

Usability of censorship circumvention tools

*Rachel Gonzalez Rodriguez, Elisabeth Ostryanin, Harshini Sri Ramulu, Niklas Niere,
Juraj Somorovsky, Yasemin Acar*

Abstract

We report on a preliminary usability analysis of open source DPI-based censorship circumvention tools. We assessed twelve such tools via expert walkthroughs and content analysis of 65 usability-related GitHub issues, finding usability issues with installation, documentation, configuration, and communicating whether circumvention is active. We hope to build on these preliminary findings to systematically improve the usability of these tools, to allow censored users better access to the internet.

1 Introduction

In many countries around the world, governments impair free access to the internet through censors. These censors analyze users' network traffic to identify and block undesired traffic. Besides crude IP blocking, censors employ Deep Packet Inspection (DPI) to extract and evaluate sensitive information from network requests. [Figure 1](#) depicts the process of a censor blocking a request for a website.

While many users rely on VPNs to circumvent censorship, VPN use inherently depends on external servers, which may themselves be blocked or may be undesirable in certain scenarios. To circumvent censorship, affected people can use dedicated *circumvention tools*. A circumvention tool is a software that runs on a local device and alters the device's traffic to circumvent censors. [Figure 1](#) shows how a circumvention tool modifies an otherwise censored request that overwhelms the censor but is compliant with the receiving server. In our example, this is achieved by capitalizing selected characters

in the domain name. In general, circumvention tools apply different modifications; some of them are general, while others are tool-specific. Circumvention tools are often developed by affected people and thus focus on censorship in specific countries. Popular circumvention tools are GoodbyeDpi [\[7\]](#), Zapret [\[1\]](#), ByeDPI [\[3\]](#), and SpoofDPI [\[9\]](#). While these tools should in theory support free access to the internet in censored countries, in practice, usability problems can hinder their effective widespread use. For example, multiple tools require expertise in using Command Line Interfaces (CLIs), which may be prohibitive to many potential users. In this work, we conduct a usability analysis of censorship circumvention tools via expert walkthroughs [\[5\]](#) and content analysis of GitHub usability issues [\[2\]](#).

This mixed method approach allows us to answer the research question: *What are usability issues in censorship circumvention tools?*

To systematically assess the usability of censorship circumvention tools, we conducted a cognitive walkthrough of twelve tools, finding that ten fail to even communicate to users whether censorship circumvention is active. For three tools, even with inspecting source code, we could not determine whether circumvention was actually being applied. We also conducted a content analysis of 65 user-generated GitHub issues that describe usability problems.

2 Background & Related Work

Previous work has identified usability problems in various tools used to circumvent censorship, including Tor Launcher [\[4\]](#). Other human-centered censorship circumvention research highlights usability issues for Chinese and Russian users who attempt to use VPNs to circumvent their censors [\[8\]](#). Issues include availability of the service, as well as a game of cat and mouse with censors, where different VPNs work at different times, as censors update their capabilities. Further prior research explores user posts in censored environments, also finding preferences for VPNs that adapt to censor strategies [\[6\]](#). In contrast to these prior studies mostly on commercial VPNs,

we explore the usability of open source tools that employ DPI-based censorship circumvention.

3 Methodology

We conducted a mixed-method study. First, we conducted cognitive walkthroughs of DPI-based open source censorship circumvention tools, documenting issues with their installation, documentation, and implementation. Tools were chosen via searching for them in a search engine, as well as exploring a website where users discuss censorship circumvention tools¹, and exploring tools referenced in either search result. Second, we qualitatively analyzed user complaints regarding the same censorship circumvention tools on their GitHub issues (with the exception of ByeByeDPI Android, where we found no GitHub issues). We detail our ethical considerations and study limitations in Appendix A.

Walkthroughs We conducted expert walkthroughs with three researchers [5], installing and deploying every tool, and assessing whether circumvention was active.

GitHub Data Collection One researcher collected GitHub issues from the tools own GitHub pages. First we sorted the posts by newest, then collected a preliminary set of ca. five posts (where available) on usability issues per tool for further analysis. We collected the links to the posts and used the automatic browser translation to analyze the posts in Russian; translations were checked by a researcher fluent in Russian.

GitHub Data Analysis Two researchers conducted content analysis of the GitHub issues [2]. Both researchers analyzed all 65 issues, discussing any discrepancies until they agreed on their analysis. Our content analysis developed iteratively; we eventually categorized the issues as follows: “app identified as malware”, “configuration issue”, “documentation issue”, “feature request”, “installation issue”, “other usability issue”, “tool not working (at all)”, “tool not working for specific uses” (for example, users desired to use the tool to access Roblox or YouTube) and “issues with using the tool”.

4 Preliminary Results

4.1 Usability issues identified using walkthroughs

Out of the twelve tools, eight had solely CLIs, three had a Graphical User Interface (UI), and one had both. We found several user experience and usability issues, most obviously documentation issues, including missing or incorrect installation guides (ByeDPI), assumed knowledge about Go build workflows (GhostTCP), or GCC compilers (ByeDI), and a lack of documentation for manually configuring browser proxy settings (GoodbyeDPI). Sometimes the documentation directed users to ask clarifications in—inactive—GitHub

discussions. In addition to documentation problems, ten tools did not inform users that they were connected to the internet; only one tool (DPIProxy) did. When we checked with packet analyzer Wireshark², we could confirm that circumvention was only applied by six tools; none of these indicated this to users.

4.2 Usability problems in GitHub issues

Our analysis of 65 usability-related issues (33 closed/ resolved) yielded problems including documentation, installation, tool use, and tools not working, documentation issues (3) discussed included lack of clarity about what is current: *“I recently noticed that the `-wait-send (-Z)` parameter, introduced in #277, isn’t mentioned anywhere, including the `help_text`. Is this intended, or should I add help text? Or is this parameter deprecated?”* (issue 36), as well as requests for more documentation *“How do I even enable it? Why is the documentation so sparse?”* (issue 10).

Installation issues (3) were also discussed: *“I’m not too savvy in Linux, so I’m at a loss trying to set this up. Any guide or support would be highly appreciated.”* (issue 43) Twelve issues discussed different challenges regarding tool usage, three highlighted the tools’ usability challenges and one specifically mentioned: *“Is it possible to explain in a language accessible to the average worker, so to speak, not the most intellectual labor, how this thing works on the following problem that I am currently facing: I need to make a certain application (essentially any application) work under the strict restrictions of my country (Russian Federation)”* (issue 39).

Other issues discussed tools not working at all (5) or not working for specific desired use cases (17), including accessing YouTube, Roblox, or Instagram. Further, we find issues about configuration (10), tools being flagged as malware by devices (2), and feature request (3). Five issues discussed other topics, including UI suggestions, uninstalling, and warning messages.

5 Conclusion & Future Work

We found evidence for usability problems in censorship circumvention tools both by systematically twelve censorship circumvention tools, and by analyzing 65 usability-related GitHub issues. Main problems centered around installation, configuration, lack of clarity whether circumvention is applied, and insufficient documentation. We further note that some tools are targeted towards specific countries, and it may be unclear to censored users which tools will work for their specific censor. Further work should systematically improve tools’ usability, to enable users to choose the tool that works for them, easily deploy the tool, and understand whether circumvention is active.

¹ntc.party

²<https://www.wireshark.org/>

References

- [1] bol van. bol-van/zapret, April 2026. original-date: 2016-02-15T13:30:50Z.
- [2] Hsiu-Fang Hsieh and Sarah E Shannon. Three approaches to qualitative content analysis. *Qualitative health research*, 15(9):1277–1288, 2005.
- [3] hufrea. hufrea/byedpi: Bypass DPI, 2024.
- [4] Linda Lee, David Fifield, Nathan Malkin, Ganesh Iyer, Serge Egelman, and David Wagner. A usability evaluation of tor launcher. *Proceedings on Privacy Enhancing Technologies*, 2017.
- [5] Ben Light, Jean Burgess, and Stefanie Duguay. The walk-through method: An approach to the study of apps. *New media & society*, 20(3):881–900, 2018.
- [6] Elham Pourabbas Vafa, Mohit Singhal, Poojitha Thota, and Sayak Saha Roy. Learning from censored experiences: Social media discussions around censorship circumvention technologies. In *2025 IEEE Symposium on Security and Privacy (SP)*, pages 1325–1343. IEEE, 2025.
- [7] ValdikSS. ValdikSS/GoodbyeDPI, April 2026. original-date: 2017-05-16T18:48:52Z.
- [8] Diwen Xue, Anna Ablove, Reethika Ramesh, Grace Kwak Danciu, and Roya Ensafi. Bridging barriers: A survey of challenges and priorities in the censorship circumvention landscape. In *33rd USENIX Security Symposium (USENIX Security 24)*, pages 2671–2688, 2024.
- [9] xvzc. xvzc/spoofdpi, April 2026. original-date: 2021-12-26T07:24:14Z.

A Ethical Considerations and Limitations

Ethics A multi-step study based on this preliminary work has received preliminary ethics review from our institution’s ethics committee. This preliminary work does not fall under the purview of our institution’s ethics committee,

Limitations As multiple tools are developed against specific censors, the original content is in languages other than English, mostly Russian. We relied on both browser translation and one team member’s ability to read Russian to work with Russian language GitHub posts. All quotes originally in Russian were translated by a browser’s translation, and checked for correctness by a researcher who is fluent in Russian.

B Deployment of censorship circumvention tools

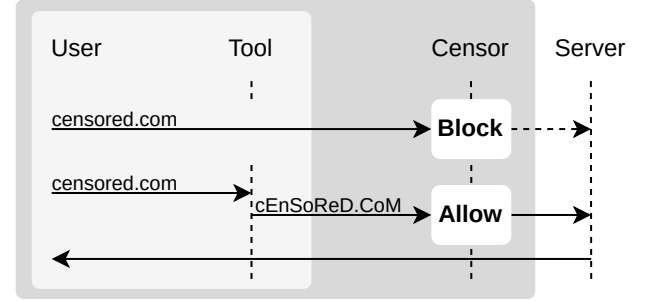


Figure 1: Deployment of circumvention tools. The unmodified request is blocked by the censor; the request modified by the locally running circumvention tool circumvents the censor by altering the original request.