local storage for invalid requests which will not be responded and have been taking up the relevant resources. And we update the user's Rule Base from the user's own web browsing data. A blocked domain often associates with multi-fold addresses. Addresses with same domain can be labeled by same regular expression to avoid redundancy and reduce storage space. For addresses that cannot be expressed in regular expression, such as short URLs, we set up a hash table to store them to support quick search. User-oriented approach on designing storage strategy meets the needs of storing invalid links for a given user while maintaining a minimized storage space locally. With different Internet access restrictions, users' Rule Bases may vary.

This paper is published under the Creative Commons Attribution 4.0 International (CC BY 4.0) license. Authors reserve their rights to disseminate the work on their personal and corporate Web sites with the appropriate attribution.

WWW '18 Companion, April 23–27, 2018, Lyon, France

© 2018 IW3C2 (International World Wide Web Conference Committee), published under Creative Commons CC BY 4.0 License.

ACM ISBN 978-1-4503-5640-4/18/04.

<https://doi.org/10.1145/3184558.3186933>

Before being sent out by a browser, all requests will be intercepted to extract addresses of requested links. Then, the extracted link is used as input to the Rule Base for pairing invalid ones. Any yielded output will designate LinCa to intercept the about-to-send request to achieve invalid link block. Once a page finishes loading, all released links will be revised for accessibility to update Rule Base by appending unlisted invalid links and removing listed ones which become non-invalid. A time-to-expire tag is appended to each invalid link to maintain effectiveness and space minimization. During links revision, LinCa checks expired invalid links for accessibility. Inaccessible ones' tag will be reset while accessible removed from Rule Base. In Figure 1, we present the work-flow of LinCa. In summary, the core procedures of LinCa are as follows:

- 1) The Rule Base for Invalid Links Storage
- 2) The Capture of Website Request
- 3) The Identification and Block of Invalid Links
- 4) The Self-adaptive Mechanism for Rule Base

Figure 1: Flow Chart of Web Page Loading Time Optimization Principle

3 EXPERIMENTS

We examined top 400 websites – ranked by Amazon’s Alexa. Collected statistics show that 62.53% of average page loading time is consumed by invalid requests, whose average processing time is 7.91 times longer than that of valid ones.

We then implemented a Google Chrome extension LinCa to demonstrate the efficiency of optimization on page loading time. Experiments are conducted to test the overall effect of reduced page loading time latency and requests loss which are essential for user experience on the web.

Experiments were conducted on a Windows 7 computer with Intel Core i5-4570 CPU and 2GB memory. Internet access restrictions we chose are a campus Internet in China for both university and nation deployed firewalls. We installed our extension in this computer’s Chrome with version 49.0.2623.110.

The efficiency of LinCa on page loading time optimization is proved by experiments using a Google Chrome extension we built. Results demonstrate a reduced page loading time with average 28.12% of original page loading time after using LinCa while only sacrifice about 2.15% of non-invalid requests. Figure 2 displays

the results. The average page loading time of our tested websites subject to Internet access restriction is reduced from 43530 ms to 12241 ms by using LinCa. The experiment we run shows positive results with effectively reduced page loading time and negligible requests loss rate. We also conducted a survey of 105 volunteers in our lab to receive feedback.

Figure 2: Average overall page loading time and request processing time illustrate the efficiency of LinCa. First three columns are calculated before installation of LinCa and following three are after.

4 CONCLUSIONS

We identify the major factor on page loading time latency subject to an Internet access restriction is loading invalid requests. We propose LinCa to optimize page loading time on client-side after identification of invalid requests. We institute Rule Base to host invalid links locally and update it periodically. By matching then intercepting invalid links to reduce page loading time latency. We implement and deploy a Google Chrome extension based on our novel approach. We conduct extensive experiments based on real-world Internet access restriction to test the efficiency of our approach. Results demonstrate a successfully reduced page loading time with merely 2.15% of non-invalid requests loss which indeed improves user experience on the web.

ACKNOWLEDGMENTS

The work is supported by National Natural Science Foundation of China with No.: 61733002 and the Fundamental Research Funds for the Central University with No.: DUT17LAB16, No.: DUT2017TB02. This work is also supported by Tianjin Key Laboratory of Advanced Networking (TANK), School of Computer Science and Technology, Tianjin University, Tianjin, China, 300350.

REFERENCES

- [1] Google Inc. 2015. Reduction of web page load time using http header compression. (2015).
- [2] Ravi Arun Netravali. 2015. Understanding and improving Web page load times on modern networks. (2015).
- [3] Elliott L. Post and Chandra N. Sekharan. 2015. Comparative Study and Evaluation of Online Ad-Blockers. In *International Conference on Information Science and Security*. 1–4.
- [4] Munyaradzi Zhou, Giyane Maxmillan, and Mutembedza Nyasha. 2013. Effects of Web Page Contents on Load Time over the Internet. *International Journal of Science Research* 28, 5 (2013), 2319–7064.