Investigating the VPN Ecosystem Through the Lens of Security, Privacy, and Usability

by

Reethika Ramesh

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in The University of Michigan
2023

Doctoral Committee:

Associate Professor Roya Ensafi, Chair
Clinical Assistant Professor Sol Bermann
Professor J. Alex Halderman
Associate Professor Baris Kasikci
Assistant Professor Elissa M. Redmiles

Reethika Ramesh
reethika@umich.edu
ORCID iD: 0000-0001-5082-7493

ACKNOWLEDGEMENTS

I would first like to express my gratitude to my advisor Prof. Roya Ensafi who from my very first meeting with her as a potential PhD advisee made me share her enthusiasm and passion for science. I learned how to formulate effective approaches toward research problems, create and maintain connections, and conduct meaningful research. I remain in awe of her ability to develop and deliver on her ambitious and impactful visions, and I am grateful to have her as a lifelong mentor and friend. I am also grateful to my committee members, Prof. J. Alex Halderman, Prof. Sol Bermann, Prof. Elissa M. Redmiles, and Prof. Baris Kasikci for their continued guidance, feedback, and earnest support. I want to thank all the other professors and staff in the the CSE department for all their support.

Next, I want to thank all of my collaborators and co-authors, mentors, colleagues, classmates: Adrian Stoll, Alexandra Dirksen, Andrey Viktorov, Anjali Vyas, Anna Ablove, Anne Edmundson, Apurva Virkud, Arham Jain, Armin Huremagic, Benjamin Mixon-Baca, Chad Sharp, David Fifield, Dirk Rodenburg, Diwen Xue, Doug Madory, Eric Wustrow, Jakub Dalek, Jed Crandall, Kyle Astroth, Leonid Evdokimov, Matthew Bernhard, Michalis Kallitsis, Muhammad Ikram, Philipp Winter, Rod Hynes, Sam Korman, Shreyas Chandrashekaran, Simone Basso, Steven Sprecher, ValdikSS, Victor Ongkowijaya, Will Scott, Yusei Uehara. All of you made my journey that much more memorable and meaningful.

I am especially grateful to Neil Spring and Ramakrishna Padmanabhan, both of whom gave me the opportunity to explore research at the University of Maryland as an undergraduate student from India, exposed me to a wholly new (to me) career option, and ultimately put me on this path to becoming a researcher and a scientist.

I want to thank all of my lab mates whose camaraderie and support got me here: Aaron Ortwein, Andrew Kwong, Anna Ablove, Armin Huremagic, Diwen Xue, Elisa Tsai, Haizhong Zheng, Hieu Van Le, Ingab Kang, Marina Minkin, Piyush Sharma, Ramakrishnan (Ram) Sundara Raman, Renuka Kumar, Wayne Wang, and all my CSEG friends.

I would also like to extend my thanks to my friends inside and outside of the UMich CS department. Thank you Abraham Mhaidli and Yixin Zou for always being generous with your research expertise, and for all the game and tea nights. Thank you Allison McDonald and Ritchie Wilson for *everything*. I could not have made it through grad school or all those months of COVID lockdowns and isolation in Ann Arbor without the two of you, and your support, friendship, and generosity have been invaluable to me. Thank you Amulya Bathini and Anand Gopalakrishnan for fostering our decade-long friendship despite all the distance. Thank you, Ben Vandersloot, Matt Bernhard, and Monica Childers for being there for me, and for helping me find my ground both in the department and in life in Ann Arbor. Thank you, Emily Kubik, Steven Sprecher, and Dakota for being one of my first friends in the department, for working with me through many deadlines, and for hosting game nights with the best hospitality. Thank you, Elisa Tsai, Diwen Xue, and Anna Ablove for your friendship and camaraderie. I could not have asked for a better group of people to work and grow with. Thank you Jane Im for fiercely championing me and all your friends, for sharing in my happiness (and sorrows), and for including me in yours. Thank you Renuka Kumar for being a rock and great friend through all of the tumults of grad school. Thank you, Ramakrishna, Janani, Ranjani, Aravind, and Shyam for taking me under your wing, and for introducing me to life as a graduate student in the United States. Thank you Niveditha Venkatachalam for being my best friend, for forgiving me for all those times I have been less than that, and for always letting us pick up where we left off last time. All of you special people have become (or have always been) lifelong friends and family, and I hope you know the positive effect you have had on my life. I would not have completed this without your invaluable support and encouragement.

Words cannot suffice, but I want to thank Ram, my best friend, confidant, collaborator, and now partner who has championed me and supported me throughout my journey. Thank you for everything.

Finally, I want to extend a special thank you to my family, many of whom supported me from more than 8500 miles away. Thank you Vaijayanthi, Ramesh, Madhurika, Sundarambal, Chandrasekaran, Seethalakshmi, Sankaran, Balaji, Radha, Megha, Mahadevan, Rema, and my entire extended family. All of this would not have been possible without your prayers, wishes, and support. A special thank you to my *Amma*, Vaijayanthi Chandrasekaran, who has never doubted that I can achieve what I set out to do, and to my *Thatha*, the late C.K. Chandrasekaran, I regret that you never saw me graduate and I wish you were here to read this. Thank you for earnestly and wholeheartedly supporting all my endeavors.

என் பாஷையிலும் சொல்கிறேன்:

என் அம்மா, அப்பா, தங்கச்சி, அம்மம்மா, தாத்தா, தாத்தி, தாத்தா, மாமா, மாமி, மாமா மகள், பெரியப்பா, பெரியம்மா, மற்றும் எனது உறவினர்கள் அனைவருக்கும் எனது மனமார்ந்த நன்றியை தெரிவித்துக் கொள்கிறேன்.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS ii

LIST OF FIGURES ix

LIST OF TABLES xiii

LIST OF APPENDICES xv

ABSTRACT xvi

CHAPTER

I. Introduction 1

1.1 Thesis Statement and Contributions 4
1.2 Proliferating Deep Packet Inspection Technologies and Privacy Harms 5
1.3 Empirical Understanding of VPN users and VPN providers 6
1.4 Building VPNalyzer: Systematic, Semi-Automated Investigation of the VPN Ecosystem 7
1.5 Leveraging Cross-Layer Network Latency Measurements to Detect Proxy-Enabled Abuse 8

II. Proliferating Deep Packet Inspection Technologies and Privacy Harms 10

2.1 Background and Related Work 14
2.1.1 Understanding Censorship Studies 16
2.1.2 Russian Information Control 20
2.2 Experiment Design 22
2.2.1 Acquiring the RUssian BLocklist ($RUBL$) 23
2.2.2 Establishing Sound Control Measurements 24
2.2.3 Conducting Direct Measurement 25
2.2.4 Conducting Remote Measurement 30
2.3 Ethical Considerations 31

2.4 Data Characterization 33
2.4.1 IPs and Subnets 33
2.4.2 Domains 34
2.5 Results 38
2.5.1 Historical Analysis of Russian Blocklist 38
2.5.2 Characterizing Censorship of $RUBL_{dom}$ 39
2.5.3 Characterizing Censorship of $RUBL_{ip}$ Measurements . 44
2.5.4 Characterizing Censorship of $RUBL_{sub}$ Measurements . 46
2.6 Broader Impact of This Work 46
2.7 Discussion and Conclusion 47

III. Empirical Understanding of VPN Users and VPN Providers 50

3.1 Background & Related Work 54
3.1.1 Virtual Private Networks 54
3.1.2 User Adoption of VPNs and Other Tools 55
3.2 Methods 57
3.2.1 User Survey 57
3.2.2 Qualitative Interviews of VPN Providers 61
3.2.3 Recruitment 62
3.2.4 Ethics 63
3.2.5 Limitations 63
3.3 Data Characterization and Validation 64
3.4 Results from the User Survey 67
3.4.1 RQ1: Motivations 67
3.4.2 RQ2: Needs and Considerations 69
3.4.3 RQ3: Emotional connection and Threat model 72
3.4.4 RQ4: Mental Model 74
3.4.5 RQ5: Perception and Trust 77
3.5 Perspectives of the VPN Providers 78
3.5.1 Key Themes 78
3.5.2 RQ6: Alignment between VPN users and providers . . . 81
3.6 Broader Impact: Actionable Recommendations 83
3.7 Discussion & Conclusion 84

IV. Building VPNalyzer: Systematic, Semi-Automated Investigation of the VPN Ecosystem 86

4.1 Background and Related Work 88
4.2 *VPNalyzer* Design 90
4.2.1 System Architecture and Components 91
4.2.2 Ethics Considerations and Consent 94
4.3 *VPNalyzer* Measurement Test Suite 96
4.3.1 Aspects of Service 97
4.3.2 Misconfigurations and Leakages 99

4.3.3 Security and Privacy Essentials 103
4.4 Data Collection 109
4.5 Findings 111
4.5.1 Support for IPv6 111
4.5.2 Traffic Leakages 112
4.5.3 Security and Privacy Essentials 114
4.5.4 Sharing of VPN Infrastructure 115
4.5.5 Deceptive and/or Malicious Behavior 116
4.5.6 Use of Public DNS Services 117
4.5.7 Use of DNS proxies and Mitigating DNS Leaks 119
4.5.8 Case Study: Testing Custom Secure Configurations . . . 119
4.6 Discussion 120
4.7 Broader Impact 122
4.8 Conclusion 123

V. Leveraging Cross-Layer Network Latency Measurements to Detect Proxy-Enabled Abuse 124

5.1 Background 127
5.1.1 Architecture of Network Proxies 127
5.1.2 The 0trace Technique 128
5.1.3 The WebSocket API 129
5.2 Method 129
5.2.1 System Architecture and Assumptions 130
5.2.2 Determining the Application-layer RTT 132
5.2.3 Determining the Transport-layer RTT 133
5.2.4 Determining the Network-layer RTT 134
5.2.5 Implementation and Deployment 136
5.2.6 Confirming the Use of Proxies 137
5.3 Building Block Evaluation 138
5.3.1 Reliability of WebSocket Pings 138
5.3.2 Reliability of TCP Handshake RTT 138
5.3.3 Reliability of 0trace Pings 140
5.4 Evaluating CalcuLatency in Practice 141
5.4.1 Control Testbed Evaluation 142
5.4.2 Real-world Crowdsourced Evaluation 146
5.4.3 Results Combining the two Evaluations 148
5.5 Ethics 150
5.6 Limitations 150
5.7 Related Work 151
5.7.1 Latency Measurements 151
5.7.2 Internet Liveness 152
5.7.3 Proxy Detection 153
5.8 Discussion and Conclusion 154

VI. Conclusion 156

APPENDICES 158

A. Appendix for Emerging Censorship Techniques: A Case Study of Russia 159

A.1 Validating the Russian Blocklist 159
A.2 Analysis of $RUBL$ 160
A.3 Blockpages observed through DNS Poisoning 161
A.3.1 $RUBL_{sub}$ Measurements 161

B. Appendix for Empirical Understanding of VPN users and VPN providers 164

B.1 Difficulty in VPN discovery (RQ1) 164
B.2 Emotional connection with VPN for different user expertise (RQ3) 165
B.3 Codes from qualitative survey responses 166
B.3.1 Reasons for use 166
B.3.2 Other Resources Used 167
B.3.3 Feeling of Safety without VPN: Limited expertise users . 167
B.3.4 High-expertise users response to mental model 168
B.4 User Survey 168
B.5 VPN Provider Interview Questionnaire 175

C. Appendix for VPNalyzer 178

C.1 Recommendation Websites 178
C.2 Full Results 179

BIBLIOGRAPHY 181

LIST OF FIGURES

Figure

2.1 **Decision graph for detecting DNS and Keyword Based blocking—** Four requests are issued: using the domain resolved from a local DNS resolver, using the domain and control IP resolved from every control vantage point, using just the IP resolved in Russia, and using just the IP resolved in controls. We decide whether the request is blocked based on whether the HTML response matches the blockpage regular expressions. . 28

2.2 **Evolution of the blocklist over 7 years**—The blocklist has grown rapidly for much of its existence, across all categories of contents. 38

2.3 **Testing $RUBL_{dom}$ from all vantage points**—The kind of blocking varies between vantage points. VPSes in Data Centers see varying levels of blocking. Residential Probes experience a larger amount of domains blocking, and they also typically receive explicit blockpages. 40

2.4 **Answers from DNS resolutions that do not match answers from any control DNS resolutions or Satellite resolutions—**Three vantage points VPS-6, Probe-9 and Probe-14) show signs of DNS manipulation. 41

2.5 **Fraction of domains blocked at the individual vantage point as well as AS (aggregated) level**—There are some vantage points and ASes that only block little content, while others block comparatively many more domains. The similarity between the lines shows that blocking is happening at the AS level. Our measurements using Satellite observed much more interference compared to Quack measurements. 42

2.6 **Pairwise Jaccard similarity of domains blocked in remote measurements**—As in the direct measurements, we observe some similarity between domains blocked in remote measurements (Satellite on the left, Quack on the right) either due to high blocking or vantage points in the same ISP. 44

2.7 **Blocking by method when testing $RUBL_{ip}$ on VPSes and residential probes**—Data center vantage points observe much higher IP blocking compared to residential probes, where domain blocking is more popular. . 45

3.1 Example of dark pattern–using countdown timers. 52

3.2 Overall statistics of the users' security and privacy expertise, their VPN subscription type, and VPN usage. 66

3.3 Importance levels users attach with criteria they look for in a VPN, presented along with number of users who chose it. Ranked from 1-most important to 7-least. 69
3.4 Trustworthiness of each resource as rated by users with different security and privacy expertise. Bars are No Opinion, Not-, Moderately- and Extremely-Trustworthy. 71
3.5 Entities from whom users with different security and privacy expertise want to protect their online activity. 73
3.6 Users indicate their concern levels towards VPN-related issues, with the number of users who answered each. 77
3.7 Users indicate the importance of trust-increasing efforts by VPNs, with number of users who answered each. 78
4.1 ***VPNalyzer* Architecture**— (1) User downloads application. (2) User installs the application, reviews our privacy policy, consents to be part of study. (3) User runs an “experiment” consisting of three stages: ensuring VPN is disabled and either granting or denying administrative privileges, enabling VPN and running *VPN case*, and disabling VPN and running *ISP case* (4) Once experiment is done, the application seeks explicit consent from user to upload experiment data to Google Cloud Storage. (5) Analysis pipeline works on the uploaded data. (6) Extracted results appear on website front-end. (7) User visits unique link pertaining to their “experiment” to view detailed results. 91
4.2 ***VPNalyzer* Application**—The homepage (left) contains the user’s current AS Name, public IP address, and geolocation detected using the public IP. The results page (right) contains a link to detailed results, and a summary displayed upon completion of an experiment. 92
4.3 **VPN Kill Switch**— The top image depicts the scenario where the VPN kill switch feature is disabled, and the bottom image shows the kill switch feature enabled and working. Our measurement induces a tunnel failure by blocking all traffic not on our “allowlist”. When the VPN kill switch is disabled (top), the traffic to the allowlist hosts is allowed to pass through the ISP link. In the bottom case, the traffic to the allowlist is also blocked due to the VPN’s kill switch feature. 100
4.4 **Summary of Results**—The number of providers with intersecting findings (as indicated by the blue circles) are illustrated. S&P Essentials includes a positive result for DNSSEC, Qmin, RPKI validation, and supports DoH. 112
4.5 **Use of Public DNS Services**—List of 29 VPN providers that use popular public DNS services. Free VPNs are colored in green, paid in blue, and self-hosted in orange. 118

5.1 **Architecture of Network Proxies**—We distinguish between three types of proxies that differ based on the layer at which they terminate client connections. Clients connecting to application-layer (5.1a) and transport-layer proxies (5.1b) first establish a TCP connection with the proxy and initiate the proxying using protocol-specific messages. The proxies create new TCP connections to the web server, and then establish the tunnel. Whereas, network layer (5.1c) proxies authenticate the client and establish a tunnel, after which all packets including TCP SYN packets are encapsulated and proxied through the server. 128

5.2 **Measurement Setup**—User connects to the Web server either via a VPN or Proxy (top) or directly (bottom). We illustrate the measurements done in both cases. The network layer measurements only reach the proxy server in case the user is using a proxy, but reaches all the way to the user's public IP if they are connecting directly. 130

5.3 **Measurement Flow**—In this example, the client uses a SOCKSv5 proxy. The service provider determines four round-trip times (on three layers) to infer what kind of proxy the client uses. 132

5.4 **Building block evaluation of different techniques used in CalcuLatency** 136

5.5 Differences between 0trace-determined RTT and RIPE Atlas-determined RTT. 139

5.6 **Decision Tree—**We create a decision tree that uses the round-trip times measured using our four different techniques. Based on their results, we use a combination of these values to decide whether a particular experiment can be labeled as a possible proxy connection. The two yellow nodes refer to cases that lack some of the measurement results and are hence, labeled "best effort" calculations. 142

5.7 **ECDF of RTT difference for the 891 control, testbed evaluation measurements**—From our measurements, we find that over 98% of direct measurements and less than 11% of VPN measurements have an RTT difference below 50 ms (shaded in yellow). 144

5.8 **ECDF of RTT difference for the 283 public crowdsourced evaluation measurements**—From our collected measurements, we find that over 98% of direct measurements and 36.1% of VPN measurements have an RTT difference below 50 ms (shaded in yellow). 146

5.9 **CDF of the distance between user and VPN when RTT difference < 50ms**—Majority of VPN experiments with a low RTT difference (below 50ms) are located close to the user. 66.2% of these experiments, the proxy is only upto 650 miles away from the user (shaded in yellow), and 89% of these experiments, the proxy is less than 1000 miles from the user (shaded in orange). 147

5.10 **ECDF of RTT difference of 1127 reliable measurements**—This analysis only contains measurements that did not have a "best effort" calculation label from our decision tree. 100% of all direct measurements and 86.2% of all VPN measurements have an RTT difference below 50 ms (shaded in yellow). 149

A.1 **Blocklist volatility over 7 years**—The two subfigures shows the volatility of the blocklist, with many spikes and downturns in response to real world events. 161
A.2 **Blocking per subnet when testing $RUBL_{sub}$ on VPSes and Probes**— Datacenter vantage points observe a large percentage of blocking in almost all subnets. Residential vantage point comparatively block intensively in fewer subnets. 162
A.3 **Three example blockpages.** 163

LIST OF TABLES

Table

2.1 **Vantage Point Characteristics** 25

2.2 **Top ten countries hosting IPs on the blocklist.** 33

2.3 **Top five TLDs and CDNs for domains in the blocklist**—.com and .ru are the most popular TLDs. 34

2.4 **Categories of responsive domains obtained using topic modeling**—The second column shows the number of documents in primarily Russian or related Cyrillic languages classified into each category, and the third column shows the same for primarily English language documents. Gambling and pornography websites dominate the blocklist. 37

2.5 **Number of subnets completely blocked by vantage points**—VPS 2 and Probe 2 block almost all of the subnets in $RUBL_{sub}$ completely, while others moderately block subnets. 45

3.1 Six themes with their definitions. 58

3.2 Demographics of the (n=1252) survey respondents. 65

3.3 Number and % of users with different subscription types and their feeling of safety without and with a VPN (from VS-Very Safe to VU-Very Unsafe). ↑ indicates more likely than the other subgroups for that column. 72

3.4 Number and % of users who indicate the types of data they think VPN providers collect. Users can choose multiple options, and we exclude users who chose "not sure" (NS) from the other counts. 76

4.1 **Measurements**—The goal of the 15 measurements performed by *VPNalyzer* and the custom backend and public services used for each measurement. 95

4.2 **VPN list**—Names and types of all 80 tested VPN providers. The 39 VPN providers included in our custom "secure" configurations case study are marked with an asterisk. 110

4.3 **Providers with traffic leakages**—26 providers leak traffic during tunnel failure. * indicates those with traffic leaks even when their "kill switch" feature is enabled, see §4.5.8. 113

4.4 **VPN Providers with shared infrastructure**—IP blocks shared by different providers in AS 9009 and 60068. 116

A.1 **Zapret-supplied blacklists' similarity to anonymously-supplied blacklists signed by Roskomnadzor, using the Jaccard index for each category as mentioned in the column name.** 160

B.1 Number and % of users from different user groups indicate how difficult it was decide on a VPN to use (from VD-Very Difficult to VE-Very Easy). Symbols indicate ↑ more, and ↓ less likely than the other rows in the column. 164

B.2 Number and % of users with different security and privacy expertise and their feeling of safety when browsing without and with a VPN (from VS-Very Safe to VU-Very Unsafe). Symbols indicate ↑ more, and ↓ less likely than the other rows in the column. Highlighted values indicate that they contribute to the relevant percentage. 166

C.1 **Websites Used**—The top 25 websites used to create a set of 34 of the most popular VPN providers. Buffered VPN was among the 34 but is currently non-operational. 178

C.2 **VPN Providers & Results**—A red circle indicates "false", and a green square indicates "true" for each test mentioned in the column. A yellow triangle denotes an inconclusive result. Under leaks during tunnel failure, findings also observed in the secure mode (§4.5.8) are marked with two asterisks**. For RPKI validation, "true" implies at least one experiment of the provider shows evidence that RPKI validation is enabled. 180

LIST OF APPENDICES

Appendix

A. Appendix for Emerging Censorship Techniques: A Case Study of Russia . . 159

B. Appendix for Empirical Understanding of VPN users and VPN providers . . 164

C. Appendix for VPNalyzer 178

ABSTRACT

Global events affect the Internet in new ways every day, be it through increased blocking during sensitive events or through user tracking, surveillance, and targeting. Governments, service providers, advertisers, and online threat actors, often enabled by powerful deep packet inspection technology, insulate users from critical information, invade users' privacy, and monitor and tamper with users' traffic. As a result, users are increasingly turning to Virtual Private Networks (VPNs) as a panacea to overcome various security, privacy, and information restrictions, thereby fueling the growth of the commercial VPN ecosystem into a multi-billion dollar industry. Nevertheless, understanding how users discover, use, and interact with VPNs, as well as investigations into the efficacy, security, and privacy provided by such critical tools, remain severely understudied.

This dissertation will demonstrate analyses of threats to user privacy by exploring how commoditized deep packet inspection technologies allow network operators to implement Internet restrictions. It will advance the understanding of key stakeholders of the commercial VPN ecosystem: VPN users and providers. By studying them in tandem, it will illuminate their needs, motivations, and incentives, and use that to highlight misalignments and key areas of concern. Next, it will demonstrate an in-depth technical investigation of VPN products by developing a scalable, rigorous system to test them from a security and privacy standpoint. Finally, this dissertation will explore a service provider's perspective on the malicious misuse of VPNs and present a usable, privacy-focused solution that uses minimal connection features to detect and deter such abuse.

CHAPTER I

Introduction

Information restriction, tracking, and online surveillance are reminders both of the fragility of the Internet and its strength in shaping users' experiences online. Users face security and privacy threats not only from governments, but also from Internet service providers, advertisers, and online threat actors who seek to disrupt, tamper with, and monitor their Internet traffic [237, 114, 219, 52]. Reports of escalating censorship and high-profile security incidents such as data breaches have all fueled a collective public awareness of online risks and restrictions [133, 28, 209, 261].

More Internet service providers (ISPs) have started using deep packet inspection (DPI) technology to examine the traffic that passes through their networks for myriad reasons including implementing censorship, tracking users, monetizing user traffic, practicing traffic differentiation, and even surveillance. With the emergence of commoditized censorship and surveillance technology, it is becoming cheaper and easier for ISPs to exercise finer-grained control and implement more such blocks.

Subsequently, the use of virtual private networks (VPNs) has been growing rapidly, not only among activists and journalists but also among average users [13, 26, 74, 127]. But despite being a growing multi-billion dollar [151] industry, various stakeholders of the VPN ecosystem have remained severely understudied. For instance, previous work has not explored the motivations, needs, and incentives of both the *users* and *VPN providers* together and previous technical evaluations of *VPN product* security [89, 103, 69] have been limited in the scale and types of VPN products analyzed and have used inconsistent heuristics that prevent monitoring of issues in the VPN ecosystem over time. Finally, from the *serverside* perspective, we see that providers who are subject to the malicious misuse of privacy-enhancing tools such as VPNs resort to surveillance of client traffic, or black-box IP reputation services to protect themselves. We do not yet have the answer to whether service providers can build a system using minimum connection features, such as latency, to infer VPN or proxy use, without jeopardizing user privacy or the need for data collection.

On a technical and commercial level, the VPN ecosystem is extremely dynamic making it complex and challenging to study, with constant changes in the features offered and new providers entering the market. Commercial VPN providers can make use of the available VPN protocols such as OpenVPN, L2TP, IPSec, IKEv2, and Wireguard [164, 222, 101, 45], or develop proprietary protocols. The providers also offer different subscription models: paid/premium services, free-to-use services, and freemium models that offer limited free features and charge for premium features and services. This flux in the commercial VPN ecosystem indicates that a large-scale and continuous empirical assessment of the VPN ecosystem requires methods that are both rigorous and scale easily across many providers and be repeated across time.

Apart from identifying issues within the technical implementations of VPNs, understanding key aspects of two important stakeholders—VPN users and VPN providers—has been lacking as well. For instance, is the popularity of VPNs grounded in an understanding of risks on the users' part? Is the rise of VPNs due to dwindling trust in Internet service providers? What benefits do users perceive to gain? What sort of use cases do people have that necessitate a VPN? Users using VPNs are essentially transferring trust from their network provider onto the VPN provider, but it is unclear as to what VPN features encourage them to make this shift. On the other hand, the VPN industry has been known to employ various marketing tactics [7] and dark patterns around discounts [227, 126], but it is yet unknown if these practices are bound to have any significant effect on VPN users. Moreover, the community has not yet understood VPN providers' incentives in sustaining such dark patterns, nor do we know what efforts they take to foster user confidence in an ecosystem plagued with mistrust. To gain a clearer picture of the inner workings of such a large consumer ecosystem, it is imperative to study both its users and its providers.

However, as with other privacy-enhancing tools, VPN tools also can be misappropriated by malicious actors to conduct fraudulent activities [235, 81]. To that end, security and privacy advocates must also ensure that service providers have specific, networking-based methods to detect malicious access facilitated by VPNs rather than enact more heavy-handed protocol-level bans or make use of black-box IP reputation systems. It is also important to ensure such serverside detection methods preserve users' privacy while still employing abuse-prevention techniques that misuse privacy-enhancing tools such as VPNs and proxies.

This dissertation will demonstrate in-depth analyses of privacy harms caused by commoditized DPI technologies, and present analyses and investigations of various stakeholders and components of the commercial VPN ecosystem: VPN users, providers, and serverside operators. First, it will explore the dangers to online freedom and the use of VPNs caused by deep packet inspection technologies. Then, it will advance the understanding of VPN users by empirically studying them through quantitative and qualitative methods. It will also augment this study with the VPN developers' perspectives to holistically identify key areas of concern and bridge gaps in the ecosystem. Next, it will provide an examination of VPN tools from a security and privacy standpoint by presenting a scalable system that brings rigor, systematization, and automation to the testing of existing VPN products. Finally, this dissertation will explore a service provider's perspectives on privacy-enhancing tools, such as VPNs and proxies, being misused to conduct fraud. It will propose a solution that leverages network latencies to detect the use of remote proxy servers for malicious purposes. Specifically, my work will create knowledge that can empower users, researchers, security and privacy advocates, and consumer protection agencies around user privacy and the commercial VPN ecosystem. It will help bring transparency and accountability to companies providing these privacy-enhancing technologies and promote their safe use.

1.1 Thesis Statement and Contributions

Systematic investigation of the efficacy of critical security and privacy-enhancing technologies such as VPNs, combined with empirical knowledge about how users find, interact with, and use these tools is key to improving and securing the ecosystem.

This dissertation will first explore **how commoditized deep packet inspection technologies proliferate privacy harms and access restrictions**, through the case study of Internet censorship in Russia. I present a study on the decentralized nature of the information control strategies we observe in Russia. Through this work, we show how the Russian government is able to leverage the commoditization of deep packet inspection (DPI) technology and enact controls via law and policy compelling network operators to comply. Our work demonstrates how DPIs can easily enable any network operator to enact privacy-violating controls such as censorship, and argues that this model could easily extend to traffic differentiation, fingerprinting traffic, and surveillance.

Next, I conduct **empirical studies of VPN users and VPN providers** using quantitative and qualitative methods to learn the key stakeholders' motivations, needs, and incentives. We augment the large-scale survey of 1,252 VPN users with nine technology developers' perspectives to holistically identify key areas of concern and bridge gaps in these ecosystems. Through this study, we put forth actionable recommendations for the Internet freedom community, security and privacy advocates, and consumer protection agencies.

Concurrently, I **investigate leading privacy-enhancing VPN tools products from a security and privacy standpoint**, by building a systematic, rigorous, and semi-automated tool to conduct testing of existing tunneling VPN products. This investigation into 80 desktop VPNs using our tool reveals several novel findings such as evidence of traffic leaks during tunnel failure, IPv6 leaks, and more implementation shortcomings. We filed 29 responsible disclosures with the concerned VPN providers. We also partnered with Consumer Reports who used our methods and our tool to create data-driven recommendations for millions of users.

Finally, I conduct a **study from serverside operators' perspectives and put forth a networking-based solution to detect the misuse of VPNs for fraud**. As with other privacy-enhancing tools, VPNs are also increasingly misappropriated by malicious actors to conduct large-scale fraudulent activities [235, 81], whereas a majority of legitimate users from our prior study state that they use VPNs for security and privacy. However, to combat this rise in abuse-related activities, service providers employ privacy-invasive, black-box techniques to detect and stop these malicious actors. In this work, I develop a solution that can provide a reliable signal to detect potential remote proxy traffic by leveraging cross-layer network latency measurement techniques. Through this work done in collaboration with Brave Software—a company that provides user-privacy-focused online services including a browser with over 57.27 million monthly users—I provide an open-source technique that can be readily deployed as a software service that can support anti-abuse efforts while also prioritizing legitimate user interests and their privacy. This technique will soon be deployed into production to add to Brave Software's suite of anti-fraud, abuse-prevention techniques.

1.2 Proliferating Deep Packet Inspection Technologies and Privacy Harms

In Chapter II, I present a study of how deep packet inspection (DPI) technology has helped proliferate censorship and enables almost any Internet service provider to disrupt, monitor, and tamper with user traffic. As a case study, we investigate the Russian government's censorship regime. We demonstrate the effectiveness of achieving censorship through law and policy by compelling Internet service providers (ISPs) to enforce censorship using DPI technologies. My team and I worked extensively with activists on the ground and obtained five authoritative lists that contained over 132,000 websites and 329,000 IP addresses that the Russian government requires ISPs to censor. We also obtained vantage points within the country using our activist connections and conducted a large-scale study from networks that cover roughly 65% of the Russian IP address space. We find that residential networks are more likely to serve blockpages with explicit notices to users when censorship is enforced, whereas data center networks are less likely to experience the same level of censorship.

Our findings have implications for existing censorship research, highlighting the need for conducting studies from more residential networks. We illustrated that Russia's censorship architecture and the use of both commercial and home-grown DPI technology could serve as a blueprint and even a forewarning of what national censorship regimes could look like in many countries that have similarly diverse ISP ecosystems to Russia's.

In the years after we published our study, Russia has expanded its capabilities using novel and powerful home-grown deep packet inspection technology called TSPU (технические средства противодействия угрозам, or technical solution for threat countermeasures), which they ensured are installed on the path, typically close to the user to enable centrally coordinated censorship across multiple ISPs. We have studied and documented these expanding capabilities and emerging censorship techniques in separate studies [250, 187].

1.3 Empirical Understanding of VPN users and VPN providers

In Chapter III, I present an empirical study of VPN users and providers. With the commercialization of VPN products, VPN tools have found their way into a regular Internet user's toolbox [106, 203]. As the adoption of security and privacy-enhancing tools such as VPNs increases, it is imperative to develop an empirical understanding of why users use it, their mental models of what a VPN provides them, and their needs and considerations when it comes to choosing a VPN. This understanding is crucial for technologists and security and privacy advocates to address any emerging issues within the dynamic VPN ecosystem.

However, to gain a clearer picture of the inner workings of such a large consumer ecosystem, it is not enough to study its users alone. It is important to also consider VPN providers, and study their motivations and key challenges to bridge gaps in understanding and incentive between users and providers.

We are the first to conduct a multi-perspective study using a quantitative survey of 1,252 VPN users in the U.S. along with qualitative interviews of nine leading VPN providers. We investigate users' motivations, needs, considerations, their mental model of how VPNs work, their threat model, and their perception and trust towards the VPN ecosystem. We augment the results from the user survey with insights we obtained by interviewing nine VPN providers, and highlight the key areas where the two are misaligned.

Our work sheds light on the issues plaguing the consumer VPN ecosystem and presents the following actionable recommendations: prioritizing user education, oversight on advertisements and marketing surrounding VPNs, coordinated efforts to bring attention to the flawed VPN recommendation ecosystem, and regulations to curb malicious marketing tactics that lead to false mental models and false expectations for users. Our work creates knowledge that can help security and privacy advocates such as EFF and CDT, technologists, and VPN providers alike, to call attention to the key areas in the commercial VPN ecosystem.

1.4 Building VPNalyzer: Systematic, Semi-Automated Investigation of the VPN Ecosystem

In Chapter IV, I present the VPNalyzer project where we conduct a systematic, and scalable investigation into the commercial VPN ecosystem. Our VPNalyzer tool brings rigor and automation to the investigation of VPNs, while simultaneously empowering users to conduct their own evaluations of VPN tools. We build a cross-platform tool that has a comprehensive measurement test suite combined with a simple installation and user interface. We implemented a test suite of 15 measurements that includes tests for aspects of service, security and privacy essentials, misconfigurations, and tests for leakages including whether the VPN has implemented an effective mechanism to protect users during tunnel failure.

Using the VPNalyzer tool, we conduct the largest state-of-the-art investigation into desktop VPNs including free and paid VPN providers, as well as self-hosted VPN solutions, and an institutional VPN. Our investigation revealed several novel findings such as evidence of traffic leaks during tunnel failure, IPv6 leaks, and more implementation shortcomings. To ensure our findings get worked upon, we filed 29 responsible disclosures with concerned VPN providers. Our testing methodology and the VPNalyzer tool was used by Consumer Reports as the first in a line of systematic investigation to evaluate a set of popular VPNs. The report resulting from our Consumer Reports partnership was cited by elected officials calling on the FTC to regulate the VPN ecosystem [55].

1.5 Leveraging Cross-Layer Network Latency Measurements to Detect Proxy-Enabled Abuse

In Chapter V, I work with a large service provider, Brave Software, to explore a different perspective of VPN (mis)use. From a serverside perspective, I investigate and present a case of proxy-enabled abuse of novel systems that seek to democratize the ad delivery ecosystem by building services that are privacy-focused and even share ad revenue with users. Balancing abuse-prevention techniques with these rigorous privacy requirements is a hard challenge, especially when the service provider seeks to uphold their privacy-focused business model. In this work, we explore the question of whether we can use minimum connection features, such as latency, to infer VPN or proxy use, without jeopardizing user privacy or the need for data collection. To that end, we present CalcuLatency, a system that leverages cross-layer network latency measurement techniques to differentiate users who are using a remote, long-distance VPN, or proxy from those who are not. We evaluate this system by conducting a two-pronged evaluation: from a controlled testbed using various commercial VPNs, browsers, and user locations, and a larger-scale real-world, crowdsourced evaluation with users participating from over 37 countries from all (six) continents. We provide an open-source technique that can be readily deployed to support anti-abuse efforts while simultaneously prioritizing user privacy.

CHAPTER II

Proliferating Deep Packet Inspection Technologies and Privacy Harms[1]

Network control has long been a goal of nation-states, and the technology to enable that control is cheaper and easier to use than ever. As more citizens of the world begin to use the Internet and social media, and political tensions begin to run high, countries have started to find tools to exert control over the Internet. Recent years have seen many unsophisticated attempts such as Internet shutdowns which, due to their relative ease of execution, have become the *de facto* censorship method of choice in some countries [244, 39, 93]. While some preliminary studies investigating information control have examined India [252], Thailand [60], Portugal [175, 176], and other countries, there has yet to be an in-depth multifaceted exploration of the specific tools and mechanisms used by governments for censorship as they evolve over time.

To our knowledge, no in-depth study has been performed to assess the feasibility of real-time, effective, and homogeneous information control in a decentralized network. Such a study would require measurements from diverse vantage points, such as ISP backbones to data centers and last-mile residential networks, among others. Furthermore, the research

[1] This chapter is based on: Reethika Ramesh, Ram Sundara Raman, Matt Bernhard, Victor Ongkowijaya, Leonid Evdokimov, Anne Edmundson, Steven Sprecher, Muhammad Ikram, and Roya Ensafi. 2020. Decentralized Control: A Case Study of Russia. In Proceedings of the Network and Distributed Systems Security (NDSS) Symposium 2020 [186]. This project was supported by the National Science Foundation grant CNS-1755841, and by a Google Faculty Research Award.

also necessitates knowledge of the country in order to determine what topics, like language, religion, or politics, governments are most sensitive to: this makes it challenging to build an exhaustive list of blocked websites. Moreover, even distinguishing between censorship and run-of-the-mill network failures is often difficult, so an insight into the intent of the censor is crucial to establishing which events are censorship events. Finally, determining *who* is actually doing the blocking can be difficult: governments, individual ISPs, and even servers themselves may refuse to serve traffic for a variety of reasons, for instance prioritizing certain customers due to their location [133]. A study examining decentralized information control must account for all of these factors to effectively test the hypothesis of whether decentralized networks can be uniformly censored.

While countries such as India, Thailand, and Portugal are also pursuing decentralized control, the largest and most aggressive country to do so is Russia, which accounts for a sixth of Europe's Internet users [92]. Their censorship regime has grown rapidly over the past decade, with the adoption of policies and laws that facilitate control. We spent a year in continuous discussion with in-country Russian activists who helped us obtain five leaked snapshots of the government's official "blocklist" digitally signed by Roskomnadzor, a primary entity in charge of nationwide Russian Internet censorship. This blocklist contains the list of domains, IPs, and subnets that the Russian authorities have required ISPs to block, and each of its daily iterations since November 1st, 2012. While we have limited historical visibility into how faithfully ISPs applied this blocklist, we can analyze its evolution to understand what the government intended to block through the years.

Our collaboration with activists in Russia also helped us gain access to a diverse set of vantage points in the country, where even renting from reliable Russian virtual private server (VPS) providers requires Russian currency and an in-country phone number and address. From these vantage points, we can perform measurements to provide a clearer picture of Russia's decentralized control—what is blocked, how it is blocked, and how much variation there is from one ISP to another. We performed measurements from within Russia from 20 different vantage points provided to us by volunteer activists, following established ethical practices to reduce risk [228, 258, 47]. We augment the data collected in Russia with two remote measurement tools—Quack and Satellite [23, 229, 170]—expanding our measurements to over a thousand vantage points within Russia and enabling us to validate our local measurements.

From our experiments, we observe that even though not all ISPs block content in similar ways, the volume of websites blocked within residential ISPs is uniformly high. Indicating that coordinated information control in countries with decentralized networks is entirely possible; debunking our initial hypothesis. However, the method by which censorship is effected is largely dependent on their network providers; we observe TCP-layer blocking, application-layer blocking facilitated by deep packet inspection, and DNS manipulation, or a combination of these methods. We also observed that residential ISPs are more likely to inject explicit blockpages, which cite the law and/or Roskomnadzor's registry as they are encouraged to do so by Roskomnadzor's guidelines.

We also observe a difference in quantity and method of blocking between the two network perspectives—residential networks and data center networks. This corroborates the insight that in most countries, residential ISPs are subject to different laws and policies for information control. Therefore, an accurate representative view of censorship is achievable only with measurements from a diverse set of vantage points.

The qualities of Russia's information controls are not restricted to Russia. As Yadav et al. note, India is already attempting to implement a similar censorship regime [252]. The United States [20] and Portugal [175, 176] are both moving away from net neutrality (though not without resistance [154]), and the United Kingdom's legal framework for identifying and restricting content is almost identical to Russia's [225].

The growth of decentralized information control can lead to different ISPs implementing censorship differently, which may contribute to the fragmentation of access to online content for users—even for neighbors who happen to subscribe to different providers. In countries such as China that practice relatively monolithic censorship, circumvention developers can optimize and test tools for use anywhere in the country, and both marketing and word-of-mouth can help users find these effective countermeasures. But in countries such as Russia, decentralized information control adds another layer of complexity: a circumvention tool that works for one user may not work for others. We hope that by highlighting this new trend of moving away from filtering at government-run technical choke points towards legally mandated censorship enforced by private ISPs, we can help inform thinking and future work on other countries pursuing more authoritarian network controls.

In the years after we published this work in 2020, Russia continued to significantly expand its methods and capabilities when it comes to censorship. Their model of censorship also evolved with these changes and became more centralized through the use and deployment of homegrown DPI technologies called *TSPU* (технические средства противодействия угрозам technical solution for threat countermeasures). As later confirmed by a government official, TSPU is a deep packet inspection (DPI) box specifically developed by RDP.RU on Roskomnadzor's orders [220, 8]. The TSPU devices offered them enhanced capabilities to exercise more uniform controls and to test new capabilities.

One such capability was showcased in 2021 during the Russian government's throttling of Twitter. We studied and investigated this event and it captured the first known centrally controlled attempt by the Russian government to use throttling (instead of outright blocking) to put pressure on social media websites. Working with activists in Russia, we detected and measured this throttling, and reverse-engineered the throttler from in-country vantage points. We illustrated that the behavior of these throttlers shows a high degree of coordination across different ISPs, marking a departure from the decentralized model, which suggests that Roskomnadzor is successfully moving towards achieving more centralized control on its decentralized network of thousands of ISPs [250].

Further, we observed an array of disruptions and network changes occurring in the weeks following the events where Russia invaded parts of Ukraine in early 2022, in a major escalation of the longstanding Russo-Ukrainian War [128]. These included numerous sanctions and restrictions: by Russia against its citizens, by Russia against the world, and by foreign actors against Russia. We found evidence of Russia enacting geoblocking of Russian government domains, forbidding access to foreign users. Through our measurements, we also analyze the real-world deployment of certificates issued by Russia's new domestic CA that emerged as a response to Western certificate authorities ceasing issuance of certificates to Russian Top Level Domains (TLDs). This new domestic CA was untrusted in most browsers due to concerns that the new CA does not comply with technical requirements. However, Gosuslugi (an e-government website) advised users to use a browser that already trusts the CA, such as Yandex Browser or Atom, or to install the certificate manually [68, 148]. We observed that both Russian authorities and foreign actors taking advantage of the decentralized nature of the Internet to implement more access restrictions and localized control [187]. This incident highlights that a threatened effect of any future geopolitical conflicts is the splintering of the Internet, leading to vastly different experiences and exposures of information to different users based on many factors including their geolocation. These incidents should serve as a cautionary tale for Internet freedom activists as they illustrate how easily censors and large Internet services may isolate specific regions from the support of the rest of the world.

2.1 Background and Related Work

Early censorship research focused on countries with more centralized information controls, such as China and Iran [73, 12]. However, new measurement techniques and in-depth studies of countries such as India and Pakistan [252, 157] have observed a move towards a decentralized approach to information control, through both technical and political means. Technical advancements are making it easier for regimes to restrict their citizens' freedoms even in countries without a history of centralized restrictive controls. Russia is a prime exemplar of this trend, and we fear that Russia will provide a model that other less-centralized countries can adapt. In this section, we delineate centralized and decentralized control, discuss past censorship research, and delve into how Russian censorship embodies an alarming trend, all of which helps guide our understanding of the mechanisms that enable increased decentralized control.

Centralized control Previous work has shown that censorship within China and Iran follows a very centralized information control scheme [73, 249, 12, 123]. This is made possible by their strict control over the network infrastructure within their respective countries. Countries with centralized control over their network can control information in a highly scalable way, and small perturbations to network reachability can have dramatic effects throughout the country. An example of this is the case in which North Korea's only ISP lost its link with China Unicom, cutting off Internet access in the whole country [172]. Censors like this tend to apply an even mix of censorship methods across the entire networking stack. For instance, China blocks Google's public DNS resolver (8.8.8.8) at the IP layer, Tor relays at the TCP layer [54], poisons many DNS queries [117], and blocks sensitive search terms in HTTP traffic flows [36].

Decentralized control More recently, several countries around the world have been deploying decentralized information control schemes. These countries do not possess control of their networks in the same way as Iran and China do. Rather, their networks mostly consist of autonomously controlled segments owned by commercial or transit ISPs, whose goals may not align with a government regime attempting to restrict information access. Lack of direct ownership by government authorities lowers their ability to unilaterally roll out technical censorship measures, and instead enact controls via law and policy, compelling the network owners to comply. We see control like this in countries such as India [252], Indonesia [64], and the United Kingdom [6], as well as Russia. In each of these cases, governments pass laws requiring ISPs to block content, and ISPs use a variety of disparate censorship methods to achieve this. For instance, Indonesian ISPs heavily rely on DNS manipulation [64], while Indian ISPs use a combination of DNS manipulation, HTTP filtering, and TCP/IP blocking [252]. These factors cause us to worry that restricting the freedom of citizens is now attainable for many countries, and, even worse, that decentralized information control is more difficult to measure systematically and circumvent. Measuring it requires multiple vantage points within the country and multiple detection techniques to provide coverage of ISP blocking policies. Decentralized control also acts as a barrier to circumvention as it makes it difficult for users to discover locally effective tools.

2.1.1 Understanding Censorship Studies

We highlight the common challenges and considerations that drive design decisions in the censorship field, as well as the overview of extant censorship measurement studies and techniques. In this background section, we aim to illustrate how decentralized information control makes it more *difficult* to discover and characterize censorship.

2.1.1.1 Censorship Techniques

On a technical level, network censorship is defined as the deliberate disruption of Internet communication. At the physical layer, a simple form of disruption is to simply "unplug the cable", cutting off all network connectivity. This extreme action has happened on several occasions in a handful of countries. Shutdowns generally are easier to implement for ISPs, but also provoke backlash from customers and impact their business. A recent analysis showed that such disruptions affected 10 countries in sub-Saharan Africa over a combined period of 236 days since 2015, at a cost of at least $235 million [39]. Most studies, including this one, focus on several protocols above the physical layer which are common targets for censorship, we expand on them below and explain common *methods* of interference, protocol and packet features that *trigger* the censor, and the censor *action*.

- *Method: TCP/IP Blocking; Trigger: IP address; Action: Filter request or response*—The censor can disrupt communication to individual services or hosts by blacklisting their IP addresses [5]. This is a particularly common, effective, and cheap way to block access to a server hosting undesired content. It can cause significant collateral damage for innocuous sites that happen to be hosted at the same IP address as a blocked site, e.g. blocking of content delivery networks' (CDN) point of presence [27]. This method has historically been used in countries such as Iran and China to block circumvention proxies such as Tor relays [54, 12].

- *Method: DNS Manipulation; Trigger: Hostname; Action: Filter or modify response*—The censor can observe DNS queries or responses containing a sensitive hostname, decide to either fabricate responses that return DNS error codes such as "host not found", non-routable IP addresses, or the address of a server that likely hosts a blockpage. A blockpage is defined as a notice that explains to the user why the content is unavailable. DNS manipulation enables fine-grained filtering, because simply poisoning the cache of a DNS resolver can be circumvented by using alternate DNS resolvers such as Google's (8.8.8.8).

- *Method: Keyword Based Blocking; Trigger: Keyword, Hostname; Action: Filter or inject*—The censor can inspect and understand the content of the HTTP(S) packets to determine whether it contains censored keywords. The trigger may also be sensitive content in the response or the request other than the hostname. If triggered, it can either drop packets, or inject TCP RSTs or a blockpage. Implementing this form of blocking is challenging, as inspecting traffic at line rate is quite resource-intensive. Naive implementations are trivially defeated; for example, Yadav et al. [252] discovered that merely capitalizing keywords that the censor was looking for entirely circumvented application layer blocking. Some protocols such as HTTPS also defeat naive implementations of application-layer blocking, but more sophisticated blockers may man-in-the-middle each connection and strip the encryption or block based on finding the trigger in the SNI (Server Name Indication) which is transferred in plaintext.

We want to acknowledge that this is a brief overview of the common methods of censorship, and with advancements in traffic filtering technology, sophisticated censors may obtain access to more fine-grained controls to effect censorship.

2.1.1.2 Censorship Measurement Challenges

With the knowledge of how common censorship is implemented, researchers need to tailor measurements to detect most if not all known implementations. There have been numerous other censorship studies that focus on a specific country. Examples of these studies include India [252], Thailand [60], China [29, 73, 249, 260, 79], Iran [12], Pakistan [105, 149], and Syria [24]. While recent work has discussed the political history of Russian's blocking of Telegram [124], our work presents the *first in-depth study of Russia's Internet censorship techniques.*

Effectively measuring censorship requires several components. First and foremost, the *"input list"* of domains or IP addresses being tested can dramatically impact results and effectiveness of any study [161]. Citizen Lab maintains several test lists [109], both general lists of sites that are frequently censored world-wide as well as country-specific lists. Hounsel et al. discusses automatically curating a culture-specific input list by analyzing web pages that are censored in China [79], noting that a lack of an authoritative blocklist can make it difficult to ascertain the intent of the censor and therefore obscure not only why certain sites are censored but also whether measurements of those sites indicate censorship. Further, drawing meaningful conclusions about global censorship and comparing countries is only possible at a category level. But identifying the category of a given website is not a trivial problem. The current state of the art is to use services like Fortiguard [58] but these services often do not work well for websites other than English.

Censorship measurement studies often suffer from the lack of ground truth which is generally used to validate findings. To compensate for this, studies need to establish strong controls from multiple geographically distributed control vantage points. These vantage points need to be in networks that are not influenced by the censorship regime being studied, and by using multiple vantage points we ensure that the controls are free of effects of transient measurement artifacts and noise. These *"control measurements"* are necessary to establish a baseline for the rest of the study.

In order to comprehensively study the extent of censorship in a particular region, we need a set of *"diverse vantage points"* that shed light on a localized view of the network it operates in. The most direct form of measuring Internet censorship involves using data from users or vantage points (machines under the control of the researcher) inside the country of interest [160]. For example, Winter and Lindskog [245] used one vantage point to study Tor reachability in China and Aryan et al. [12] used one vantage point in their study of Iranian censorship. While one or a few vantage points may be sufficient for measuring centralized censorship regimes, decentralized regimes require a diversity of perspectives.

By making requests to sensitive domains or IP addresses, researchers can directly observe responses from censors and this has been useful for in-depth investigation of censorship techniques in specific countries. These techniques—which we refer to as "direct measurements"—are limited in scale, robustness, and reliability. This is in part due to the difficulty in obtaining vantage points and volunteers and further, due to the potential *"ethical burdens"* of connecting to known-censored content on infrastructure that is likely owned by citizens subject to the jurisdiction of the censor being studied.

In recent years, the popularity of remote censorship measurement tools have grown because of their capability to use more vantage points and perform ethical measurements [53, 169, 229, 170, 205]. These tools do not directly control the vantage points they use for measurement, and thus are not useful for in-depth investigatory testing, but perform well for global censorship measurement. Data collected from remote measurement is also highly complementary to direct measurement since they use different techniques and offer different visibility into the network. Together they are able to offer a more complete view of censorship practices.

Due to observed temporal and spatial variability, recent efforts have focused on developing platforms to continuously collect measurement data on global censorship. One successful platform is Tor project's Open Observatory of Network Interference (OONI) [160], which performs an ongoing set of censorship measurements from the vantage points of volunteer participants. Censored Planet [23], another global censorship observatory, performs continuous remote measurements to identify the prevalence of a variety of censorship techniques in real-time, leveraging the techniques discussed in [169, 229, 170, 205].

2.1.1.3 Censorship Measurement Ethical Considerations

It is important to be aware of the ethical considerations censorship studies take to safeguard participants, regardless of whether they have directly participated (e.g. volunteers) or used as remote vantage points (e.g. organizational servers). Volunteers, especially those in less than democratic regimes, face a risk in accessing sensitive websites. In § 2.3 we provide comprehensive guidelines that we followed for this study in the hope that it benefits other researchers interested in performing similar work.

2.1.2 Russian Information Control

So far we have established common mechanisms by which censorship can occur, and challenges in the way of detecting censorship. In this section, we turn our attention to why Russia's censorship regime is such a compelling example of decentralized control, worthy of study. Russia's censorship regime has seen increased activity in the past decade, but recent events have thrust Russia's information controls into the spotlight. In a famous example, Russia's decision in 2017 to block all Telegram traffic had a massive impact on Internet reachability, as the first attempt to censor Telegram simply blocked millions of IP addresses belonging to the CDNs that Telegram was hosted on [124]. The blocking of these IPs resulted in significant collateral damage, with other services hosted on Google and Amazon becoming unreachable [243].

In order to gain insight into the capability of the Russian government to restrict access to content on the Internet within its borders, we began collaborating with activists within Russia. This collaboration was necessary as Russia has a complex regime of government institutions, each of which control one or a few specific topics that ultimately cause sites to be censored. Our interest stems from the fact that the Russian censorship model can be easily adopted by another country with a similar network structure. In fact, as we discuss in § 2.7, other countries such as the United Kingdom already have a censorship regime similar to Russia's (albeit less aggressive). Therefore, we hope that the lessons learned from Russia can help hone future censorship research and meet international regulatory needs to ensure global Internet connectivity.

The rest of this section discusses the specific regulatory and historical characteristics that created Russia's censorship regime. This information helped us shape our research questions, which we present in the following section.

Russian Legal Framework The primary entity in charge of nationwide Russian Internet censorship is called Roskomnadzor (Federal Service for Supervision of Communications, Information Technology, and Mass Media) [198]. Other government bodies may request that Roskomnadzor block sites, often with content directly related to their scope of duty. The full set of illegal subjects are thoroughly documented by a number of normative acts spanning multiple signed federal laws [194].

Roskomnadzor maintains a singular and centralized Internet blocklist,[2] officially called the Registry of Banned Sites. This registry is an implementation of federal law 139-FZ, passed on July 28, 2012. Currently, Roskomnadzor's registry of banned sites is available to the public, although not in its entirety—only singular queries of an IP address or domain are supported, via a web interface protected with a CAPTCHA [194]. Since its creation, the blocklist has grown in size as new laws were passed to enable the censorship of many subject matters.

[2] However, there is anecdotal evidence that ISPs sometimes receive slightly different versions and at least one account of Crimea having its own blocklist altogether [226].

Russian Technical Framework Although Roskomnadzor maintains the central registry of banned sites, they are not behind the technical implementation of censorship in Russia (though they do provide guidelines [199]). Upon the identification of a website with illegal content, Roskomnadzor sends notice to the website's owner and hosting provider. If the illegal content is not removed within three days, the corresponding site is added to Roskomnadzor's registry, and all ISPs across Russia are required to block access to websites in this registry. Therefore, the implementation of censorship falls on Russian ISPs. Complying content owners are able to reinstate access to their websites once violating content has been removed [40]. Notably, the specific method of blocking is not specified, which enables ISPs to implement different censorship mechanisms. ISPs that do not comply with censorship orders sometimes incur fines [211].

While the Russian government itself does not directly censor traffic, it has promulgated some mechanisms for enabling its ISPs to censor traffic. Russia has developed deep packet inspection technology called SORM (System of Operative Search Measures) [178] that it requires ISPs operate in their data centers. The interception boxes themselves are constructed by a variety of commodity manufacturers [178, 230]. While SORM is primarily used for surveillance purposes [212, 213], some ISPs also use it for traffic filtering [230].

Leaked Blocklist While the blocklist used in Russia is not fully available to the public, we obtained a link to the repository that has regular updates dating back 7 years, as well as official copies of the "current" blocklist signed by Roskomnadzor via our work with activists within Russia. We believe this is the first in-depth study of censorship that has been performed on an authoritative blocklist intended to be used for censorship.

2.2 Experiment Design

Our experiments to measure Internet censorship in Russia must consider the following factors (1) What to test?–An input list of sensitive content that censors in Russia are likely to block, (2) Where to test?–A set of vantage points from where we can test reachability to websites in the input list, and (3) How to test?–How can we infer details about censorship implementation? In this section we describe how we designed our experiments based on each of these considerations.

2.2.1 Acquiring the RUssian BLocklist ($RUBL$)

We worked extensively with activists within Russia to identify what websites the Russian government has been concerned about. This investigation resulted in our discovery of a leaked blocklist repository [193] with over 26,000 commits dating back from November, 2012, when Russian Internet censorship was still in its infancy. This GitHub repository, Zapret, is well-known within the "Digital Rights guardians" community and is rumored to represent frequent snapshots of the daily blocklists received by ISPs.

We also obtained 5 different digitally signed samples of the blocklist that were distributed by Roskomnadzor, shared with us from multiple sources. We verified that these leaked blocklists are authorized by *CN*=Роскомнадзор and *CN*=Единая информационная система Роскомнадзора *(RSOC01001)* which translates to Roskomnadzor, and Unified Information System of Roskomnadzor. These blocklists are identical to what Russian ISPs would receive. We then compared these blocklists to the Zapret counterpart's contemporaneous commits to corroborate the validity of the repository data and found that the Jaccard similarity between these lists were greater than 0.99. We furnish more details of this validation in Appendix A.1.

We used the digitally-signed blocklist dated April 24, 2019, which we refer to as $RUBL$, as the input list for all our measurements. A single entry in $RUBL$ contains any combination of IP addresses, IP subnets, domains, and domain masks (wildcards). We have no knowledge of how and when DNS resolution was done, or even if resolution was done at all. If the intent was to block domains, we do not know how the accompanying IP addresses were obtained, and vice versa. We break $RUBL$ into $RUBL_{ip}$, $RUBL_{dom}$, and $RUBL_{sub}$, containing the unique IPs, domains, and subnets respectively, that pass our controls. Since our measurement tools cannot utilize masks, a domain mask `*.domain.com` is replaced with both `domain.com` and `www.domain.com`. In total, $RUBL$ contained 324,695 unique IPs, 132,798 unique domains, and 39 mutually exclusive subnets prior to control measurements which we explain in the following section. While we mainly focus on $RUBL$, we also provide historical analysis of the Zapret repository commits from November 19, 2012, to April 24, 2019 in § 2.5.

2.2.2 Establishing Sound Control Measurements

Prior to running the measurements from Russia, we need to run control tests to remove IP addresses and domains that are not responsive. To that end, we obtained 13 geographically diverse control vantage points outside of Russia: 4 in North America, 4 in Asia, 4 in Europe, and 1 in Australia. To verify responsive domains, we send a HTTP GET request for every domain from every control vantage point using ZGrab [259], an open-source application layer scanner that operates with ZMap [47]. Our ZGrab tests are customized to follow (a maximum of 10) redirects. We also resolve each of the domains from the control vantage point using ZDNS [257], an open-source command-line utility that provides high-speed DNS lookups. If we get a response for both tests on at least one control vantage point, we include it in the final list. This resulted in a list of 98,098 (73.9% of the original list) domains, which we will refer as $RUBL_{dom}$ from hereon. We characterize $RUBL_{dom}$ further in § 2.4.2.

We test the responsiveness of the IPs and subnets in $RUBL_{ip}$ and $RUBL_{sub}$ by making TCP connections to port 80 from each control vantage point using ZMap. If we receive a SYN-ACK from the IP to at least one of our control vantage points, we include it in the rest of our measurements. This resulted in 121,025 IP addresses (37.2% of the original list). For $RUBL_{sub}$, we excluded 8 subnets out of the total 39 subnets as they didn't have *any* responsive IP addresses. In total, 567,848 IP addresses (77.2%) were reachable out of

VP Type	Num. of VPs	Num. of ASes	Num. of ISPs
VPS in Data Centers	6	6	6
Residential Probes	14	13	13
Quack (Echo Servers)	718	208	166
Satellite (Open DNS Resolvers)	357	229	197
Unique Total	1095	408	335

Table 2.1: **Vantage Point Characteristics**

735,232 IP addresses in the expanded subnets. These filtered lists are what we will refer to by $RUBL_{ip}$ and $RUBL_{sub}$, respectively. We characterize them further in § 2.4.1.

2.2.3 Conducting Direct Measurement

2.2.3.1 Obtaining Vantage Points

We perform measurements from diverse vantage points, including *VPSes* in data centers and *Probes* in residential networks. An overview of the characteristics of all our vantage points is shown in Table 2.1. To increase our measurement coverage, we also conduct remote measurements discussed later on in § 2.2.4).

- *VPSes in Data Centers:* With help from activists, we obtained six reliable VPSes confirmed to be hosted in Russian data centers, each in a different ISP. We explored obtaining vantage points from over 35 different providers but many of them observed *no* censorship and some were not conducive to measurement. Renting these machines can only be done with Russian currency and an in-country phone number and address.

- *Residential Probes:* With the insight that different information control policies might apply to residential networks versus data center networks, we also conducted measurements from residential networks. We recruited fourteen participants within Russia to run our probe code (the same that was run at the VPSes, adjusted for lower bandwidth). No information about the participants' network was collected, except for the IP address from which the measurement was performed. To recruit participants, we used the established process of OONI [160] and followed the ethical precautions detailed in § 2.3. We attempted to recruit participants from diverse networks, leading us to cover thirteen ISPs (two of our probes were in the same ISP).

In total, our direct measurement platform consists of 20 vantage points. With remote measurements, discussed in § 2.2.4, we perform measurements from well over 1,000 vantage points. With respect to coverage within Russia, our vantage points are in 408 unique ASes that control ≈65% of Russian IP address space, according to Censys [46].

2.2.3.2 Identifying Censorship Methods

With an established measurement platform and the $RUBL_{dom}$, $RUBL_{ip}$, and $RUBL_{sub}$ lists, we investigate the following: For a given IP address or domain, determine whether it is being blocked; if yes, determine how the blocking is performed. We focus on three common types of blocking: TCP/IP blocking, DNS manipulation, and keyword based blocking based on deep packet inspection. DNS manipulation and keyword based blocking can actuate censorship explicitly by returning a blockpage, or implicitly by forcing a timeout or returning a TCP RST.

Detecting TCP/IP Blocking We use ZMap to attempt a TCP handshake with each IP address in $RUBL_{ip}$ and in the expanded $RUBL_{sub}$ list. Running this test produces a set of IP addresses that successfully responded to our TCP SYN packet with a TCP SYN-ACK packet. Any IP addresses that do not respond are considered to be blocked, since these IP address were responsive in our control measurement phase.

Detecting Resets and Timeouts Some censors, when observing an undesirable keyword, drop the packet that forces the connection to timeout or reset the TCP connection. To detect this, we request each domain in $RUBL_{dom}$ interspersed with benign domains such as `example.com` by locally resolving the domain on the vantage point and attempting a HTTP GET request for the domain. This is to ensure that this behavior is not due to transient network errors. If the tests for the benign domains succeed but $RUBL_{dom}$ domains fail, we classify this as censorship due to resets or timeouts, based on the error type received during our test.

Detecting DNS and Keyword Based blocking More typically when a censored domain is requested, ISPs that employ this method of blocking respond with a blockpage. Detecting blockpages from other unexpected error pages such as server-side blocking errors (e.g. HTTP status code 403), and page not found errors (e.g. status code 404) is not a trivial task. There have been multiple blockpage detection methods proposed in previous work to reduce manual effort [133, 98].

Building on the methodology from Jones et al. [98], our blockpage detection algorithm works as follows: we apply single-link hierarchical agglomerative clustering to HTML web pages to detect blockpages. We extract representative unigrams and bigrams from the clusters under the assumption that pages known from anecdotal sources [19] to contain Russian phrases equivalent to "Access Restricted" and "Roskomnadzor" are usually blockpages, while other sites would not normally contain this kind of language. This is further confirmed by Rozkomnadzor's own recommendations for blockpage content [201].

Using these representative unigrams and bigrams, we manually create regular expressions to match known blockpages. We then validate these regular expressions by grouping pages with the exact same content. We verify that the groups with pages matching the regular expressions contain only blockpages (no false positives). Since ISPs typically return the same blockpage for every censored domain, the *groups* that do not match any regular expressions are not likely to be blockpages, which we manually confirm to eliminate false negatives.

We designed tests that use $RUBL_{dom}$ as the input to characterize DNS and keyword based blocking by employing the decision logic laid out in Figure 2.1. We explain each test and provide a walk through of the flowchart below.

Test 1: For every domain in $RUBL_{dom}$, we send a GET request from all of our vantage points within Russia, allowing the domain to locally resolve. For all responses that did not

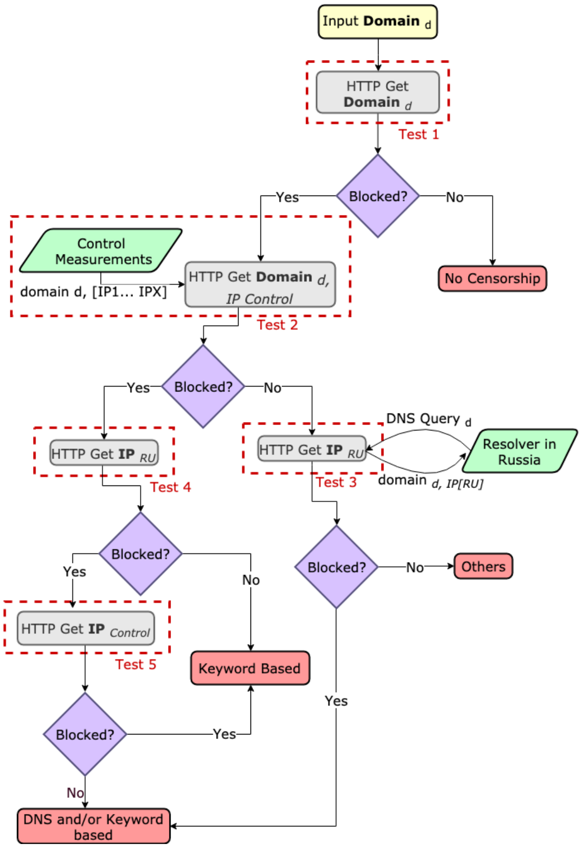


Figure 2.1: **Decision graph for detecting DNS and Keyword Based blocking**—Four requests are issued: using the domain resolved from a local DNS resolver, using the domain and control IP resolved from every control vantage point, using just the IP resolved in Russia, and using just the IP resolved in controls. We decide whether the request is blocked based on whether the HTML response matches the blockpage regular expressions.

contain an error (resets and timeouts categorized and treated separately), we check whether the returned web page matches at least one of the blockpage regular expressions, and if so classify them as "blocked". If this first request is not "blocked", we determine that the domain is not censored. If the request is blocked, we must identify the method of blocking using the results of the following tests.

Test 2: We make another HTTP GET request for the domain, this time using the domain and every unique IP that the domain resolves to in each of the control vantage points. We then pass the web page from the response to our blockpage detection algorithm.

Test 3: If the web page from this Test 2 is not blocked, we look at the result of a GET request for just the IP of the domain resolved in Russia (without the domain name). If the response is classified as blocked from our blockpage detection algorithm, we only know that the domain is either blocked at the application layer by keyword based filtering (if the Russian IP actually points to the site), DNS poisoning (if the Russian IP does not point to the site), or both (if the Russian IP does not point to the site but a blockpage was injected before the connection could reach the poisoned address). If the response is not blocked from this third request, we classify the type of blocking as "Others". Upon investigating what falls under this category, we observed that there are instances where a combination of DNS and TCP/IP blocking is applied, i.e. the actual website is not accessible from the vantage point, even though a blockpage was not received; the reasons may be that the connection was reset or DNS resolution failed every time.

Test 4: If Test 2 is blocked, we look at the result of the GET request with only the IP address resolved from Russia (the same as Test 3), and observe the response. If this is not blocked we can safely conclude that the blocking was only triggered by the presence of the domain name in the request, and thus was blocked at the application layer by keyword based blocking.

Test 5: If Test 4 is blocked, we look at results from the final GET request with only the IP address that was resolved from the control machines. If this request is blocked, we can again definitively declare keyword based blocking, based on some keyword in the *response* from the site also acting as the trigger. If it is not blocked, we can only be certain that it is either DNS manipulation, keyword based blocking, or both.

In cases where we are unable to distinguish keyword based blocking and DNS manipulation we compare the resolved IPs in the Russian vantage points to the resolved IPs in our controls and the answers which are deemed "Not Blocked" in Satellite. The results of this experiment are described in § 2.5.

2.2.4 Conducting Remote Measurement

Our direct measurements provide a high-fidelity, in-depth view of Russian information control, particularly from the data center and residential network perspectives. However, acquiring these vantage points is quite resource intensive, and our measurements are inherently limited by the number of vantage points we can obtain. To complement this data, and to determine whether our direct measurements are representative, we use two remote measurement tools: Satellite [205, 170] and Quack [229]. Remote measurement tools such as Satellite and Quack use the behavior of existing Internet protocols and infrastructure to detect censorship, i.e. researchers do not need to obtain access to vantage points but just interact with remote systems to learn information about the network. Satellite remotely measures DNS manipulation using open DNS resolvers and Quack detects application-layer blocking triggered on HTTP and TLS headers using Echo servers. These remote measurements select only vantage points that are part of organizational or ISP infrastructure, hence providing a complementary perspective to direct measurements.

2.2.4.1 Obtaining Remote Vantage Points

With operational help from the Censored Planet team [23], we used 357 open DNS resolvers in Russia located in 229 different ASes (197 unique ISPs), and 718 Echo servers located in 208 different ASes (166 unique ISPs). As shown in Table 2.1, this increases our coverage considerably. We annotate the vantage point locations with the Maxmind GeoIP2 database [130], and find the AS information through RouteViews data [200].

2.2.4.2 Identifying Censorship

On our behalf, the Censored Planet team performed Satellite and Quack using $RUBL_{dom}$ based on the techniques described in [229] and [205]. Both tools have their own methods to label a domain as being “manipulated” or “blocked”. Satellite creates an array of five metrics to compare the resolved IP against: Matching IP, Matching HTTP content hash, Matching TLS certificate, ASN, and AS Name. If a response fails all of the control metrics, it is classified as blocked. Quack first makes an HTTP-look-alike request to port 7 of the Echo server with a benign domain (`example.com`). If the vantage point correctly echoes the request back, Quack then requests a sensitive domain. Quack makes up to four retries of this request in case none of the requests are successfully echoed back. If the vantage point fails for all 4 requests, Quack tries requesting a benign domain again to check whether the server is still responding correctly. If so, the failure to echo back the sensitive domain is attributed to censorship.

2.3 Ethical Considerations

Censorship measurement studies involving active network measurement raise important ethical considerations. Most censorship measurement studies, including ours, aim to trigger censors from various vantage points which might cause risk of retribution from local authorities. Aiming to set a high ethical standard, we carefully designed our experiments to follow or exceed the best practices described in the Belmont [152] and Menlo [44] reports. Before initiating any of the measurements, we consulted with our university's IRB, who determined that we were exempt from regulation but advised us to discuss with the university's General Counsel, which we did. We vetted the risks of our study and shaped our data collection methods through a year of continuous communication with prominent activists within Russia, with colleagues experienced in censorship and measurement research, and with our university's General Counsel.

Gaining background understanding of the laws of the country is imperative to designing ethical measurements. Prior to engaging with us, our activist collaborators had been actively participating in open-source projects such as OONI and Tor, and had traveled outside of Russia to present details about Russian censorship in international forums. Their guidance was essential for us to ensure we were aware of Russian law and policy regarding accessing censored content. These collaborators facilitated renting VPSes and running measurement from the residential probes.

Our direct measurements involve sending requests for potentially censored content from vantage points inside Russia. This creates a potential risk to participants who own and control these vantage points. We consulted with our activist collaborators, who assured us that even if the anonymized vantage points, data centers, or ISPs are discovered, there has never been any punitive action on the part of the Russian government or others against entities who do not comply with the blocklist. We then begin the process of obtaining informed consent from participants by customizing the OONI consent form which was drafted by the Harvard Cyberlaw Clinic and is attached to our published paper [186]. This form documents in detail the measurements performed and data collected and seeks explicit approval. Before our activist collaborators asked participants to run measurements from residential probes, they used our consent form and drafted an email in Russian to solicit explicit consent from the volunteers, who were recruited from a tech-savvy population already involved with activist groups that advocate for Internet freedom.

We obtained our VPSes from commercial VPS platforms, whose operators understand the risk in offering network and computing services. In collecting the data from our VPS platform, we did not subject anyone in Russia (or elsewhere) to any more risk than they would already incur in the course of operating a VPS service.

Our remote measurements seek only vantage points that are not owned or operated by end users and are part of organizational or ISP infrastructure. As in the case of our VPSes and residential probes, there is a possibility that we place the operators of these remote vantage points at risk. Again, there is no documented case of such an operator being implicated in a crime due to any remote Internet measurement research, but we nonetheless follow best practices to reduce this hypothetical risk. From the list of all available open DNS resolvers in Russia, we identify those that appear to be authoritative nameservers for any domain by performing a reverse DNS PTR lookup and only select those resolvers whose PTR begins with the regular expression "`ns[0-9]+|nameserver[0-9]`". Similarly, we ran Nmap on all the Echo servers in Russia and exclude those whose labels do not indicate an infrastructural machine. Using only infrastructural vantage points decreases the possibility that authorities might interpret our measurements as an attempt by an end-user to access blocked content. Moreover, we initiate the TCP connection and send the sensitive requests, and there is no communication with the actual server where the sensitive domain is hosted. We also set up reverse DNS records, WHOIS records, and a web page served from port 80 on each machine in the networking infrastructure we use to run measurements, all indicating that our hosts were part of an Internet measurement research project.

We also follow the principle of good Internet citizenship and reduce burden on the vantage points by rate limiting our measurements, closing TCP connections, and maintaining only one concurrent connection. Our ZMap and ZGrab scans were conducted following the ethical guidelines proposed by Durumeric et al. [47, 46].

2.4 Data Characterization

The most recent sample of *RUBL* contains 132,798 unique domains and 324,695 unique IP addresses. It also contains a list of 39 subnets ranging from /24s to /16s. This section characterizes both the full *RUBL* blocklist and the final filtered list obtained after running control measurements described in § 2.2.2.

2.4.1 IPs and Subnets

#	Country	IPs	#	Country	IPs
1.	United States	203,107	6.	Russia	6,328
2.	Germany	31,828	7.	Finland	6,057
3.	United Kingdom	25,931	8.	Japan	2,490
4.	Netherlands	16,161	9.	Estonia	2,327
5.	France	8,117	10.	Iran	2,070
Other					19,622
Total					324,038

Table 2.2: **Top ten countries hosting IPs on the blocklist.**

TLD	Domains	CDN	Domains
1. .com	39,274	1. Cloudflare	44,615
2. .ru	11,962	2. App Engine	89
3. .info	5,276	3. Cloudfront	80
4. .net	4,934	4. Incapsula	48
5. .xyz	3,856	5. Akamai	12
	—	In two of the above	47
Others	32,796	No CDN	53,301
Total	98,098	Total	98,098

Table 2.3: **Top five TLDs and CDNs for domains in the blocklist**—.com and .ru are the most popular TLDs.

As mentioned in § 2.2, we examined the responsiveness of the IPs on the blocklist. Only 121,025 IPs on the blocklist (37.3%) were reachable from our controls. Our control measurements were highly concordant; over 99% of IPs that were reachable at some control vantage point were reachable at all control vantage points. The low rate of responsiveness (37.3%) might be the artifact of our measurement, as these IPs might be alive but not responding on port 80, such as proxies configured on custom ports.

For the 324,695 unique IPs in the list, we examined their geolocation using the MaxMind Geolite2 [130] database. 324,038 (99.8%) IPs were found in the database. We saw that over 200k IPs (>61%) were located in the US. Somewhat surprisingly, Russia was only the sixth most popular country in which IPs were located as shown in Table 2.2. These IPs spanned over 2,112 Autonomous Systems (ASes) based on RouteViews lookup.

The blocklist also contains 39 subnets, ranging from /16s to /24s. 31 out of 39 of these subnets contain at least one IP reachable to one of our controls. The remaining eight unreachable subnets geolocated to Moscow.

2.4.2 Domains

For the 132,798 domains in the list, over 49,583 (37.3%) are `.com` domains and 15,259 (11.5%) are `.ru` domains. As discussed in § 2.2, 34,404 (25.9%) domains on the blocklist are not responsive, so for the analysis that follows we only focus on the 98,098 responsive domains. `.com` and `.ru` still dominate responsive domains as shown in Table 2.3.

Inspired by McDonald et al. [133], we looked at what CDNs the sites in the blocklist were hosted in, if any. We were able to identify the CDN for 44,797 (45.7%) domains following their methodology. As shown in Table 2.3, an overwhelming majority of domains which were served by a CDN (99.6%) were hosted on Cloudflare, which provides some of its services for free with little vetting of the sites. 47 domains had signs that they used more than one CDN service. In these cases, we counted them as customers of both.

We initially experimented with using the Fortiguard document classification service [58] to categorize domains and ascertain what types of websites are in the blocklist. Unfortunately, the Fortiguard classification was not effective for Russian language domains. Also, a large number of domains—27,858 (28.4%)—were classified into the "Business" category, which did not reveal much information about the services hosted on those domains. Therefore, we developed our *topic modeling* algorithm designed after the technique introduced in Weinberg et al. [242]. Our topic modeling algorithm processes the text received from control measurements, and uses Latent Dirichlet Allocation (LDA) clustering [18] to identify pages with the same topic. To accomplish this, we adopt the following steps:

- *Text Extraction*—From the control measurements, we obtained the HTML responses for all the 98,098 domains. We first filter out all the responses that returned an empty HTML body, have an error code in the status line, or have encoding issues in the server response. This reduced the number of classifiable domains to 70,390 (71.8% of the original list). We then use Python's `Beautiful Soup` library [15] to extract useful text and remove boilerplate text.

- *Language Identification*—The LDA algorithm requires input documents to be in the same language; as described in [242], it detects semantic relationships between words based on the probability of them occurring together within a document. We used Python's `langdetect` library [112] to identify the primary language for each document. Out of 70,390 classifiable documents, 44,270 (62.9%) primarily contained Russian or related

Cyrillic text, and 19,530 (27.7%) contained primarily English text. We choose to focus on this portion of the classifiable pages as the other 9.4% contained documents in 42 different languages. We thus reduce our manual effort in labeling topics by only using LDA only twice, once for Russian pages and once for English pages.

- *Stemming*—Before applying the LDA algorithm, we reduce all words to stems using Snowball [210]. We then apply term frequency-inverse document frequency (*tf-idf*) [197] to select terms that occur frequently. We preserved terms whose combined *tf-idf* constitutes at least 90% of the total document.

- *LDA analysis*—We then use LDA for Russian and English documents separately. We used Python's `gensim` [61] and `nltk` [153] libraries for our implementation, and we used all documents for training. We found N=20 topics to be optimal, and α is determined optimally by the library based on the training data.

Using LDA, we obtain 20 topic word vectors from the English documents and 20 topic word vectors from the Russian documents. Two researchers independently labeled the topics by reviewing the top words in each topic. Disagreements were resolved through discussion between the researchers. Many topics were given the same label; as discussed in [242], this is one of known limitations of LDA analysis. We manually merge these topics into 9 categories. Additionally, we manually selected a random subset of documents within each topic cluster and ensured that all the documents belonged to the category they were assigned.

The number of English and Russian documents classified into each category is shown in Table 2.4. The majority of domains (67.6%) fall into the "Gambling" category, indicating the stringent crackdown of Russian authorities against gambling websites. Our analysis suggests the high number of gambling websites to be an effect of websites quickly cloning to an alternate mirror domain when added to the blocklist. This can be seen by many of the gambling website domains on the blocklist having slight variations in their names, for example `02012019azino777.ru`, `01122018azino777.ru`, `01042019azino777.ru`,

Category	Num. Russian	Num. English	Total
Gambling	33,097	10,144	43,241
Pornography	5,576	2,821	8,397
Error Page	134	3,923	4,057
News and Political	1,883	No clusters	1,883
Drug Sale	1,811	No clusters	1,811
Circumvention	1,769	No clusters	1,769
Multimedia	No clusters	1,610	1,610
Parking Page	No clusters	601	601
Configuration Page	No clusters	431	431
Categorized Total	44,270	19,530	63,980
Other Language Pages	—	—	10,464
No HTML or Error	—	—	23,654
Total			98,098

Table 2.4: **Categories of responsive domains obtained using topic modeling**—The second column shows the number of documents in primarily Russian or related Cyrillic languages classified into each category, and the third column shows the same for primarily English language documents. Gambling and pornography websites dominate the blocklist.

and so on. This also suggests that the blocklist is not actively maintained. Unsurprisingly, pornography websites also feature prominently in the blocklist.

$RUBL_{dom}$ contains news, political, and circumvention websites that feature exclusively Russian-language media (`chechenews.com`, `graniru.org`) and activist websites such as `antikor.com.ua`, which is a self-proclaimed national anti-corruption portal. Some of the pages were also categorized into error pages, parking pages and configuration pages, indicating that these domain owners have moved since being added to the blocklist. These pages are primarily in English because they use templates from popular web server error pages (e.g. Apache, Nginx etc.)

There are a few *caveats* to our topic modelling algorithm. First, the documents we determine as Russian and English may contain text in other languages, but we only choose those documents that are predominantly in either Russian or English. Nevertheless, a significant amount of other language text may lead to miscategorization of some websites. Second, our labeling is primarily based on the top words in each word vector. This may also lead to some pages being categorized incorrectly, but our manual verification did not find any false positives.

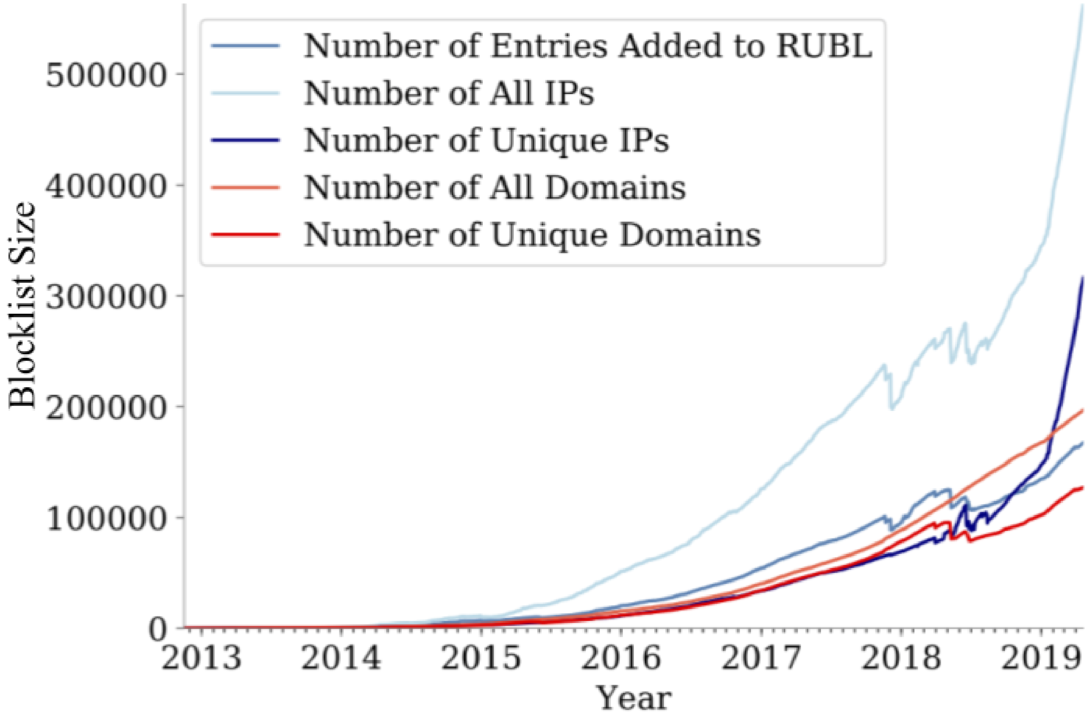


Figure 2.2: **Evolution of the blocklist over 7 years**—The blocklist has grown rapidly for much of its existence, across all categories of contents.

2.5 Results

We divide this section into four parts: first, we begin with an analysis of the Zapret repository and present data about how it has evolved over time. Then we present results from $RUBL_{dom}$, $RUBL_{ip}$, and finally, $RUBL_{sub}$ measurements.

2.5.1 Historical Analysis of Russian Blocklist

We analyze the Russian blocklist's evolution over a seven-year time period, from November 19, 2012, to April 24, 2019 at a daily granularity. Since it may be updated multiple times a day, we utilize only the latest version, which is most often published close to midnight. Any activity of smaller granularity, such as the occasion of an addition or removal of an IP address in a time span of less than 24 hours, is not considered. IP subnets are not included in this analysis, which amount to an approximately additional 26,000 addresses beginning in the middle of 2017 and 16 million addresses beginning in April 2018 due to the banning of Telegram. These addresses are omitted because their inclusion obscures graph clarity due to their significantly greater scale.

As shown by Figure 2.2, *the size of the blocklist appears to have grown rapidly since its conception in 2012.* The plot shows three size metrics: number of entries, raw number of both IPs and domains, and number of unique IPs and domains. Each of these metrics is cumulative and the drops in the number are due to “removal” of entries, IPs, or domains. Since an entry may contain multiple IPs and domains, the number of IPs and domains far exceeds the number of entries.

An unexpected finding is how the raw number of IPs significantly exceeds the number of unique IPs. This discrepancy can be attributed to potentially unintentional duplication—one IP added to the blocklist because it hosts one domain name may later be entered again for a different domain. Multiple domains may share IPs because of the prevalence of sites hosted on CDNs in the blocklist (as discussed in § 2.4.2). More details on this analysis can be found in Appendix A.2.

One important observation is the sharp increase in the number of raw IPs, *unique* IPs, and a moderate increase in the number of unique domains in the past year. This suggests that there is a deliberate effort to increase the accuracy of the list. This is further punctuated by a number of drops in all the metrics in the past year, which suggests that there has been conscious effort put into making the list more meaningful and to avoid repetitions.

2.5.2 Characterizing Censorship of $RUBL_{dom}$

As described in § 2.2, we have six VPSes in data centers and 14 residential probes. Figure 2.3 shows the type of censorship observed at each vantage point. We divide the rest of this section by vantage point type, in order to highlight the complementary nature of the results from each of them.

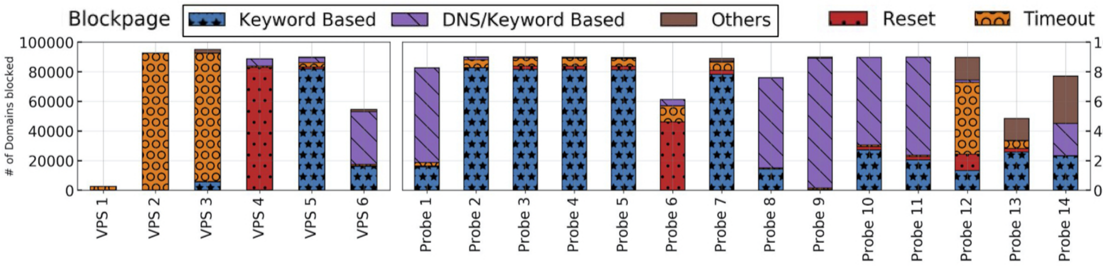


Figure 2.3: **Testing $RUBL_{dom}$ from all vantage points**—The kind of blocking varies between vantage points. VPSes in Data Centers see varying levels of blocking. Residential Probes experience a larger amount of domains blocking, and they also typically receive explicit blockpages.

2.5.2.1 VPSes in data centers

We observed some amount of censorship at all of our VPSes in data centers. The number of domains blocked per vantage point is shown in Figure 2.3. Four out of six VPSes show that more than 90% of $RUBL_{dom}$ is blocked, with the highest blocking 96.8% of all $RUBL_{dom}$ domains.

The censorship method varies between each VPS, confirming our hypothesis that *the lack of prescription of censorship mechanism enables data center network providers to employ any method of censorship.* While most VPSes observe multiple kinds of blocking, one method of blocking typically dominates at each vantage point. For example, VPS 5 and VPS 6 mostly observed blockpages, while VPS 2 and VPS 3 observed more connection timeouts. In VPS 4, we observed that TCP connections were reset when domains in $RUBL_{dom}$ were requested. We suspect that VPSes observe more than one type of blocking due to content being blocked at different locations along the path to the server, such as at transit ISPs. Content restriction at transit ISPs would cause most content to be blocked across the country, even if ISPs closer to the user do not censor all content in $RUBL$.

2.5.2.2 Residential Probes

Figure 2.3 shows that residential probes show higher amounts of blocking overall, suggesting that ISPs closer to the user block almost all the domains more uniformly. Nine

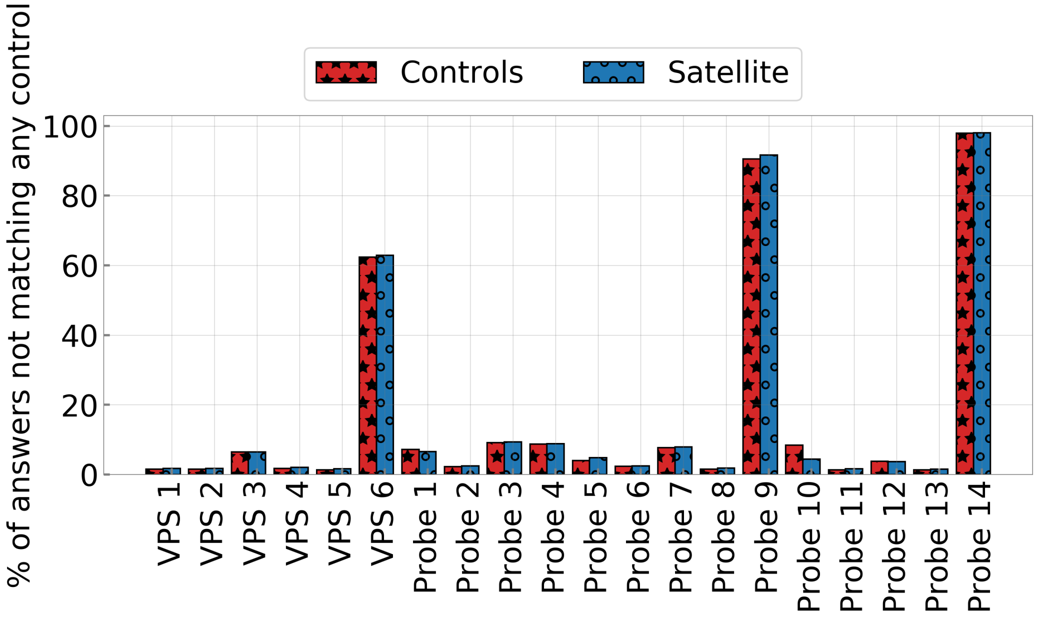


Figure 2.4: **Answers from DNS resolutions that do not match answers from any control DNS resolutions or Satellite resolutions—**Three vantage points VPS-6, Probe-9 and Probe-14) show signs of DNS manipulation.

out of 14 residential probes observe more than 90% of the domains blocked and all of the probes observe at least 49% of the domains blocked.

While VPSes saw high occurrences of timeouts and resets, most residential probes observed a blockpage. We believe this is in part due to the fact that residential ISPs are encouraged by Roskomnadzor's guidelines [199] to cite the law and/or Roskomnadzor's registry and provide explicit information regarding blocking to users. As for the other methods of blocking, we found that Probe 6 predominantly observed a large amount of connection resets and Probe 12 observed a large number of timeouts.

As mentioned earlier, a blockpage is shown to the user when the blocking method is either "Keyword Based" or "DNS/Keyword Based". In the latter the trigger is the hostname but the method of blocking is not clear. In an effort to distinguish between the two methods of blocking, we compare the IPs from domain resolution in the residential probes with the IPs received in domain resolution from all control vantage points and with the answers that were determined as "Not manipulated" in Satellite. The percentage of IPs from each

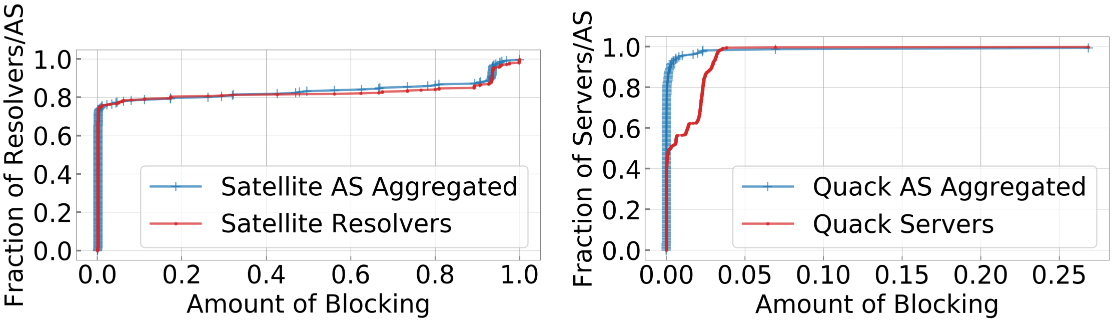


Figure 2.5: **Fraction of domains blocked at the individual vantage point as well as AS (aggregated) level**—There are some vantage points and ASes that only block little content, while others block comparatively many more domains. The similarity between the lines shows that blocking is happening at the AS level. Our measurements using Satellite observed much more interference compared to Quack measurements.

vantage point that does not match any control IP or any resolved IP in Satellite is shown in Figure 2.4. VPS 6, Probe 9 and Probe 14 observe a large percentage of resolved IPs that do not match any of the control responses. This lends credence to the hypothesis that these three vantage points may be subject to DNS manipulation rather than keyword based blocking. To corroborate this, we investigate all instances of "DNS/Keyword Based" blocking and found that *each of the three vantage points observed a single poisoned IP respectively*. We looked at the content hosted at these three IPs and found a blockpage being returned which can be seen in Figure A.3 in the Appendix.

We observe blockpages in that was categorized as "Other" specifically in Probes 12, 13, and 14, meaning we could not exactly determine the *method* of blocking. Upon investigation, we saw that Probe 14 received a blockpage when queried with the domain but was unable to retrieve the page when queried with the IP received from control. Considering Probe 14 also sees high IP blocking as shown in Figure 2.7, we believe Probe 14 observes a combination of DNS and IP blocking. Similarly for Probes 12 and 13, we observe behavior consistent with Keyword Based blocking but the blockpage was unable to load in some cases.

2.5.2.3 Remote Measurements

We conduct remote measurements for $RUBL_{dom}$ using 357 vantage points for Satellite and 718 for Quack. The CDFs in Figure 2.5 show the blocking behavior for resolvers in Satellite and echo servers in Quack. There are large variations in the fraction of blocking between vantage points in both Satellite and Quack. There are some vantage points that do not observe any blocking, while others observe a large amount of blocking. Between Quack and Satellite, Satellite observed considerably more blocking, which is in line with at least three of our vantage points that observed large amounts of DNS manipulation. We suspect that many Russian ISPs may not be blocking content on port 7, and hence are not captured by Quack. This is a known shortcoming of Quack by not triggering censors that only act on port 80 and 443. This suggests that one method of circumventing censorship might be serving content over non-standard ports.

Figure 2.5 also shows the fraction of blocking aggregated at the AS level. The similarity between the two CDFs shows that *blocking does indeed happen at the AS or ISP level.* In Satellite, we observe that more than 70% of vantage points observe little to no blocking, while in Quack 50% of vantage points observe no blocking, and close to 90% observe minor blocking.

In our Quack measurements, we were able to look at the kind of blocking observed at each of the echo servers. Similar to our observations in the VPSes in data centers, some vantage points observe blockpages, many others observe resets and timeouts (more frequently resets), showing that censorship mechanisms vary widely in networks all over Russia.

We looked at the similarity between domains being blocked in our remote vantage points. The pairwise similarity is shown in Figure 2.6. We see that our observations from the VPSes and residential probe measurements are consistent with remote measurements as well. Both Satellite and Quack see instances of high similarity, which is either because the vantage points see a high percentage of domains blocked (top left) or because vantage

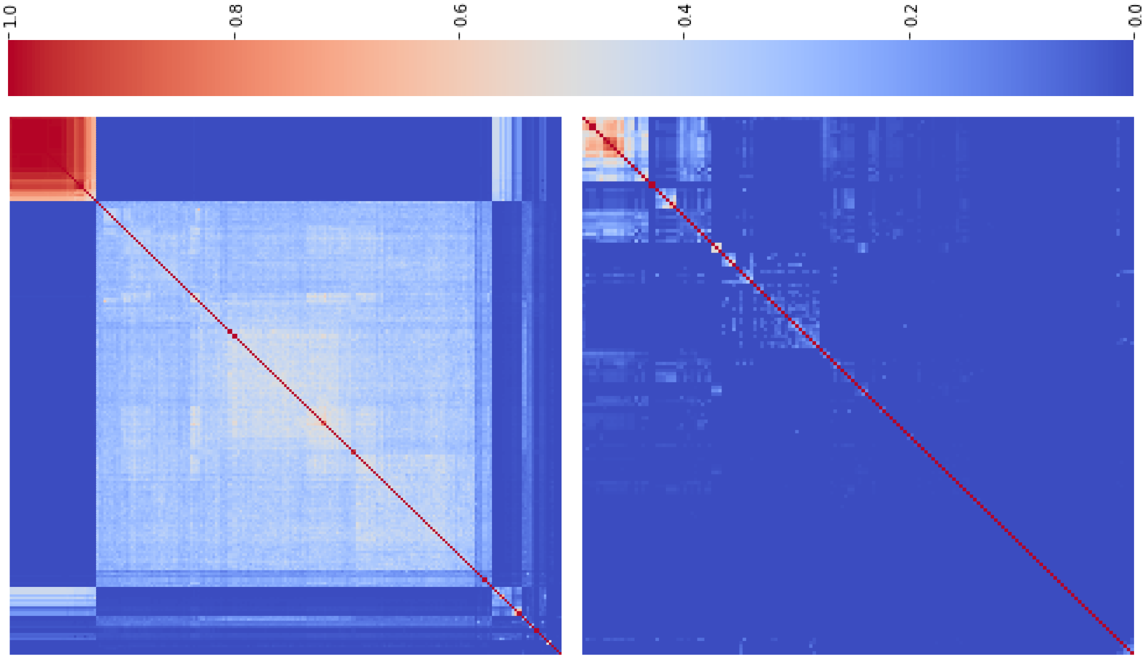


Figure 2.6: **Pairwise Jaccard similarity of domains blocked in remote measurements—** As in the direct measurements, we observe some similarity between domains blocked in remote measurements (Satellite on the left, Quack on the right) either due to high blocking or vantage points in the same ISP.

points are inside the same ISP (small square stripes along the diagonal line). The large blue portions on both plots show that vantage points which observe little blocking do not see the same domains being blocked.

2.5.3 Characterizing Censorship of $RUBL_{ip}$ Measurements

We study the extent of blocking of IPs in $RUBL_{ip}$ by analyzing the output of our TCP/IP measurements from both our VPSes and probes. The amount of IP blocking is shown by the red bars in Figure 2.7. For comparison, we overlay the total percentage of domains blocked in these vantage points as well. Overall, we see a smaller percentage of IPs being blocked compared to domains, which could indicate a desire by the censors to minimize collateral damage (other services hosted on the same IPs would be blocked as well). Alternatively, it could be that residential ISPs do not observe much traffic to IPs, and opt to censor only the traffic they see.

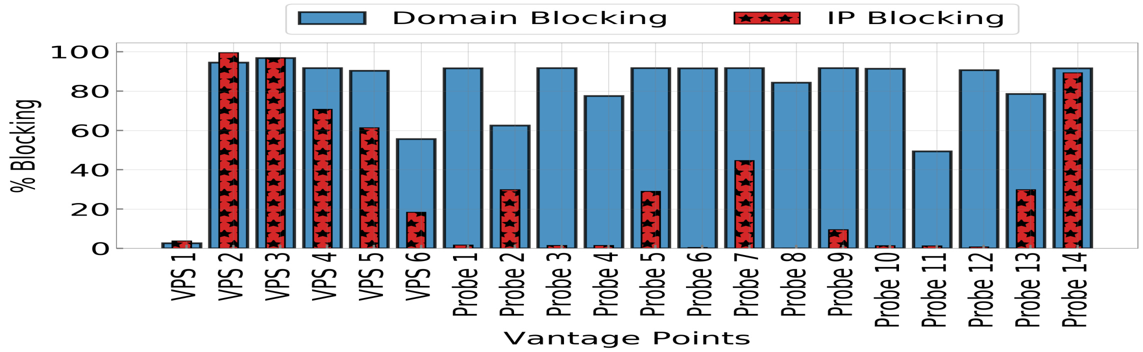


Figure 2.7: **Blocking by method when testing $RUBL_{ip}$ on VPSes and residential probes**—Data center vantage points observe much higher IP blocking compared to residential probes, where domain blocking is more popular.

Vantage Point	Num. of subnets	Vantage Point	Num. of subnets
VPS 1	2	Probe 1	5
VPS 2	31	Probe 2	27
VPS 3	4	Probe 5	6
VPS 5	5	Probe 9	2
VPS 6	1	Probe 10	5
		Probe 11	5
		Probe 13	2
		Probe 14	6

Table 2.5: **Number of subnets completely blocked by vantage points**—VPS 2 and Probe 2 block almost all of the subnets in $RUBL_{sub}$ completely, while others moderately block subnets.

Similar to our observations in $RUBL_{dom}$, we find that there are some vantage points which observe blocking of many IPs, while other vantage points only observe a few blocked IPs. VPSes observe a considerable amount of IP blocking, while the blocking is more sparse in probes. Our experience suggests that *data center VPS providers could also be injecting resets and forcing timeouts to these measurements as well.* In the residential probes, only Probe 14 observes more than 50% of IPs being blocked, while four out of six VPSes observe more than 50% IP blocking. This seems to corroborate the hypothesis that *residential ISPs tend to block the kind of traffic they see more frequently, which is predominantly traffic involving domains.*

2.5.4 Characterizing Censorship of $RUBL_{sub}$ Measurements

Table 2.5 shows the number of subnets that were completely unreachable from our vantage points, omitting the vantage points where at least one IP from each subnet was reachable. Keeping in line with our previous observations, we see that there are some vantage points that block nearly all of the subnets (e.g. VPS 2 and Probe 2) some that block a moderate amount (e.g. VPS 6), and some that do very little blocking (e.g. Probe 12) corroborating our findings in § 2.5.3 that *different ISPs may prioritize blocking different items in* $RUBL$.

Similar to our observation in $RUBL_{ip}$, VPSes in data centers observe much higher blocking in $RUBL_{sub}$ compared to residential probes, where only Probe 2 observes a large amount of $RUBL_{sub}$ blocking. Our $RUBL_{ip}$ and $RUBL_{sub}$ study suggest that *most residential ISPs prefer to block using the domain in the request, as opposed to the IP to which users are ultimately connecting to.* Further $RUBL_{sub}$ analysis can be found in Appendix A.3.1

2.6 Broader Impact of This Work

The findings from this work demonstrates implications for other existing censorship research, highlighting the need for conducting studies from more residential networks. This work illustrates that Russia's censorship architecture that uses deep-packet inspection technology to enable censorship is a blueprint, and even a forewarning of what national censorship regimes could look like in many other countries. We have created reports highlighting the takeaways of our work to encourage the community to better understand its implications [183].

This work has been covered in over 90 news outlets globally [4, 50]. This work was was selected as one of the Top 10 papers at the CSAW'20 Applied Research Competition.

2.7 Discussion and Conclusion

Russia's move towards more restrictive Internet policies is illuminating in the broader context of tightening information controls around the world. Censorship studies have until this point mostly focused on centralized networks like those in China and Iran; Russia's network, however, like that of most countries throughout the world, was shaped gradually by many competitive market forces. The development of effective censors on a decentralized network such as Russia's raises important questions on the future of censorship including in western countries that have not historically favored censorship.

Our study has shown that the implementation of decentralized control breaks the mold of the traditional definition of "censorship": a synchronized and homogeneous process of blocking sensitive content throughout the country. With the advent of SORM and the commoditization of censorship and surveillance technology, it is becoming cheaper and easier for ISPs to comply with government demands. Furthermore, in the years since this study was published, Russia has developed TSPUs (технические средства противодействия угрозам, or technical solution for threat countermeasures) which enable them to exercise control in a more centralized way. TSPUs also have facilitated a unified rollout of censorship policy [250, 187], thereby solving their challenges with this decentralized model.

The variegated nature of Russia's censorship regime has significant implications for censorship research moving forward. It is no longer sufficient to perform measurements from one or a few vantage points within the censoring country. Even two end-users in the same physical location may have dramatically different experiences with censors based on their ISP; they would both see very different results than data centers. Measuring the actual impact of censorship also proves difficult: it requires diverse vantage points, including residential network ISPs, infrastructural machines as well as data centers.

Russia's censorship policy has had implications for censorship resistance as well. During Russia's invasion of Ukraine in 2022, the Russian government's desire to control messaging about the invasion led to increased censorship of alternative sources of information and limited the use of circumvention tools [102]. Russia sparked an arms race in censorship and circumvention, finding new and novel ways to block large circumvention tools, and forcing circumvention tool developers to react to these new blocks.

We have already started to see other large nations begin applying schemes similar to Russia's. In the United States, ISPs have been rolling out DPI boxes over the past decade that can dynamically throttle connections to specific websites, [65, 122, 142] or favor certain content over others [20]. The United Kingdom's censorship model is similar to Russia's, with the government providing ISPs a list of websites to censor [246] and having governing bodies that correspond to various types of censored material [225]. For both the U.S. and the U.K., what this means is not that the current regimes are restricting the volume of information that Russia is, but that the option to follow the same path is cheap, and readily accessible.

The same can be said for nations around the world. Portugal has recently been cited for not supporting net neutrality [175], Indonesia recently implemented broader content filtering [90], and India has been ramping up censorship [252]. A recent report [239] finds that Russian information controls and the technology used for surveillance and censorship capabilities are being exported to at least 28 countries. As more countries move towards stricter Internet access, Russia's model for censorship may become more commonplace, even in countries with a tradition of freedom of expression on the Internet.

In conclusion, Russia's censorship regime raises the stakes for censorship measurement and resistance. Its censorship architecture is a blueprint, and perhaps a forewarning of what national censorship regimes could look like in many other countries that have similarly diverse ISP ecosystems to Russia's. As more countries require ISPs to deploy DPI infrastructure for purposes of copyright enforcement or filtering pornography, we risk a slippery slope where Russian-style censorship could easily be deployed. The Russian government's move from a decentralized censorship model to a more centralized, centrally-controlled model with TSPUs illustrates that governments' censorship models can evolve with support from homegrown technologies. Further, the emerging censorship technique of using throttling sets a dangerous precedent for all countries that seek to discourage citizens from accessing prohibited resources. The proliferation of these homegrown and commercialized "dual-use" technologies such as DPI devices has equipped censors around the world with a more complex toolkit to implement more advanced techniques than outright blocking.

CHAPTER III

Empirical Understanding of VPN Users and VPN Providers[1]

Since their introduction over two decades ago, the use of Virtual Private Network (VPN) technologies has grown rapidly. With commercialization, VPN products have found their way into a regular Internet user's toolbox [106, 203]. Though the VPN ecosystem has expanded into a multi-billion dollar industry [177], questions regarding why VPNs have been adopted so widely are still unanswered. Is the popularity of VPNs grounded in an understanding of risks on the users' part? Is the rise of VPNs due to dwindling trust in Internet service providers? What benefits do users perceive to gain?

A majority of previous studies have found various issues in the technical implementations of VPNs [89, 103, 240, 184, 251]. Only limited prior work has delved into the human factors of VPN use: factors that contribute to retention of VPNs [150, 261], attitudes of university students and corporate users towards VPNs [17, 49, 48], and the widespread misconceptions of how privacy-enhancing tools work [217].

However, no study has combined both the users and VPN providers perspectives to answer fundamental questions about the VPN ecosystem. For instance, users using VPNs are essentially transferring trust from their network provider onto the VPN provider, but

[1] This chapter is based on: Reethika Ramesh, Anjali Vyas, and Roya Ensafi. 2023. "All of them claim to be the best": Multi-perspective study of VPN users and VPN providers. In the 32nd USENIX Security Symposium (USENIX Security) 2023 [189]. This project was supported by the Open Technology Fund, Consumer Reports Digital Lab Fellowship, and National Science Foundation grant CNS-2141512.

it is unclear as to what VPN features encourage them to make this shift? On the other hand, the VPN industry has been known to employ various marketing tactics [7] and dark patterns around discounts [227, 126], but it is yet unknown if these practices are bound to have any significant effect on VPN users. Moreover, the community has not yet understood VPN providers' incentives in sustaining such dark patterns, nor do we know what efforts they take to foster user confidence in an ecosystem plagued with mistrust. To gain a clearer picture of the inner workings of such a large consumer ecosystem, it is imperative to study both its users and its providers.

This is the first multi-perspective study that uses a *quantitative survey of* ***(n=1,252)*** *VPN users* in the U.S. along with *qualitative interviews of nine leading VPN providers*. We choose to survey 1,252 users, that have either used or currently use a VPN, to provide us with practical insights into our various lines of inquiry that we systematize into the following research questions:

RQ1: [Motivations] Why do users use VPNs?

RQ2: [Needs and Considerations] What factors around VPNs do users consider when choosing a provider?

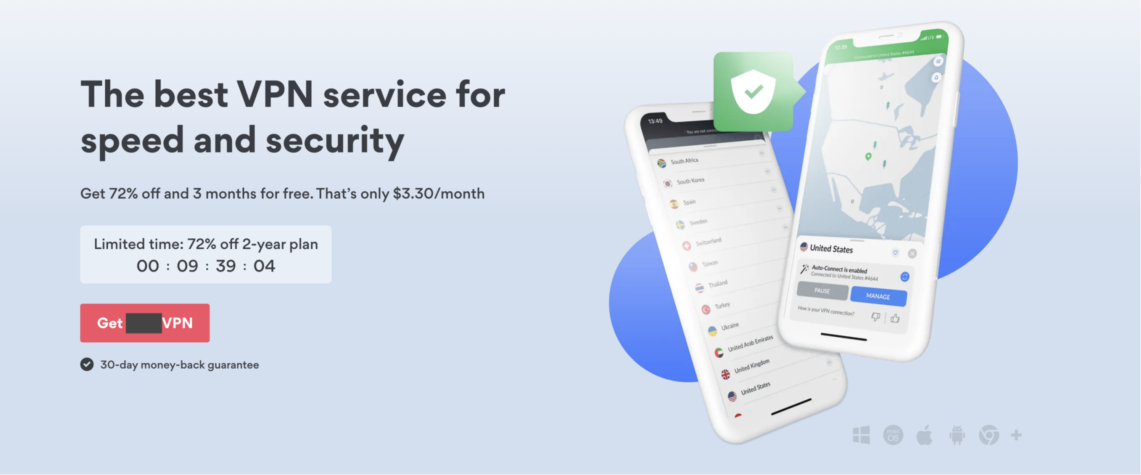
RQ3: [Emotional Connection and Threat Model] How safe do users feel when browsing the internet with and without a VPN? (If and) From whom do users want to secure/conceal their online activity?

RQ4: [Mental Model] Do users have an accurate understanding of how VPNs work and what data they collect?

RQ5: [Perception and Trust] How do users perceive the VPN ecosystem?

RQ6: [Alignment between VPN users and providers] What are the key areas of (mis)alignment in priorities and incentives between the two?

We find that users rate speed, price, and easily understandable GUI, as the top requirements from VPNs rather than features such as the variety, number of available VPN servers, and their locations. We also find that in alignment with VPN providers' expectations, pricing



The best VPN service for speed and security

Get 72% off and 3 months for free. That's only $3.30/month

Limited time: 72% off 2-year plan
00 : 09 : 39 : 04

Get [redacted] VPN

30-day money-back guarantee

Figure 3.1: Example of dark pattern–using countdown timers.

plays a key role with users. Thus indicating that discounts, and marketing around pricing can have a significant effect on them. Prior research suggests that malicious marketing tactics [7] and dark patterns around discounts are common, which are often used to ensure customer lock-in [227, 126]; an example of such a dark pattern in the VPN ecosystem is shown in Figure 3.1.

Interestingly, we find that when it comes to choosing VPNs to use, more users seem to lean towards using search engines (61.1%), and recommendation websites (56.5%), rather than relying on more traditional methods such as word of mouth (5.7%). Furthermore, almost 94% of these users rate these websites as trustworthy. On the other hand, our interviews with VPN providers highlight that the VPN recommendation ecosystem is mostly money-motivated, with widespread malicious practices that include having paid review spots, and auctioning off the #1 spot. Some "review" sites have been reported to send emails to VPN providers asking for higher cost-per-action/click to get ranked on their list [253]. Users' reliance on such websites further amplifies our worries of an unregulated marketing ecosystem around VPNs.

Exploring reasons for why users use VPNs, we discover that users attach an emotional connection with using a VPN, namely a feeling of safety (86.7%), which was found by prior work to be a key factor in retention of VPN use [150]. Our intuition suggested that exploring users' threat models could explain why they attach such considerations; indeed, we find that 91.5% of users indicate they use VPNs for securing or protecting their online activity. When exploring who they aim to protect it from, we find that their top concerns are hackers/eavesdroppers on open WiFi networks (83.9%), advertising companies (65.4%), and internet service providers (46.9%). This marks a departure from known prior concerns such as government surveillance (30%), and indicates a shift of attitude towards surveillance capitalism and user privacy.

Given the emotional attachments and user dependency on VPNs for security and privacy concerns, we find that an alarmingly high proportion of users (39.9%) have a *flawed mental model* of what VPNs provide them and what data they collect. These users believe their ISP can still see the websites they visit over the VPN. More worryingly, we do not see a significant difference between users of different expertise having flawed mental models (χ^2-test, p=0.0927, N=1252). From our VPN provider interviews, we find that providers also mention that they recognize the need for improving user knowledge, and consider effective education a key challenge. We also find that dark patterns in the industry may also be a key issue; multiple VPN providers mention "malicious marketing" is problematic, including preying upon users' lack of knowledge and overselling of service.

Continuing to explore the confusion surrounding the operations of VPN providers, we find that a significant portion of limited expertise users believe that the data is being collected for monetization, such as advertising (36.4%), user tracking (36.4%), and selling to third parties (33.6%). Although a majority of all users (79.2%) believe the main reason for data collection is internal analytics, confusion found amongst the limited expertise users may be even more widespread among the VPN users in the general public. Moreover, users also expressed high degrees of concern towards VPN providers selling their data (73.2%). This is yet another area of misalignment between users and providers because multiple VPN providers believe that they clearly communicate their logging practices, and/or have released audits to prove this.

From our study of 1,252 users and 9 VPN providers, we present the following actionable recommendations for the VPN ecosystem: prioritizing user education, oversight on advertisements and marketing surrounding VPNs, coordinated efforts to bring attention to the flawed VPN recommendation ecosystem, and regulations to curb malicious marketing tactics that lead to false mental models and false expectations for users. We believe that our work will help security and privacy advocates such as EFF and CDT, technologists, and VPN providers alike, by calling attention to the key areas in the commercial VPN ecosystem.

3.1 Background & Related Work

3.1.1 Virtual Private Networks

Virtual Private Networks were initially created in 1996 [138] as a peer-to-peer tunnelling protocol developed in Microsoft to facilitate private communication in enterprise settings. Virtual private networks (VPNs) provided a way to create private connections between computers and transfer data between them securely over the public Internet. These are still the guarantees that VPNs provide for general users today. VPN products (VPNs hereon) create a secure connection, often called a "tunnel", to a secure server that then connects them to their intended destination. This tunnel typically provides an extra layer of encryption that serves as protection from surveillance by the intermediate networks, bypasses access restrictions active in those networks, and hides the user's actual IP address from their destination service [125].

Commercial VPN providers make use of the available VPN protocols such as OpenVPN, L2TP, IPSec, IKEv2, and Wireguard [164, 222, 101, 45], or develop proprietary protocols which are typically extensions of existing ones, optimized to fit their particular needs and business model. VPNs offer different subscription models: paid/premium services, free to use services, and freemium models that offer limited free features and charge for premium features and services.

While some work has focused on analyzing technical aspects [184, 221, 103, 89], Weinberg et al. [240] focused on evaluating the claims of VPN server locations, and found at least one-third of the 2269 servers were definitely not in the country advertised, and another one-third probably were not in the location they claim. Investigating an often overlooked source of security advice, Akgul et al. [7] studied 243 YouTube videos containing VPN ads and find a number of concerning misleading claims, including over-promises and exaggerations which may lead to users forming inaccurate mental models of internet safety. There have also been news reports of data breaches, leaks and misuse by VPN providers, some of which were published on VPN recommendation websites [202, 233, 254, 182].

3.1.2 User Adoption of VPNs and Other Tools

As users adopt more privacy-enhancing tools such as VPNs for a variety of reasons, their privacy needs become important to assimilate. The level of security and privacy a user needs may depend on myriad factors like the reasons for use, tolerance of failure, legality of these VPN services in the country of the user *etc*. Only few community efforts focus on providing threat model based VPN (and other tools) recommendations, such as the Security Planner [34]. We present the related work summarized in relevance to the topics studied:

Prior work studying VPN users: Namara et al. [150] conducted a study with 90 technologically savvy users and studied the adoption and usage of VPNs, and the barriers they encounter in adopting them. They find users with emotional reasons to use a VPN such as fear of surveillance or desire for privacy, are more likely to continue using them rather users who use it for practical reasons. Similarly exploring the factors that influence user decisions to adopt VPN apps, Sombatruang et al. [214] interviewed 32 users in UK and Japan and found that user review rating and price significantly influenced the choosing of a VPN to use.

Prior work exploring particular sub-populations: Binkhorst et al. [17] studied the mental models of 18 users in the context of corporate VPNs, and found that experts and non-expert users have similar mental models of VPNs, and experts also tend to have false beliefs on security aspects of VPNs. Dutkowska-Zuk et al. conducted a study focused on a specific sub-population of 349 university students to find how and why they use VPNs, and whether they understand the various privacy risks caused by VPNs [49]. They found that students are mostly concerned with access to content rather than privacy, and that most students did not use VPNs regularly. Extending this study, they looked at how these students compare to general VPN users in the awareness of risks of VPN use and how they adopt VPNs [48]. Specifically, they found that despite having different use cases, both groups had low understanding of the risks of data collection by VPNs, highlighting the need for better awareness campaigns.

Prior work studied user attitudes and use of privacy-enhancing tools: Various prior works have shown that users, particularly in the U.S. are aware of risks such as tracking, and are concerned about online tracking in different situations [136, 25, 181]. However, some prior work has shown that they are unclear on how to protect themselves [206]. Story et al. [217] highlighted this in their study of the use of and perceptions about web-browsing privacy related tools. In their survey of 500 U.S. users, they ascertain user perception of the protection provided by different tools across 12 different scenarios, and interestingly, they find that users having more experience using VPNs is associated with confusion about their protection. Further, studying the adoption and abandonment of 30 commonly recommended security and privacy practices, Zou et al. [261] surveyed 902 users and find that security practices were more widely adopted and privacy related practices were among the ones most commonly abandoned.

These prior work, though useful, are limited in the scale, topics studied, and have focused on particular sub-populations of VPN users. **In our work, we create a novel line of inquiry to study the motivations, needs, and considerations of VPN users in depth, and improve greatly upon the scale of users surveyed. We are also the first to**

conduct a study of VPN providers. We augment insights from both users and providers to characterize any misalignments between them, which could be exploited by bad actors to further deepen problems in the VPN ecosystem. Given the wide reach of the VPN ecosystem, our study will help technologists and security and privacy advocates gain a deeper understanding of the key problem areas where they can focus their efforts.

3.2 Methods

We set out to study VPN users and providers to understand their unique perspectives on the VPN ecosystem and the issues surrounding it. We conduct a large-scale survey of VPN users as well as a qualitative interview of nine VPN providers.

3.2.1 User Survey

Small-Scale Interviews and Interactions. Any successful large-scale quantitative study must be preceded by a smaller-scale qualitative study and community research to extract key concerns. To that end, we conduct seven user interviews (4 men, 3 women, ages 18-45), and we participated in various VPN-focused community events with VPN providers and users in attendance in order to gather topics and research questions that interest the community.

For the small-scale user interview study, we design a questionnaire with open-ended questions to serve as the framework for each interview, and obtain approval from our Institutional Review Board (IRB). During the interview, we collect general demographic information, including gender, age range, occupation, country of residence, and level of education. Our introductory questions ask about the interviewee's awareness of their own threat model and of online risks such as trackers. Next, we ask about the perceived positives and negatives of VPN use. We then ask participants to sketch their understanding of how VPNs work while walking through the steps of setup and use, diagrammatically. The interview concludes with questions about how the VPN ecosystem can improve.

We recruit seven participants via a pre-interview survey at the Citizen Lab Summer

Themes	Definitions
Reasons for using VPN	Motivations for and reasons to use a VPN
VPN Use	General thoughts about commercial VPNs, what they look for
Threat model for using a VPN	Personal threat models for needing, using, and/or recommending a VPN
Mental Model of VPN	What is a VPN and what does it provide me? (Sketching exercise included)
Attitudes towards VPN services	What is lacking in current ecosystem, their perception of what the VPN ecosystem looks like
Improving ecosystem	Thoughts to improve ecosystem and boosting adoption and safe usage

Table 3.1: Six themes with their definitions.

Institute [140] that has global attendees who are passionate about technologies aiding Internet freedom, security, and user rights. Prior to the start of each interview, we obtain explicit consent for participation and permission to audio record it using an IRB-approved consent form. Participants are also given the chance to ask any questions before the interview begins and are allowed to stop at any point. After completion of the interviews, the first author transcribed all the recordings. Overall, the interviews lasted 15-20 minutes not including setup and conclusion.

Developing the Large-Scale Survey Instrument. After completing the interviews, we use an inductive open-coding method for analysis. Two members independently coded all the transcripts, and held a meeting to resolve any disagreements and create a codebook. The research team then met to collaboratively go through the codebook and identify emerging themes [21] and hence, we do not present inter-rater reliability for this case [134, 139]. We augment these with the knowledge extracted from attending several Internet freedom community gatherings organized around VPNs and VPN use including the IFF VPN Village [91]. Finally, we combine our work to arrive at six common themes, shown in Table 3.1.

Using these themes, we devise an initial survey instrument to study VPN users. The instrument contains questions aimed at understanding users' motivations, needs, and considerations when it comes to VPN services, and discerning their threat models, perceptions of VPNs, and understanding of how VPNs work. During the design phase, we also create a consent form and obtain IRB approval. Our survey questions only collect the information we need and do not involve the collection of any personally identifiable information.

Cognitive Pre-testing. In order to reduce the potential for biases that arise from the ordering and/or phrasing of questions, we conduct systematic pretesting in *three phases*, iteratively improving the survey between each phase.

First, we recruit test participants (from the target demographic, VPN users) at an Internet freedom, security and privacy focused event organized by the Open Tech Fund [163] to pretest the initial survey instrument and obtain unbiased opinions about the survey. The pretesting involves vocally stepping through the survey while a facilitator from our team takes notes. We use these notes to detect biases, signs of confusion regarding the intent of the question, as well as "leading" questions. In this round, 17 pretesters worked through the survey, and we learned that comparison-scale adjectives (None at all, Little, Somewhat) were unclear for participants. We amend the scales to avoid ambiguity and provided clearer distinctions e.g. we use Likert-type Scale when asking about concern or importance. The scale is provided on the Qualtrics software and is a psychometric scale developed for scaling responses in survey research [115]. We learned that participants had varying understandings of what "commercial VPNs" mean, and that participants were not sure if the questions pertained to personal or professional VPN use. To remove ambiguity, we define "commercial VPNs" on the survey landing page and present examples within the survey.

After refining our survey using the initial pretesting, we requested external user-study experts to go through our survey and provide feedback. They helped us refine our matrix style questions and simplify the organization of our survey.

After incorporating expert feedback, we run the last round with eight new pretesters. This round helped us refine some of the examples used in the survey, improve consistency of language, and disambiguate a handful of questions.

The Final Survey Instrument. The final survey instrument contains six parts, 28 questions (with sub-parts), and we incorporate one quality check (where they must confirm they use a VPN) and two attention checks. The survey starts with a demographic section, where we follow the community best practices for inclusive language, and also have a "prefer not to disclose" option for all demographic questions [216, 192]. Then, we ask users general questions about their VPN usage, reasons for using a VPN, the resources used in discovering VPNs, importance of different criteria and features, their mental model of VPNs and the data it collects, their emotional connections tied to their use of a VPN (e.g. safety), and their expectations from a VPN provider. We specifically avoid using words such as privacy and security in the text, since these concepts are broad, subjective, and mean different things to different users. Instead we allow users to select from list of options, and we distill into certain buckets during analysis. The final survey instrument is available in our pre-print [188, Appendix C].

Analysis of Survey Data. For the quantitative data from the survey, we report the results summarizing the users' responses. We aim to understand how different subgroups of users answer the same question, i.e. users with different security and privacy expertise, and users who prefer to use certain subscription type (free or paid VPNs). We conduct χ^2-tests, where all the assumptions are satisfied in each case, to examine if users in different subgroups of the same type (e.g. expertise) answer questions differently. If there were significant differences between subgroups, we conduct pairwise comparison Z-tests (α=0.05), where we adjust the significance levels for multiple comparisons through the FDR-BH adjustment [16] and present how they compare to each other.

We analyze the survey participants' open-ended text box responses using inductive coding. A primary coder created an initial codebook and assigned codes to all responses. A second coder analyzed 20% of the responses for each coded question and ensure high inter-rater reliability [111]. Cohen's κ between the two raters is 0.81, 0.86, 0.75, 0.81 for each question in Appendix B.3, indicating moderate to strong agreement [33, 135]. The coders also coded responses for “Other” write-in options in questions, and present the responses in the results (§3.4).

3.2.2 Qualitative Interviews of VPN Providers

Interview Instrument. Using the same parent themes as mentioned in Table 3.1, we create a questionnaire to interview VPN providers. These questions aim to extract insights from the providers about their users, VPN users in general, their business decisions, and what they see as the main issues in the VPN ecosystem. We design the topics for the questions to be counterparts to the VPN user survey.

Interview Procedure. We design a semi-structured interview with eight broad open-ended questions, and five additional questions to ask in case we have time. We obtain IRB approval prior to conducting the interviews. Our questionnaire[2] serves as a framework for the interviews to ensure we maintain structure and consistency from one provider to another. However, we also explore statements made by the interviewees for clarity and insights.

We begin all the interviews by presenting the interviewees with an overview of our project. Using an IRB-approved consent form, we obtain explicit consent for participation and audio recording the interview. On average, the interviews were ≈44 minutes in length, not including set up and conclusion. We conclude the interviews by thanking the participants, and provide ways to contact us to learn more about our project.

Analysis using Qualitative Coding. The first author transcribed all the interviews and the analysis is done using inductive open-coding, and thematic analysis [21]. Although we have nine VPN providers, we have eight transcripts in total because two of the providers opted to interview together[3] and each of them answered each question independently. A

[2] The questionnaire is presented in our pre-print [188, Appendix D]

[3] They are (non-commercial) partner projects, with separate services.

primary coder coded all transcripts, and two additional coders independently coded five and three transcripts each. Then, the team went over each coded transcript together to reconcile any differences. We then collaboratively identify the emerging themes for each question, and common themes that appear across different questions. Since the team collaboratively analyzed the coded transcripts together to identify themes, we do not present inter-rater reliability [134, 139].

Since this interview is meant to shed light on the VPN provider's perspectives and form a clearer picture of the VPN ecosystem, we only report aggregate results after performing thematic analysis. We anonymize the comments and do not attribute statements to particular providers.

3.2.3 Recruitment

User Survey. In partnership with Consumer Reports, a leading consumer research and advocacy organization with over 6 million members, we launched our user survey on March 1, 2021. We ask VPN users to participate in our survey by distributing the recruitment message in Consumer Reports' tech-focused mailing list, subreddits such as `r/VPN`, `r/asknetsec`, `r/samplesize`, and on Twitter using the research team's own personal accounts. We also request participation from users on mailing lists belonging to Open Tech Fund and Internet Freedom Festival. We opt to recruit participants organically and to ensure anonymity, we did not offer any compensation for taking the survey.

VPN Provider Interviews. We reached out to 15 leading VPN providers; nine of whom agreed to our interview. We chose to contact commercial VPN providers based on their popularity in the U.S., and included non-commercial projects that develop VPNs for users, based on their involvement in the Internet freedom and anti-surveillance community. We did not compensate the interviewees for participation.

The VPN providers we interviewed are the following (in alphabetical order): CalyxVPN, Hide.me, IVPN, Jigsaw Outline, Mullvad VPN, RiseupVPN, Surfshark, TunnelBear VPN, and Windscribe. We interviewed CEOs, CMOs, and/or researchers working in the company who were authorized to speak to us on behalf of the company.

3.2.4 Ethics

Our user study is approved as exempt from ongoing review under Exemption 2 as determined by our Institutional Review Board (IRB), and the VPN provider interview received a "Not Regulated" status. Furthermore, we draft a privacy policy document that was reviewed by experts from Consumer Reports, and add it to our website. We also provide information on our study's Qualtrics page and ensure that our participants, pretesters, and interviewees explicitly consent to the study.

We follow user survey best practices such as using mindful, inclusive language in collecting demographics data [216]. We also offer "prefer not to answer" as an option on our required demographics questions as per American Association for Public Opinion Research code of ethics [113, 192]. We did not collect any personally identifiable information from our participants, and our results from the VPN providers are anonymized as well.

We solicit participation as mentioned in §3.2.3 and to ensure anonymity, we offer no compensation for any of our studies. We deeply analyze the collected responses to ensure response quality, as we detail in §3.3. Audio-recordings of the interviews (both the small-scale user ones, and the VPN providers) were only accessed by the first author who did all the transcriptions.

3.2.5 Limitations

As with many user surveys, some of our comparisons rely on self-reported data, which is prone to biases. We take efforts to reduce these biases to our best extent, elaborated in §3.3, such as by explicitly explaining the different levels of privacy and security expertise in Q7.

Our participants are not fully representative of the global users of VPNs. Our respondents skewed older, male, and more educated than the general U.S. population; this reflects the main user population for VPNs, especially in the U.S. [232]. Our collaboration with Consumer Reports demonstrated to us that their user base, who formed a large part of our recruitment, are avid VPN users that express the need for recommendations and advice from experts. Though we study a more-educated and possibly more tech savvy user base, the issues that we identify in our results (e.g., inadequate understanding) lead us to believe that such problems may be *even more prevalent* among the general U.S. population. Therefore, we argue that our results serve as an upper bound, and our recommendations will benefit the larger, more-general user base as well.

We restricted our analysis to only people located in the U.S. While VPN users outside the U.S. have diverse and valuable perspectives, their use cases are also different. Future studies could specifically explore the perspectives of users from countries where VPNs are commonly used to circumvent censorship or other access restrictions.

We intentionally only include users of commercial VPNs, university VPNs (typically managed by the university or a third-party), and users of free, and non-commercial VPN services in this study. We do not include users of self-hosted VPN solutions or (managers and users of) workplace-specific VPNs. We leave it to future work to explore these specific subgroups of users, since they are typically more highly-skilled, and/or possess high levels of technical knowledge.

3.3 Data Characterization and Validation

Survey Responses. In total, we collected the user survey responses for six months, from March 1 to September 1, 2021. We had a total of 1,514 valid, completed responses out of which 1,374 (90.8%) indicated they are in the U.S.. The second-highest country (China) had 23 participants, and 20 countries had only 1 participant each. We decided to focus on the U.S.-based participants (and VPN providers popular in the U.S.) as there is not enough

Demographic	**Respondents**	**%**
Man	1011	80.75%
Woman	202	16.13%
Prefer not to disclose	35	2.8%
Non-Binary	4	0.32%
Over 65	741	59.19%
56-65	260	20.77%
46-55	105	8.39%
36-45	56	4.47%
26-35	36	2.88%
Prefer not to disclose	34	2.72%
18-25	20	1.6%
Post-grad education	527	42.09%
College degree	508	40.58%
Some college, no degree	150	11.98%
High school or eqlt	41	1.20%
Other	15	1.2%
Prefer not to disclose	11	0.88%
High-expertise users (Knowledgeable/Expert)	511	40.81%
Moderately knowledgeable users	631	50.40%
Limited-expertise users (No or mildly knowledgeable)	110	8.79%
Total	1252	

Table 3.2: Demographics of the (n=1252) survey respondents.

sample to draw meaningful conclusions about other countries.

Quality Checks. We have three questions, one quality- and two attention-checks, to ensure high-quality responses. Among the 1,374 U.S.-based participants that finished the survey, 1,264 or 92% passed our generic quality check. Next, we filter out users that failed both of our attention checks (Q11 and Q21). Furthermore, we review open-ended responses from the 259 participants that failed at most one attention check, as done in [131], and find that over 95.8% of these users had insightful responses. Hence, we consider **1,252 users** that passed at least one attention check for the rest of the analysis.

Participant Demographics. Ours is the largest survey of VPN users to date, and we report on the 1,252 valid, high-quality responses. Shortly after launching the survey, we served on the panel of a VPN workshop organized by Consumer Reports with over 1,500 enthusiastic users in attendance, and sent out our study recruitment message to them. We believe that our various recruitment methods ensure that we study users who are highly motivated about commercial VPNs and actively use them. Our participants skewed older, male, and highly

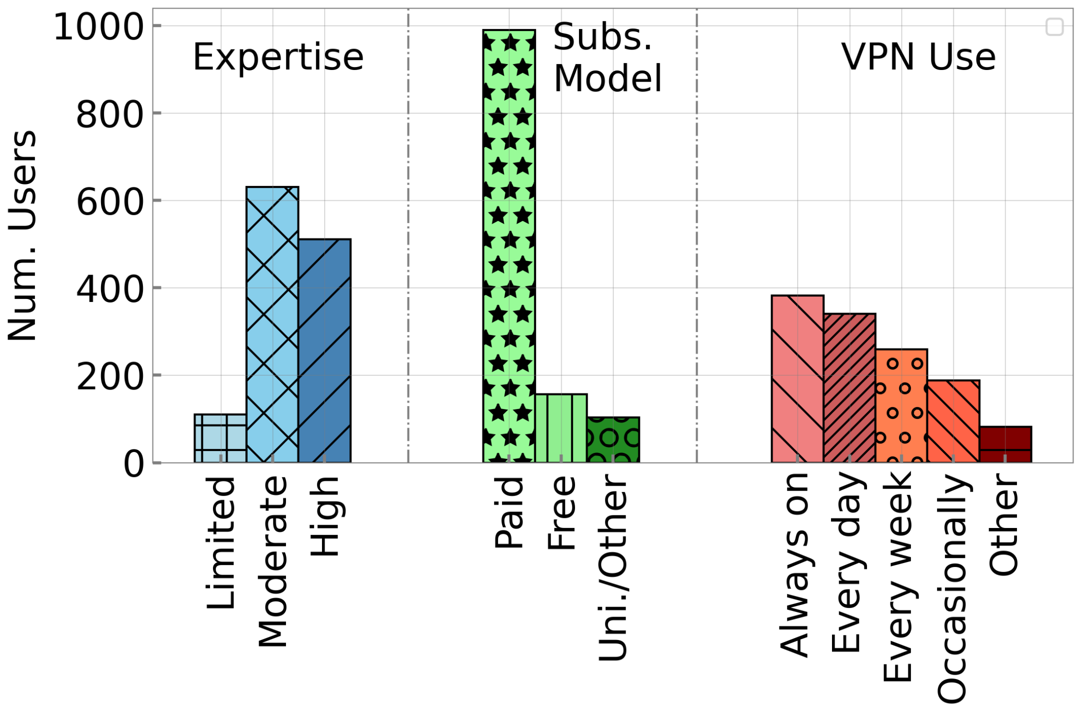


Figure 3.2: Overall statistics of the users' security and privacy expertise, their VPN subscription type, and VPN usage.

educated. However, due to the high number of responses we obtained, we are still able to make significant conclusions from the data. Though our participants do not represent **all** VPN users, our results (§3.4) indicate concerning issues even amongst the more educated, more tech savvy users, implying that our recommendations likely will benefit the more general VPN user population. The demographics are described in Table 3.2.

Cross-validating Self-reported Expertise. We report our results for different sub-groups of users based on their self-reported expertise, and type of VPN subscription they generally use, shown in Figure 3.2. We bucket participants based on their reported expertise in security and privacy: high expertise users (knowledgeable, expert), moderate expertise users, and limited expertise users (no, mild). In order to mitigate self-reporting biases, we follow all the recommended survey design methodology best practices by including descriptive explanations for each expertise level. We craft these explanations using our expertise and incorporating feedback from user survey practitioners. We use the terms “security” and “privacy” in these descriptions to allow users to use their own judgements, and we use our threat- and mental model questions later to have the user expound on their definitions. Furthermore, we analyze the open-ended text box responses to cross-validate users’ expertise, and find that high expertise users provided insightful details to add to their mental models (presented in Appendix B.3.4) and limited expertise users were more likely to admit they do not know what protection the VPN offers them (§3.4.4).

3.4 Results from the User Survey

Security and privacy advocates, and technologists need a deeper understanding of the VPN ecosystem, and the misalignment of understanding between the stakeholders (VPN users and providers) can be exploited by bad actors to further deepen problems in the VPN ecosystem. To investigate and illuminate such issues, in this study, we conduct quantitative and qualitative studies of VPN users and VPN providers. Based on the responses from our survey and interviews, we answer the following research questions:

3.4.1 RQ1: Motivations

First, we explore the reasons for which users use VPNs and allow them to choose multiple reasons. We provide them various options that we then distill into different categories.

Security and privacy are the main reasons why users use a VPN. We find that protection from threats, which we consider a *security motive* (82.1%, 1,027 of 1,252) and making public networks safer to use, which we term *privacy motive* (58.4%, 731) are the biggest reasons why users use VPNs. On the other hand, censorship circumvention (8.8%, 110) and file sharing such as torrenting (12.1%, 151) are among the least popular reasons. Our results are in contrast with [49] which finds university students prefer access to content (institutional, media streaming) over privacy, possibly due to the different priorities of the user populations. The overwhelming number of users that use VPNs for protection from perceived threats indicates the successful marketing of VPNs as a panacea for all security and privacy issues in the Internet.

Furthermore, 118 users also write-in additional reasons why they use VPNs (Appendix B.3.1). Users mention *privacy* (60.2%, 71 of 118; from ISP, tracking, surveillance, ad targeting) , *security* (12.71%, 15), being *offered the service* (10.1%, 12; by a company, with a purchase), *during travel* (7.6%, 9), and *anonymity* (2.5%, 3) as the main reasons for use.

Since finding a suitable VPN is not a trivial task, we ask users whether they had difficulty in selecting a VPN provider. Although the responses are almost evenly spread over the difficulty scale, we find differences between users with varying security and privacy expertise shown by a χ^2-test (p = 0.004206, with N=1251). As mentioned in 3.2.1, we perform pairwise z-tests (α=0.05) with FDR-BH correction to find how different user groups relate to each other.

High expertise users less likely to find VPN discovery very difficult, more likely to find it somewhat easy. We find that only 3.7% (19 of 511) of high expertise users find the discovery process very difficult which is significantly less than the 7% (44 of 631) of the moderate- and 11.9% (13 of 109) of the limited expertise users who find it so. High expertise users are significantly more likely to find the process somewhat easy (21.1%, 108 of 511, compared to 11% of the limited expertise users).

Furthermore, we find significant difference between users that use different subscription types (free, paid/premium, other) also shown by a χ^2-test (p = 0.000005, with N=1249). Understandably, university and "other" VPN users (most use a VPN provided as part of a software suite) are significantly more likely to say the process was somewhat or very easy (58.8%, 60 of 102) compared to 33.7% (334 of 990) of paid VPN users and 28% (44 of 157) of free VPN users. A portion of both the free VPN (40.8%, 64 of 157) and paid VPN users (34%, 337 of 990) find the process at least somewhat difficult. All of these findings are detailed in Table B.1 in the Appendix B.

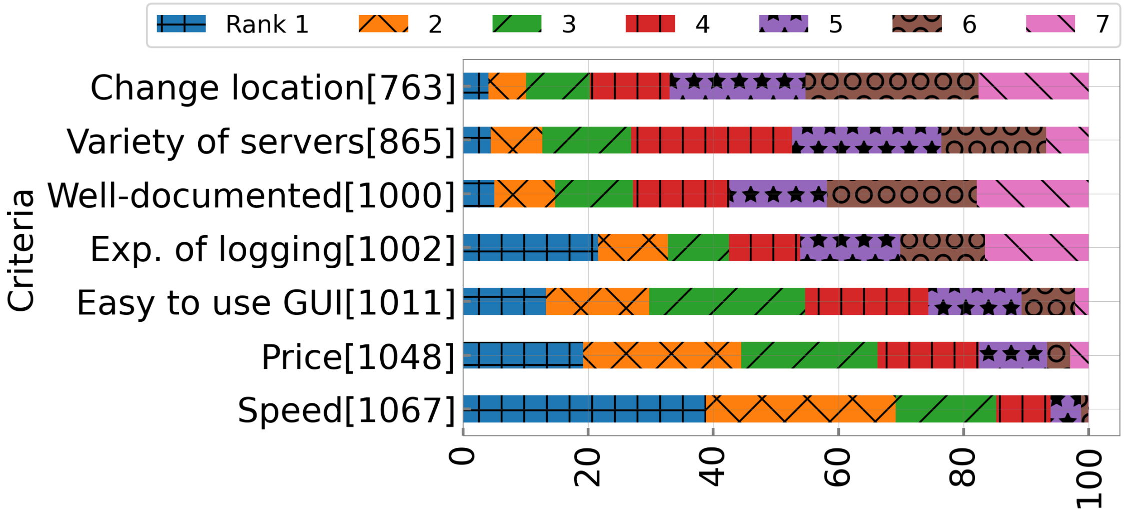


Figure 3.3: Importance levels users attach with criteria they look for in a VPN, presented along with number of users who chose it. Ranked from 1-most important to 7-least.

3.4.2 RQ2: Needs and Considerations

To understand the needs that different users have, we ask them choose and rank criteria that they look for in a VPN. We ask the users to select the criteria they require in a VPN, and/or prefer to see in a VPN and then ask them to rank those criteria, from most important to least.

Speed, price, and an easy to use app are among the top three requirements in a VPN. We see that speed (72.6%, 909 of 1,252), price (55.4%, 694), and easy to understand app/GUI (44.1%, 553) are consistently among the top three requirements for VPN users, and over 216 users (17.3%) ranked clear explanation of logging and data practices as their number one, as shown in Figure 3.3. On the other hand, variety or number of servers (18.8%, 235 of 1,252), and using a VPN to change location for media sites such as Netflix (12.4%, 155) are among the lowest ranked requirements. We also find that logging data practices, which have received relatively little study, are ranked more highly than criteria like changing location for content or number of VPN servers, which have received more attention in the literature [240]. We highlight that understanding real-world user requirements can help shape future research focus.

Price is a big criteria for limited-to-moderate expertise users. Interestingly, users of all expertise rank speed equally highly as a top three criteria (no significant differences, p=0.348, N=1067). But limited-to-moderate expertise users are significantly more likely to rank *price* higher (χ^2-test, p=0.000150, N=1048); 71.1% (436 of 613) of these users rank it in their top three, compared to 59.3% (258 of 435) of high expertise users. This means that prices, discounts, and marketing around these factors is bound to have a vast effect on these users, similar to the study on UK and Japan users [214]. As we will demonstrate in §3.5, malicious marketing around pricing is common and dark patterns are often used to ensure customer lock-in.

On the other hand, **high expertise users rank clear explanation of logging significantly higher** (53.4%, 237 of 444 who chose it put it the top three) than all other users (33.8%, 164 of 485 moderate- and 34.2%, 25 of 73 limited expertise users) as shown by a χ^2-test ($p \ll 0.0001$, N=1002). Also, we find that significantly more high expertise users value an easy to understand GUI lower (only 38.6%, 158 of 409 high expertise users chose and rank it in their top three) compared to 64.4% (334 of 519) of the moderate- and 73.5% (61 of 83) of the limited expertise users, shown by a χ^2-test ($p \ll 0.0001$, N=1011). This indicates that high expertise users may be more confident in their ability to use a VPN application, and place higher value on the clarity of communication about the VPN service and the provider's data practices.

Users rely on search and recommendation sites rather than word of mouth to choose a VPN. Given these different needs and criteria, we explore what resources users use to discover and choose the right VPN for them. Users report that actively researching on the Internet (61.1%, 765 of 1,252), using recommendation websites (56.5%, 708), and reading the VPN providers' websites (48.1%, 602) are the top three ways they use to find a VPN for their needs. Users lean on these search engines and recommendation websites, rather than traditional methods like word of mouth from friends and family (5.7%, 167), or digital training workshops (1.19%, 35). This highlights the perils of an unregulated

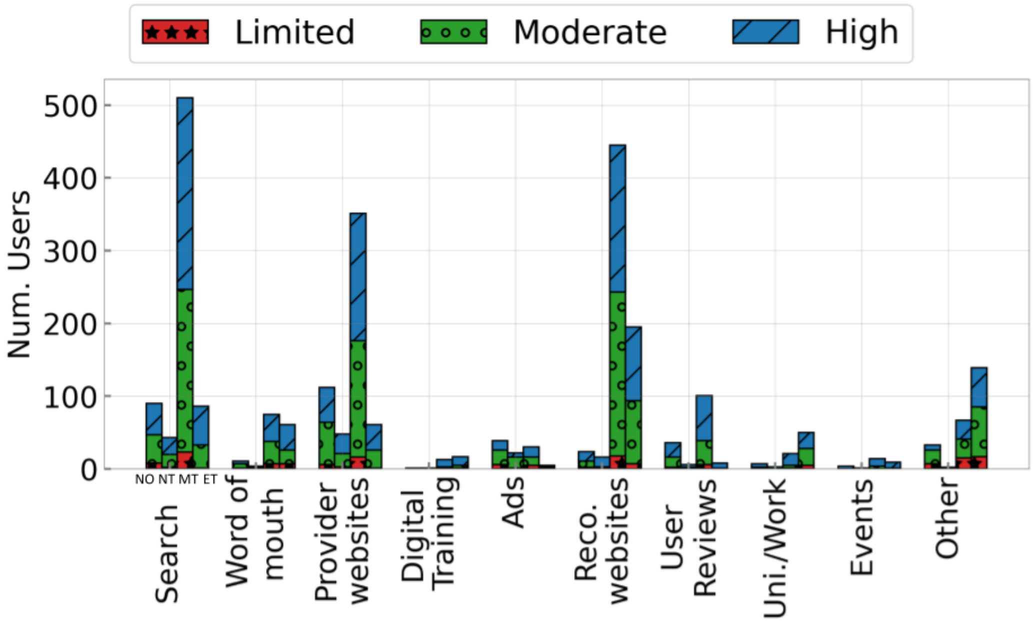


Figure 3.4: Trustworthiness of each resource as rated by users with different security and privacy expertise. Bars are No Opinion, Not-, Moderately- and Extremely-Trustworthy.

advertising and marketing ecosystem around VPNs, as we expound in §3.5.2.

Users rate recommendation websites as trustworthy sources. Interestingly, among the top three resources they use, more users rate recommendation websites as trustworthy compared to the other two; 93.9% (665 of 708) of them rate them moderately to extremely trustworthy. Figure 3.4 illustrates how users rate the trustworthiness of each of the resources. Notably, a high proportion of users whose work or school provides their VPN service rank it extremely trustworthy (61.1%, 55 of 90), highlighting that these users expect work/university VPNs to be of a high-quality.

Interestingly, 281 users use the "other" option to write-in other resources they may have used. From our qualitative coding of these responses, we notice that the VPN being offered as part of a software/security suite is the most common response (36.3%, 102 of 281). Other responses include: trusted service provider recommendations (9.6%, 27), and prior experience (5.3%, 15). Appendix B.3.2 contains all the codes.

Population	Safety without VPN		Safety with VPN	
Subscription	VS/SS/NO/**SU/VU**	**U%**	**VS/SS**/NO/SU/VU	**S%**
Paid/premium	27/243/73/**(491/156)**	**65.4**↑	**(350/538)**/44/38/20	**89.7**↑
Free	9/37/20/(78/13)	58.0	(22/97)/**24**/11/3	75.8
Uni.&Write-in	10/36/12/(37/8)	43.7	(35/42)/18/7/0	75.5

Table 3.3: Number and % of users with different subscription types and their feeling of safety without and with a VPN (from VS-Very Safe to VU-Very Unsafe). ↑ indicates more likely than the other subgroups for that column.

3.4.3 RQ3: Emotional connection and Threat model

To understand if users attach emotional considerations such as a feeling of safety with using a VPN, we first ask them their perception of safety when browsing without a VPN and then, with a VPN. We find that there are significant differences between users that use different VPN subscription types (paid, free, and university and other) and their perception of safety without a VPN, (χ^2-test, $p = 0.0001$, N=1250). We explore differences between users with varying expertise levels in Appendix B.2. **Users indicate they feel unsafe without a VPN, especially those who use paid/premium VPNs.** Overall, users indicate that they feel unsafe (62.6%, 784 of 1,252) browsing the Internet without a VPN. Interestingly, we find that paid/premium VPN users are significantly more likely to feel at least somewhat unsafe when browsing without their VPN (65.4%, 647 of 990) as compared to users that use university and other VPNs (43.7%, 45 of 103).

Paid VPN users are more likely to feel safe with their VPN, while free VPN users likely to indicate no opinion. Subsequently, there are also significant differences between users with different subscription types and their perception of safety with a VPN, ($p \ll 0.001$, N=1250). While large sections of all populations feel somewhat or very safe using a VPN (86.7%, 1,086 of 1,252), we find that paid/premium users are significantly more likely to indicate they felt safe when using their VPN (89.7%, 888 of 990), compared to free VPN users (75.8%, 119 of 157) and university/other users (75.5%, 77 of 102), who are significantly less likely. Free VPN users are significantly more likely to indicate no opinion about security (15.3%, 24 of 157) as compared to the 4.4% of paid users alone (44

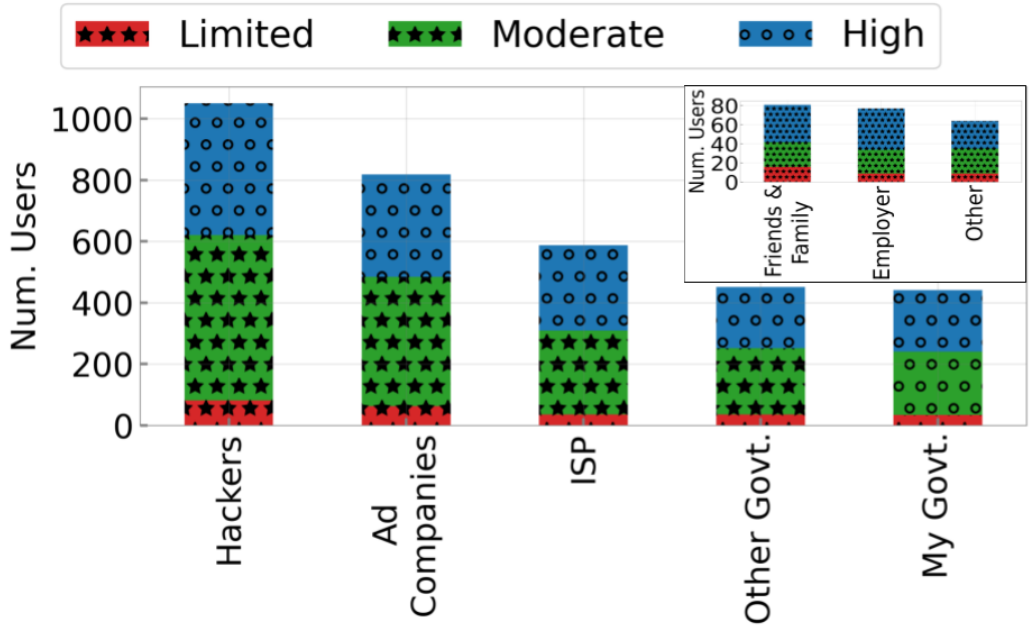


Figure 3.5: Entities from whom users with different security and privacy expertise want to protect their online activity.

of 990), shown in Table 3.3. Overall, we find that a large number of users attach emotional considerations such as safety with VPN use, and hence are likely to continue using VPNs, according to prior work studying retention [150].

A majority of users use VPNs to protect and secure their online activities. To understand users' threat models when it comes to using a VPN, we first ascertain whether users use a VPN to secure their online activities, and if yes, who they want to protect it from. Notably, 91.5% (1145 of 1,252) of users indicate they use VPNs for securing or protecting their online activity. When exploring who users aim to protect themselves from, we find that hackers/eavesdroppers on open WiFi networks (83.9%, 1,051 of 1,252), advertising companies (65.4%, 819), and internet service providers (ISP) (46.9%, 587) are the top three responses. Notably, only ≈30% of users are concerned about the U.S. government or other governments. This is intriguing because post Snowden's surveillance revelations in 2014, more users moved towards privacy tools such as VPNs and anonymity tools such as Tor [208]. Our results indicate a shift in user's attitudes, and show a growing concern towards corporate and advertisement surveillance. This shift could be due to the influence of the marketing around VPNs and the security advice to which users are exposed. Prior work also shows YouTubers often cite "the media" and "hackers" as common adversaries [7]. Figure 3.5 shows the number of users for each of these options.

High expertise users more likely to list their ISP in their threat model. We test each option independently to see if there are significant differences between users with varying expertise. We find that significantly more high expertise users indicate their ISP as one of the reasons (54.4%, 278 of 511), as compared to other users (43.3%, 273 of 631 moderate-, and 32.7%, 36 of 110 limited expertise users) ($p \ll 0.0001$, N=1252). While no significant difference was found between users selecting advertising companies (χ^2, p=0.157, N=1252), significantly less proportion of limited expertise users indicate that hackers and eavesdroppers are a concern (73.6%, 81 of 110) as compared to 85.6% (540 of 631) of the moderate- and 84.1% (430 of 511) of the high expertise users, as confirmed by a χ^2-test (p=0.00695, N=1252).

3.4.4 RQ4: Mental Model

To evaluate users' mental model of VPNs, we ask them a scenario question which aims to elicit their understanding of the protections VPNs actually provide. In the given scenario in the question, the user concluding that their ISP learns what websites they visit while connected to a VPN indicates a flawed mental model.

Almost 40% of users have a flawed mental model. We find that a high portion of users (39.9%, 500) have a flawed mental model and believe their ISP can see the websites they visit over the VPN. Worryingly, we see no significant difference between users of different expertise based on the χ^2-test (p=0.0927, N=1252). Our results are concordant with previous work which find that users, even experts, have misconceptions about the protections certain tools offer [217, 17]. We initially also considered the 135 users who answered "Nobody [can see what website I visit]" as having a flawed mental model. But we instead opt for a conservative approach and did not include them because four users clarified their response using the textbox accompanying this question. They state that since their VPN says no logging, tracking, or sharing, *ideally* nobody should know what website they visited.

Limited expertise users are more likely to have an unclear mental model, while high expertise users more likely to add insightful details. We find significant difference between users that chose "I don't know" to this question, based on a χ^2-test ($p \ll 0.00001$, N=1252). We find that users with limited expertise are more likely to choose "I don't know" (30.9%, 34 of 110 users) compared to 16.3% (103 of 631) of the moderate- and 5.5% (28 of 511) of high expertise users. High expertise users are significantly more likely to use the "other" option and write-in their answer (14.9%, 76 of 511) as compared to 8.9% moderate- and 1.8% limited expertise users (p= 0.00003, N=1252). Analyzing these write-in responses, we find that high expertise users add insightful details such as *DNS providers* knowing what websites user visits, and *site owner* learning about the user using logins or cookies. They identify other threat actors such as the site's partners, search engine used to navigate to the site, government agencies, and browser fingerprinters; all of the codes are presented in Appendix B.3.4.

To understand if VPN users have a good idea about the data VPNs can collect about them, we present many options and ask users to indicate the various kinds of data they think a VPN provider collects about them. During the analysis, we bucket these options into: *typical*, *dangerous-unreasonable*, *miscellany*, *not sure*, and *custom input*. While the last two are self-explanatory, "typical" includes demographics and account holder information, VPN servers connected to, timestamps at when VPN is in use, and device type. We consider them typical since the data is readily available to a VPN provider. The "dangerous-unreasonable" bucket includes: private messages, audio/video recordings, and keystrokes from device, all of which are not usually collected by a VPN provider, unless they are operating a malicious service, while "miscellany" includes website visited, geolocation, and interests for ads. While a reasonable provider would not collect this type of data, it is possible that some

Expertise	NotSure	**NS%**	Typ/Dang/Misc/O.	**Typ.%**	**Dang.%**
High	132	25.83	326/35/217/58	86.02	9.23
Moderate	304	48.18	292/44/220/32	89.30	13.46
Limited	68	61.82	35/18/36/3	83.33	42.86

Table 3.4: Number and % of users who indicate the types of data they think VPN providers collect. Users can choose multiple options, and we exclude users who chose "not sure" (NS) from the other counts.

VPN providers do collect them.

At least 40% users indicate they are unsure what data is collected, and ≈13% of the remaining users think unreasonable kinds of data are collected by VPNs. We find that 40.3% (504 of 1,252) of users indicate they are not sure what data is collected, limited- (61.8%, 68 of 110) and moderate expertise users (48.2%, 304 of 631) are significantly more likely to indicate uncertainty as compared to 25.8% (132 of 511) of the high expertise users (χ^2, $p \ll 0.0001$, N=1252). We exclude these users from the analysis and from the remaining 748 users, we see that in general users believe typical data (87.3%, 653) is collected by VPN providers. However, 13% (97 of 748) of users think VPNs collect dangerous-unreasonable data. The fact that users of all expertise levels have this belief, reiterates the need for better, more effective user education. Table 3.4 summarizes these results.

Finally, we explore the reasons why users think such data is being collected by VPN providers. A majority of respondents (79.2%, 992) believe the main reason is for internal analytics and quality of service reasons. Interestingly, significantly more limited expertise users believe that the data is being collected for advertising (36.4%, 40 of 110), as compared to 20.4% of moderate- and 16.4% high expertise users (χ^2,p=0.000014, N=1252). A significantly high portion of limited expertise users also believe data is used for user tracking (36.4%, 40, p=0.019), and selling to third parties (33.6%, 37, $p \ll 0.0001$), highlighting that limited expertise users believe VPNs use data collected about users for monetary benefit.

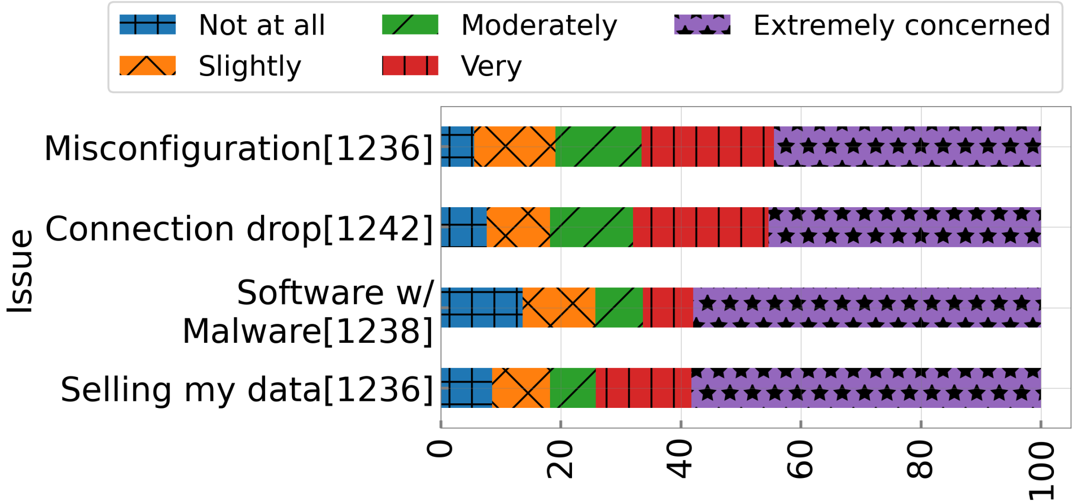


Figure 3.6: Users indicate their concern levels towards VPN-related issues, with the number of users who answered each.

3.4.5 RQ5: Perception and Trust

In order to understand users' perception of the VPN ecosystem and its issues, we ask users to rate their concern levels towards VPN related issues. We find that users are very or extremely concerned about VPN providers selling their data (73.2%, 917 of 1,252), and the VPN software containing malware (65.6%, 821). Users also express higher degrees of concern towards more technical issues such as VPN software failing without warning (67.4%, 884), and misconfigured VPN services (65.7%, 823), illustrated in Figure 3.6. We find no statistically significant differences between users of different expertise or subscription types for these options (χ^2, p»0.05).

Finally, we ask users what level of importance they associate with efforts that VPN providers undertake to earn and increase trust from the user base. We find that users consistently rate security protocols and disclosure of breaches (62%, 776 of 1,252) as an extremely important effort, followed by having a clear logging policy (46.7%, 585), and independent security audits (41.6%, 521), as shown in Figure 3.7. While there may be other efforts that we do not list, we hope that VPN providers and researchers use these insights gleaned from the users' perspectives to inform their future efforts and campaigns to secure

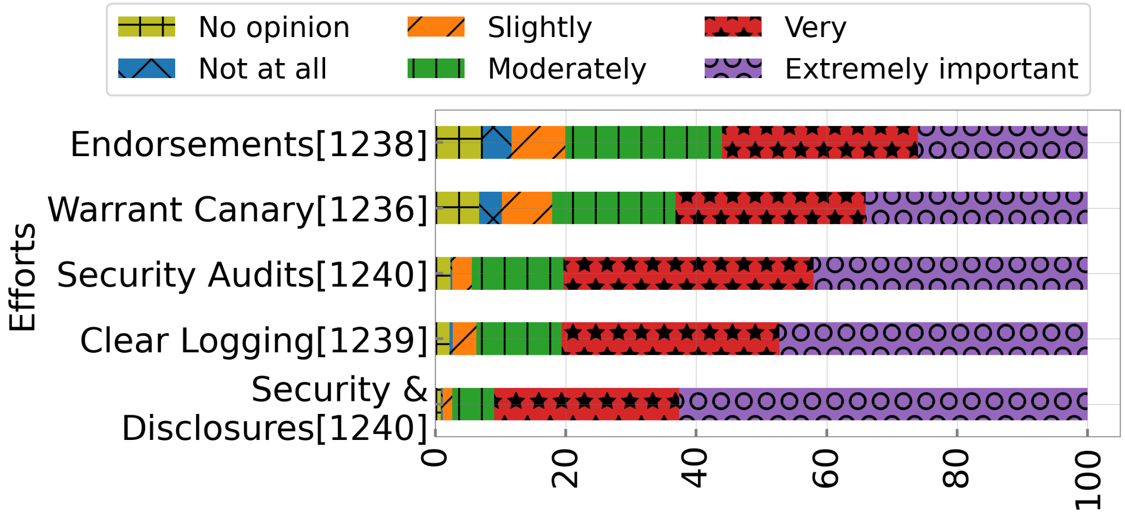


Figure 3.7: Users indicate the importance of trust-increasing efforts by VPNs, with number of users who answered each.

and foster user trust.

3.5 Perspectives of the VPN Providers

In this section, we present exploratory results from our VPN provider interviews and summarize the key issues and themes, with number of providers per theme in brackets. We compare these insights with results from the user survey, and highlight the key areas where the two are misaligned.

3.5.1 Key Themes

Key Efforts. We learn from providers that they focus on cross-platform security development (6/9), product simplicity (4/9), and usability (5/9) of their product. They also mention that they try to be reliable, gain trust over time (5/9), and practice transparency (5/9). We also noticed many VPN providers mentioned offering additional features, such as filtering, ad- and tracker-blocking similar to anti-virus software, indicating that VPNs are evolving beyond their normal functionality to retain users. From a mental model perspective, this could potentially be harmful as it sets an over-expectation of security and privacy, while users are already unclear about protections that standard VPNs offer them.

High-level Challenges. When asked about the biggest challenges in the industry, providers explain that *building trust* (6/9) is hard because there is a large number of providers and little transparency. We find that providers agree that problems generally stem from lack of trust, focusing on features and not privacy, and overestimation and overselling of service. Providers also mention that users do not understand risks (7/9), and that it is their responsibility to do better in user education and ensure honesty in their disclosures to users.

User Base. When asked about their user base and whether they conduct studies to understand them, almost all providers explain that having a *privacy-focused service deters user studies*, and that they try not to learn about their users (7/9). Instead, they typically depend on inbound user feedback such as in-app surveys or support tickets. We notice that commercial providers mention that they prefer privacy-centric users, and that western users are more likely to be paying customers.

Pricing & Marketing. Providers mention that development, labor and marketing are the main factors affecting pricing (5/9). Other factors include deals with server and cloud providers, organization build, technical means, and infrastructure. They mention that growing the user base is imperative as it creates economies of scale. Providers also note the existence of malicious practices around discounts that are not user-friendly (5/9), like marketing gimmicks to lock users. Multiple providers remark that it is the norm of the industry (3/9), one of whom says:

> "I think it's not good for consumers but why everyone does it, because everyone else does that."

A majority of providers agree that marketing plays a big role; noting that the marketing costs are high, and the competition is harsh. Regarding marketing methods, many providers mention that they do ethical marketing by being involved with the user community, relying on user reviews and word of mouth.

VPN Review Ecosystem. We discover that a main theme from the interviews is the issue of the VPN review ecosystem. One provider calls it a “parasitic industry” and a majority of providers (6/9) remark that the review ecosystem mostly runs on money, e.g. paid reviews, and cost-per-action (CPA). They also explain that VPNs or their parent companies may own different review sites [191], many review sites even *auction the #1 spot*, and do reviews for money. Multiple providers also mention that Google search results are unreliable, and that there are few good reviewers left; one provider says:

> “You honestly cannot find even one ranking site that is honest, if you just tell people that...so that people know”

Dark Patterns in the Industry. Another recurring theme was about dark patterns in the industry. Since most of these patterns are usually not readily apparent to users and researchers, we also explicitly ask a question about them. We divide the issues mentioned by various providers into:

Operational Issues (7/9): These include VPN providers having anonymous or unknown owners, having deceptive subscription models, and tracking users on their own sites, which was also highlighted in a recent report [156]. Providers also remarked on aggressive and unethical marketing such as retargeting users with VPN ads, and relying on users forgetting to cancel subscriptions. On the other hand, providers mention that VPNs get attacked as well (by other providers, bad users, and by those who abuse free VPN services).

Malicious Marketing (6/9): Providers mention several issues, that we term as *malicious marketing*, including the use of affiliate marketing, preying upon users’ lack of knowledge, and overselling of service including selling anonymity even though that is not a VPN guarantee. They also foster a false sense of security around VPNs through misinformation, fearmongering, dishonest non-expert reviews, and lying to users in disclosures. One provider, on fearmongering:

> "The best ways to get people to pay for something is to scare them and to tell them that they need security"

Factors Enabling Dark Patterns (4/9): Providers bring up several challenges that exacerbate these practices, such as the fact that the VPN ecosystem has no accountability, lacks transparency, and has few marketing and advertisement standards. Since the VPN industry is spread over multiple jurisdictions, it is hard to regulate. One provider calls it *the wild west*:

> "You know we could just say literally anything...there's absolutely no oversight. There's no one to tell you, *"Ah, you can't say that because that's not true."* There's no regulation, there's no kind of governing body"

3.5.2 RQ6: Alignment between VPN users and providers

We highlight several key areas where VPN users and providers are misaligned in their understandings and incentives, in addition to issues that both parties agree on. By highlighting these issues, we hope that technologists, and security advocates prioritize users' challenges, and focus on key problem areas. We arrange these issues from most aligned to least.

Privacy-centric Users. We note that providers explicitly mention that they prefer and cater to privacy-centric users, which aligns with the findings from our survey where over 91% of users mention that they use VPNs for security and/or privacy. Since providers mention they respect privacy and are unable to conduct user studies of their own, it is imperative for researchers to develop an understanding of VPN users.

Users' Mental Model of VPNs. Providers say that users have flawed mental models of VPNs (6/9) and our survey concurs that ≈40% of users do indeed have a flawed mental model. Providers and the security advocacy community should hence place high priority on user education. Providers mention that challenges in improving users' mental models include the lack of positive reinforcements (visual signs that a VPN is working), constant exposure to negative experiences (increased encounters of CAPTCHAs, media sites blocking VPN use), and striking a balance in technical communication. We emphasize that user-onboarding, clear communication, and responsible advertising are key drivers for change.

Importance of Pricing. From our user survey, we see that pricing is among one of the highest priorities for users, especially for limited-to-moderate expertise users. However, providers on the other hand mention that certain malicious marketing gimmicks are often used—such as fearmongering, fake countdown timers, and being always on sale—to lock-in users. We fear that since pricing is key for users, malicious tactics used by certain providers may chain users to a service that may not necessarily meet security standards. We strongly urge that advocates focus on regulations to protect consumers.

Users' Reliance on Review Sites. Despite most providers agreeing that the review ecosystem is not objective about the services and is instead largely motivated by money, our survey shows that users strongly rely on them and believe they are trustworthy. Though our survey studies only U.S. users, the VPN providers believe that the western population (including U.S.) are more likely to pay for their subscriptions. It is important to deter the exploitation of these users by informing them of the nature of the review ecosystem and how the reviews and rankings are made. As we highlight from the providers' interviews, a lot of the malicious marketing preys on users' misunderstandings. Hence, shedding light on these behaviors in the review ecosystem is crucial to ensure that they do not continue profiting off users via paid reviews and CPA. One provider says:

> "[Running costs have reduced] in the last 10 years, yet [VPN] prices are all

> the same. Why is that? Well it's because the VPN review sites are getting all the money."

Users' View on Data Collection. We find that over 40% of users are not sure exactly what data is being collected about them by VPN providers. Of the remaining users, we find that 13% think that VPNs collect dangerous or unreasonable kinds of data. On the other hand, multiple VPN providers say that they clearly communicate their logging practices, or that they do no logging and have audits to prove it. From our survey, we also find that having a clear logging policy is among the top important indicators for increasing trust with users. Alongside improving users' mental models of how VPNs work, this is another key issue that VPN providers can address by better informing users about their operation.

3.6 Broader Impact: Actionable Recommendations

Traditional approaches to regulating including standardization by government bodies may not be the best solution for VPNs because the providers and VPN servers span multiple jurisdictions. Another approach can be self-regulation within the industry. However, though coalitions look good on paper, it is necessary to bring enough providers together, and ensure oversight in order to hold these coalitions accountable. One provider, on why having such an alliance is hard:

> "[VPN providers] don't want to be held accountable for the [mistakes] of other providers...there's not a lot of trust."

Even if providers do form coalitions, we find that they do not really hold to their own self-regulated principles. In our prior work, we also find that the lack of regulation and standardization leads to VPN providers offering varying levels of security and privacy [184].

We strongly recommend that FTC and other government organizations exert oversight on VPN advertising and curb malicious tactics used by VPNs, because such aggressive and misleading ad campaigns could degrade users' mental models about VPNs. An example of successful oversight is NordVPN's ad being banned in the UK for misleading users [35]. In addition, we advocate for coordinated efforts from the industry, academia, and consumer protection organizations to bring attention to the flawed VPN recommendation ecosystem.

Finally, our study also shows that user education campaigns regarding VPNs and the VPN ecosystem must be prioritized. We find key areas that need the most improvement: users' mental model of what a VPN provides, what data it can collect, and the threat models for which VPNs can be most useful. Since the user population surveyed in our study is on average older and more educated, our results suggests that incomplete and flawed mental models may be even more prevalent among the general U.S. population. We urge security and privacy advocates such as the EFF and CDT, consumer protection agencies such as the FTC, and community initiatives such as IFF to devote their efforts towards VPN user education, raise awareness, and advocate for VPN industry oversight.

3.7 Discussion & Conclusion

VPNs have quickly gained popularity as a security and privacy tool for regular Internet users. Commercial VPNs are now a multi-billion global industry with numerous VPN providers, and apps on almost every platform. In our interviews with them, multiple providers mention that setting up a VPN and offering a service is not technically difficult, especially with the existing open source solutions [164, 45], and highlight that many VPN companies have unknown or anonymous ownership. One provider says there is a low bar to entry:

> "Technically it's not that hard to run a VPN...two people in a basement with a half decent power....can run a VPN."

For users however, exposure to risk of surveillance, reports of ISPs selling data, and increasing access restrictions have all led to an increased awareness of online risks. VPNs are marketed as technological solutions to many of these issues, though not all users will be able to verify these claims. In simple terms, a user using a VPN is simply transferring trust, say from their Internet provider, onto the VPN provider. Internet service providers (ISPs) have been around for longer and have many regulations globally. However, such regulations and advocacy has not yet caught up to the VPN industry.

In this work, we conduct studies on VPN users and providers and present actionable recommendations on important problem areas in the VPN ecosystem. Our interviews with VPN providers helps open up communication between academia and companies developing privacy-enhancing tools, which can lead to transfer of knowledge, foster collaboration, and help develop solutions for issues in the ecosystem that ultimately impacts users. We highlight that understanding real-world user needs and requirements can help shape future research focus. We hope that by shedding light on issues such as the ones rampant in the VPN review ecosystem, we raise awareness and encourage investigation, advocacy, and regulation to improve the entire VPN ecosystem for the better.

CHAPTER IV

Building VPNalyzer: Systematic, Semi-Automated Investigation of the VPN Ecosystem[1]

Internet service providers, advertisers, and online threat actors are increasingly disrupting, tampering with, and monitoring Internet traffic [237, 114, 219, 52]. High-profile security incidents, widespread reports of ISPs selling data about their users, and the increasing prevalence of geographic discrimination have fueled an increased public awareness of online risks and access restrictions [133, 28, 209]. As a result, the use of virtual private networks (VPNs) has been growing rapidly, not only among activists and journalists but also among average users [13, 26, 74, 127]. This trend has been further accelerated by more people working from home due to the COVID-19 pandemic [75]. Notably, statistics from Egypt, France, the UK, and the US point to a surge in VPN adoption over the past year [132].

Despite being a growing multi-billion dollar [151] industry, the VPN ecosystem remains severely understudied. Previous security evaluations of VPN products [89, 103, 69] have been limited in the scale and types of VPN products analyzed and have used inconsistent heuristics that prevent monitoring of issues in the VPN ecosystem over time. Specifically,

[1] This chapter is based on: Reethika Ramesh, Leonid Evdokimov, Diwen Xue, and Roya Ensafi. 2022. VPNalyzer: Systematic Investigation of the VPN Ecosystem. In Proceedings of the Network and Distributed Systems Security (NDSS) Symposium 2022 [184]. This project was supported by the Open Technology Fund, and the Consumer Reports Digital Lab Fellowship.

the latest reliable investigation into the VPN ecosystem was performed in 2018, as a one-time study of mostly free and trial versions of commercial products [103]. These previous studies, though valuable, all involved a large amount of manual effort.

The VPN ecosystem is extremely dynamic, with constant changes in the features offered with new providers frequently entering the market. This means that a large-scale and continuous empirical assessment of the VPN ecosystem requires methodology that can scale easily across many providers and can be repeated across time, thus making manual investigation as in previous work impractical. Any solution should ultimately empower researchers and average users with an extensible and convenient tool that facilitates investigation into VPN providers.

In this work, we present ***VPNalyzer***—a system that enables systematic, semi-automated investigation into the VPN ecosystem—and perform large-scale empirical assessments of 80 popular VPN providers using *VPNalyzer*. As part of the *VPNalyzer* system, we build a cross-platform tool that has a comprehensive measurement test suite combined with a simple installation and user interface. *VPNalyzer* is also designed to be modular and configurable to facilitate additions and upgrades to adapt to frequent changes of the VPN ecosystem. Our tool is equipped with 15 measurements that test for aspects of service, security and privacy essentials, misconfigurations, and leakages including whether the VPN has implemented an effective mechanism to protect users during tunnel failure. All in all, we cover essential tests from previous work, with six measurements that take direct inspirations, and nine new measurements for which we implement our own methods.

Using the *VPNalyzer* tool, we conduct the largest state-of-the-art investigation into desktop VPNs on both MacOS and Windows, which includes free and paid VPN providers, as well as self-hosted VPN solutions, and our institutional VPN. In total, we have 230 experiments from 80 unique VPN providers. This study, in addition to contributing valuable insights about the providers, highlights the value and effectiveness of *VPNalyzer*.

Our investigation reveals several previously unreported findings highlighting key issues and implementation shortcomings in the VPN ecosystem. Surprisingly, we find that a majority of VPN providers do not support IPv6, and worse, five providers even leak IPv6 traffic to the user's ISP, including our own university VPN. We find evidence of traffic leaks during tunnel failure in 26 VPN providers which seriously risk exposing sensitive user data, especially to adversaries such as governments and ISPs that are capable of inducing such failures. More specifically, we are the first to measure and detect DNS leaks during tunnel failure, which we observe in 8 providers. Further, we observe that multiple VPN providers use the same underlying infrastructure, making colocated servers easier to block, and 29 providers (including paid ones) configure clients to use public DNS services. Two providers do not tunnel all user traffic in their default configuration, which deviates from users' expectation. Finally, we conduct a case study testing custom "secure" configurations of 39 top providers. Alarmingly, even in their secure configuration, 10 VPN providers leak traffic, six of which *even had a "kill switch" feature enabled*. These results are shocking considering that these VPNs are popular, with millions of users that trust them with sensitive data.

VPNalyzer is designed to empower researchers and users and to be easily adoptable as a user-friendly tool empowering the community to be vigilant about issues in the VPN ecosystem. *Consumer Reports*, a leading consumer research and advocacy organization, used *VPNalyzer* as part of their efforts to produce a data-driven and reliable recommendation for their millions of users [70, 72, 71]. Following our future public release, we hope that *VPNalyzer* benefits users and helps the general public choose better VPN providers.

4.1 Background and Related Work

The VPN ecosystem is especially dynamic due to the constant influx of new providers and frequent changes in their popularity. This has been attributed to a variety of factors, such as increasing user demand, varying censorship trends, prevalence of geographic restrictions on content, countries banning VPN use, providers' loss of reputation, and companies being acquired or rebranded [13, 74, 127, 132, 133, 99, 9].

Users of VPN products get conflicting advice from online recommendations and often lack the time and knowledge to conduct evaluations of their own. Moreover, VPN providers often employ marketing tactics that use jargon and exaggerated claims, making it hard for users to discern what is true. Different VPN providers also offer a variety of subscription models: free services, freemium models (sometimes with limited features), and paid VPN services that range anywhere from about $5–$20 a month, often with discounts for long-term plans. Although there has been a plethora of reports ranking these commercial VPNs, they are either limited, or lack objectivity and use inconsistent heuristics to evaluate VPN providers [129, 62]. There is a dearth of trusted, objective reviews and the few that exist are limited in scale, only capture a snapshot of the VPN ecosystem at the time, and are not repeated across time.

A small number of prior academic studies and closely related research efforts have analyzed VPN products. These studies, though limited, have been adept at identifying problems plaguing the ecosystem. In 2016, Ikram et al. performed static and dynamic analysis of 283 VPN permission–enabled Android apps and revealed serious privacy and security issues, including instances of malware in VPN apps' source codes [89]. In 2018, Khan et al. conducted an empirical analysis of 62 commercial VPN providers and found that many VPNs leak user traffic through a variety of means [103]. But they also found that commercial VPN providers are less likely to intercept or tamper with user traffic than previously studied forms of traffic proxying. However, they predominantly tested providers with *free and trial versions* of desktop applications or used OpenVPN configuration files. Another study explored vulnerabilities in 30 commercial VPN products focusing on the configuration of VPN clients and software [22] and found that vulnerabilities can stem from unsafe instructions to users, insecure third-party binaries, and use of fixed pre-shared keys. These studies, while valuable in measuring and identifying issues with VPN providers, involved a large amount of manual work and hence are not *scalable*, and *cannot be repeated* *easily*.

Other studies that focus on identifying vulnerabilities that exploit leakages and privilege escalation attacks demonstrate how adversaries can use these attacks to infer the identity of the user or execute arbitrary code. Perta et al. in their manual analysis of 14 popular VPN providers, identify developer-induced bugs and misconfigurations which lead to IPv6 and DNS leaks, which could deanonmyize users [173]. Fazal et al. showed how an attacker could penetrate into the VPN tunnel by exploiting VPN clients with a dual-NIC to bypass connection to the VPN server and gain control over the VPN tunnel [56].

Further, there have been studies focusing on verifying the locations of network proxies. VPN providers advertise servers in many countries with little proof of their claims. A study by Weinberg et al. [240] found that of the 2,269 servers studied, at least one-third of them are definitely not in the geolocation or the country advertised, and another one-third of them *may not be* in the location advertised.

Finally, studies such as Netalyzr [108], IoT Inspector [80], Wehe [143] and others [118, 160, 42] paved the way for leveraging end-users to conduct network measurements, and they also provide insights to future measurement tools on effectively involving users in conducting end-host measurement [107].

The inconsistencies of online recommendations and the limitations of previous work have emphasized the need for a system that can that can help users perform systematic investigation of the VPN ecosystem. We fill this research gap with *VPNalyzer* and show its benefits and value by performing empirical assessments of 80 popular VPN providers.

4.2 *VPNalyzer* Design

Our aim is to build a system that has sufficient functionality combined with a simple installation process and user interface to empower average users to investigate different security and privacy aspects of VPN providers.

We build a cross-platform desktop application for Windows, MacOS, and Linux using

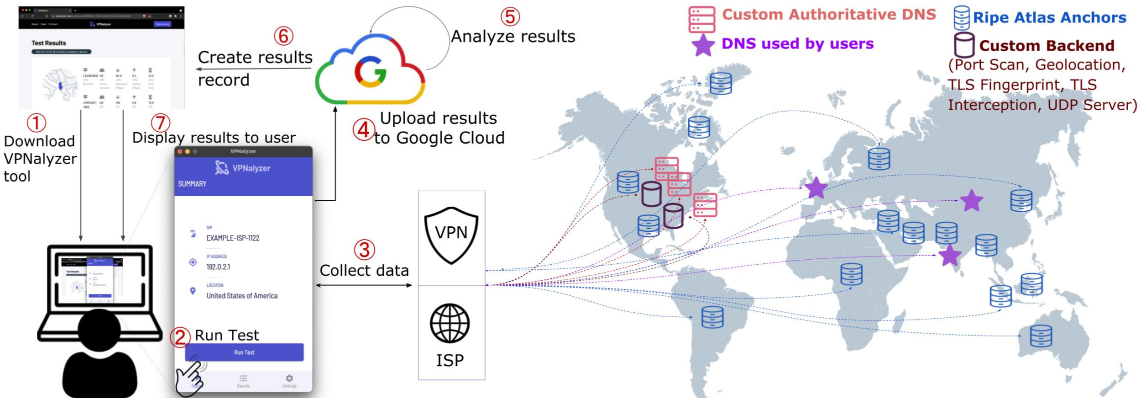
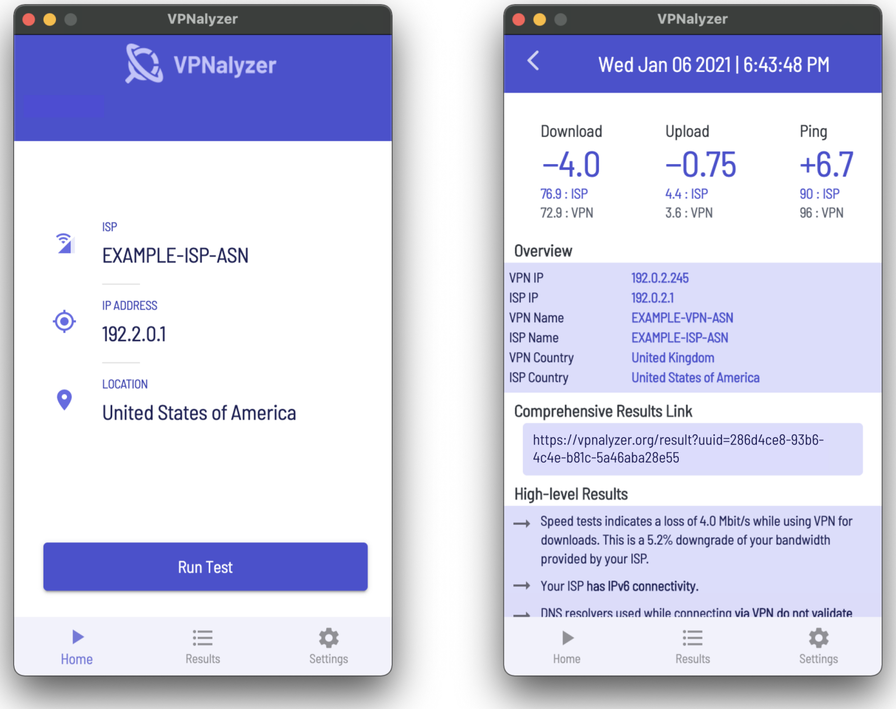


Figure 4.1: ***VPNalyzer* Architecture**— (1) User downloads application. (2) User installs the application, reviews our privacy policy, consents to be part of study. (3) User runs an "experiment" consisting of three stages: ensuring VPN is disabled and either granting or denying administrative privileges, enabling VPN and running *VPN case*, and disabling VPN and running *ISP case* (4) Once experiment is done, the application seeks explicit consent from user to upload experiment data to Google Cloud Storage. (5) Analysis pipeline works on the uploaded data. (6) Extracted results appear on website front-end. (7) User visits unique link pertaining to their "experiment" to view detailed results.

the open-source `Electron` framework [51], chosen for its cross-platform compatibility and native API availability. We develop the UI for the application with `React` and implement the measurements using `Node.js`. Initially, we explored creating a browser-based test suite, extension, or plugin. But none of these alternatives provide the level of functionality, fine-grained access for robust measurements, and convenience that a desktop application affords us. Furthermore, desktop VPNs have not been previously studied *at scale*, since methods to test them are notoriously hard to automate.

4.2.1 System Architecture and Components

Our system architecture (as described in Figure 4.1) starts with a user visiting our website to download the application for their specific platform in the form of a *.zip* file. Once the application is extracted and run, the user is first presented with our privacy policy and consent form (more details provided in §4.2.2). Upon agreeing, the user proceeds to the homepage where the user's current public IP address, Autonomous System (AS) Name, and geolocation are displayed, as shown in Figure 4.2.



ISP
EXAMPLE-ISP-ASN

IP ADDRESS
192.2.0.1

LOCATION
United States of America

Wed Jan 06 2021 | 6:43:48 PM

Download	Upload	Ping
–4.0	–0.75	+6.7
76.9 : ISP	4.4 : ISP	90 : ISP
72.9 : VPN	3.6 : VPN	96 : VPN

Overview

VPN IP	192.0.2.245
ISP IP	192.0.2.1
VPN Name	EXAMPLE-VPN-ASN
ISP Name	EXAMPLE-ISP-ASN
VPN Country	United Kingdom
ISP Country	United States of America

Comprehensive Results Link

https://vpnalyzer.org/result?uuid=286d4ce8-93b6-4c4e-b81c-5a46aba28e55

High-level Results

→ Speed tests indicates a loss of 4.0 Mbit/s while using VPN for downloads. This is a 5.2% downgrade of your bandwidth provided by your ISP.

→ Your ISP has IPv6 connectivity.

→ DNS resolvers used while connecting via VPN do not validate

Figure 4.2: ***VPNalyzer* Application**—The homepage (left) contains the user's current AS Name, public IP address, and geolocation detected using the public IP. The results page (right) contains a link to detailed results, and a summary displayed upon completion of an experiment.

Desktop Application Currently, the *VPNalyzer* test suite contains 15 "measurements" that test for aspects of service, misconfigurations, leakages, and support for a set of security and privacy essentials, (more details in §4.3). Each run of the application is termed an "experiment" that takes ≈20 minutes. An experiment flow is divided into three stages: bootstrapping in the ISP stage, performing measurements with VPN on (*VPN case*), and performing measurements with VPN off (*ISP case*). We perform the 15 measurements sequentially with the VPN and again without the VPN. This flow is necessary to confirm and corroborate our observations in the case of VPN leaks and misconfigurations. There are also essential background services, such as packet capture, that run throughout an experiment.

Building a system such as *VPNalyzer* comes with various technical challenges. Cross-platform development requires specialized knowledge especially since our tool requests administrative privileges. Designing a test suite conducive for testing VPNs, as well as ensuring that results are comparable between the VPN and ISP cases is a significant task. Further, considering the dynamic nature of the VPN ecosystem, *VPNalyzer* must be modular and configurable to facilitate additions and upgrades to the test suite. To facilitate broad distribution, our Windows and MacOS applications must be code-signed and notarized.

Experiment Flow In the *bootstrapping in ISP stage*, the user is asked to confirm that the VPN is disabled. Then, the application runs bootstrapping code and prompts the user to optionally provide administrative privileges. If granted, packet captures are initialized which allows us to investigate any ambiguous results. If not, the application skips the privileged measurements and conducts all the others. In the next stage, *VPN case*, the user is asked to *turn on their VPN*, and upon confirming, the set of measurements for the VPN connection is performed. Then, in the *ISP case*, the user is asked to *turn off their VPN*, and the same measurements are repeated for the ISP connection.

Next, the user is prompted to answer a short survey about the VPN provider they tested. With their explicit consent, the packet captures and the experiment log are uploaded, and they may decline with minimal loss of client-side functionality. Finally, the user is presented with a preliminary results page and a link to more detailed findings, as shown in Figure 4.2.

Front-end and Backend Hosts The *VPNalyzer* system uses several public and custom backend components necessary for different measurements as well as a website front-end which hosts the latest release of our application, results pages, and other miscellany about our research. We use reliable public services to serve as measurement helpers, for instance, the `RIPEStat Data API` to get public IP address and AS information and the Measurement Lab's `Locate Service (mlab-ns)` [141] to find the closest available M-Lab server.

Furthermore, we developed various custom backend components, such as authoritative DNS nameservers, port-scanner, custom UDP heartbeat server, TLS interception tester, and webserver hosting configuration files. We host backend components in academic institution subnets which are unlikely to be blocked, having multiple options for public DNS-over-HTTPS (DoH) servers, and hosting configuration files on a GitHub pages website. Considering that Google resources are unavailable in some countries, we instruct users to turn on their VPN and retry if the uploading of the experiment log fails. Additionally, to ensure availability, we implement a Prometheus monitoring system that alerts us in case any of our custom backends malfunction [179]. The complete list of public services and custom backends used for each measurement is in Table 4.1.

Data Storage and Analysis: VPNalyzer uses `Google Cloud` [66] for both storage of the experiment data and our analysis pipeline that works on the data and creates the detailed results that will be shown on our website front-end to the user.

4.2.2 Ethics Considerations and Consent

Since our system involves collecting data by running measurements from users' machines, conducting ethical measurements and following good Internet citizenship are integral parts of our design principles. First we approached our institution's IRB, which determined our study to be "Not Regulated by the IRB," as we study the VPNs rather than the user/human subjects. Aiming to set a high standard for ethical measurement, we designed our measurements and system to follow the principles described in the Menlo report [44]. We highlight some of our key considerations below.

We offer users our detailed privacy policy and consent form before they can proceed to run any measurements. These documents were carefully designed following the language used by OONI's data policy document and inspired by the Harvard CyberLaw Clinic's guide to risks of security research [162, 168]. They were also shared with colleagues experienced in such studies, and revised using their feedback, ensuring that we adequately inform and help our users make an informed decision. We also provide means for users to contact us for more information.

Measurement Type	Measurement Name	Goal	Inspired by Previous Work	Public Services and Custom Backends Used
Aspects of Service	Bandwidth and Latency Tests	Calculate the performance penalty/overhead incurred by using the VPN	[110, 141]	`M-Lab` Infrastructure
	Geolocation Test	Fetch Cloudflare's IP geolocation and collect RTT measurements to `RIPE Atlas` Anchors	[240, 31]	Custom Cloudflare backend, `RIPE Atlas` Anchors, Anchors list Updater
	RPKI Validation	Test if the user's VPN and ISP are implementing BGP safely using RPKI validation		Cloudflare RPKI endpoints
Misconfiguration and Leakages	AS Mismatch	Detect possible IP leakage using AS Mismatches	[196]	`RIPEstat Data API`
	VPN Kill Switch Test	Detect whether VPN has implemented the kill switch feature correctly		`RIPEstat Data API`, custom UDP heartbeat servers
	DNS leak during tunnel failure	Detect whether VPN providers' killswitch mechanisms leak DNS traffic		Public DNS `whoami` helpers
Security and Privacy Essentials	Port Scan	Scan to discover different services running on the VPN server using `Nmap`		Custom port scanner backend
	Router Scan	Ascertain if the user's home (ISP) router's management interface is reachable while connected to the VPN		–
	DNS Discovery	Discover all possible DNS resolvers available to the user in both *VPN case* and *ISP case*		Public DNS `whoami` helpers, `RIPEstat Data API`
	Presence of DNS Proxy	Identify if the VPN has a DNS proxy that targets all the DNS resolvers used by the user's machine		Public recursive DNS resolvers, public DNS `whoami` helpers
	Support for DNSSEC	Test if an available resolver validates DNSSEC signatures		Custom domain, authoritative nameservers
	Use of QNAME Minimization	Test if an available resolver implements qmin	[41]	Custom domain, authoritative nameservers
	Lack of support for DoH and Presence of DNS64 Resolver	Test if an available resolver intentionally signals that the network is unsuitable for DoH, and if the resolver is a DNS64 resolver		Mozilla canary domain, `ipv4only.arpa`
	TLS Interception	Identify presence of TLS interception using the certificate	[97, 103, 219]	Certificate Fetching backend, Certificate Transparency logs
	TLS Fingerprinting	Identify presence of TLS interception using TLS fingerprint	[59]	TLS fingerprinting backend

Table 4.1: **Measurements—**The goal of the 15 measurements performed by *VPNalyzer* and the custom backend and public services used for each measurement.

We provide users with the final authority on the data our application collects. For example, the application requests administrative privileges, which are necessary to run certain measurements, but users are free to decline, in which case the application skips the privileged measurements and conducts the others. The application can collect packet captures during each experiment, but users are informed and must explicitly consent before packet captures are uploaded for each experiment.

We attempt to be good Internet citizens when conducting network measurements. When public services and endpoints are part of our measurements, we use them only for their intended purposes. Since our quality of service measurement depends on running the NDT7 (Network Diagnostic Test) using infrastructure operated by Measurement Lab (`M-Lab`) [141], we obtained their consent to use their servers. We contacted and obtained permission from npcap (the packet capture library for Windows) maintainers to bundle it with our application for Windows [158]. We also bootstrap most of the measurements in the ISP stage, due to the consideration that users may be paying for the VPN bandwidth.

Some countries have laws that prohibit using VPNs, so we explicitly inform users to be cognizant of these restrictions before using *VPNalyzer*. Since some tests are run without the VPN enabled, we inform users that their local ISP and VPN provider, and possibly their government, will be able to detect that the user is running *VPNalyzer*.

Finally, with respect to the issues we discovered, detailed in §4.5, we have contacted the VPN providers and are in the process of disclosing our findings to each of them.

4.3 *VPNalyzer* Measurement Test Suite

In this section, we describe the 15 measurements that are performed during each experiment in the application. Every experiment is assigned a universally unique identifier (`UUID`), which is an RFC 4122 version 4 `UUID`, and the application creates an "experiment directory" named after the `UUID` under the application's data directory path. This directory contains the experiment log file and, if the user grants administrative privileges, the packet capture files.

Previous studies have identified key issues, known security and privacy flaws, leakages, and best practices for VPNs [103, 89, 173, 41, 240]. Our *VPNalyzer* test suite covers all the essential tests from previous work as well as implementing new measurements. While these tests are comprehensive and significantly improve the testing methods used in previous work, they do not necessarily cover *all* aspects of a VPN. For instance, we do not compare

VPNs based on their usability or measure the logging policies of a VPN.

Our focus is on testing for a breadth of important issues and facilitating collection of extensive data about the ecosystem from the end user's machine. Going in-depth and investigating edge cases, while interesting, do not justify the additional complexity. For instance, our *support for DNSSEC* measurement checks if a resolver validates DNSSEC signatures but does not compare resolver behaviors with respect to different DNSSEC algorithms. Further, our measurement results are meant to be informative for the user, and for the tests that need further inspection such as the kill switch test and geolocation, we leverage our analysis pipeline to take extra steps such as verifying results using packet captures to prevent false positives before presenting them to the user. A short description of each of the measurements is given in Table 4.1.

4.3.1 Aspects of Service

Bandwidth and Latency Tests: This measurement is designed to calculate the performance overhead incurred by using the VPN. We first find an `M-Lab` server closest to the VPN and use it to run several ndt7 (Network Diagnostic Tool) measurements over IPv4 and IPv6 in the *VPN case* and the *ISP case*. Finding an `M-Lab` server closest to the VPN is important in order to allow a fair comparison of the quality of service. If the server position is chosen based on the user's location, it may cause a bloat in the measured performance overhead due to the trombone effect [32].

We calculate the performance overhead by comparing the bandwidth and latency between the *VPN case* and the *ISP case*. We choose ndt7 over other public performance-measuring services such as iPerf3 or Ookla because we are not interested in measuring last-mile speed or approximating a browsing experience. Rather, we want to estimate the performance that is obtainable when the user is connected to a particular VPN server as compared to when they are connected to their ISP [110].

Geolocation Test: This measurement is designed to estimate the geolocation of the

VPN server. Instead of using free IP-to-geolocation databases, which are notoriously unreliable and contain many errors especially regarding VPN services [63, 174, 240], we opt to conduct this measurement using the following two methods.

First, we leverage Cloudflare's IP geolocation service offered to site owners where the country code (in ISO 3166-1 Alpha 2 format) of the visitor's IP is included in the header of each request. To that end, we hosted a custom webserver at our University and added the domain to Cloudflare's free plan tier. The measurement makes a request to the domain and our webserver extracts the information from the `CF-IPCountry` request header [31]. Since this same information is typically used by Cloudflare to offer geo-specific services, we believe it is a good approximation to use for our purpose as well.

Our second approach uses the Weinberg et al. [240] upgraded Constraint-Based Geolocation algorithm (CBG++) by collecting round-trip time measurements to hosts in known locations. Similar to previous work, we use `RIPE Atlas` anchors as the "landmark" hosts since they are reliable, their documented locations are accurate, and conveniently, they continuously publish public databases containing the round-trip times between each other. Currently, there are over 723 `RIPE Atlas` anchors, and given their added reliability as compared to `RIPE Atlas` Probes, we choose to use only the anchors, which we fetch and update on our webserver daily.

Our measurement initiates a TCP connection to port 80 of each anchor, due to knowledge that a large number of VPN servers ignore ICMP ping requests and others drop TCP and UDP packets to unusual port numbers. Upon receiving a response from the anchor, it records the round-trip time for each query and saves it in the experiment log for future analysis.

The application displays the country name as reported from our Cloudflare custom webserver. Considering the amount of computation required for the Weinberg et al. method, we plan to publish them in an aggregate form because providing its result in real-time to the user is not feasible.

RPKI Validation: We conduct this measurement to test if the user's VPN provider (and ISP) implement the Border Gateway Protocol (BGP) safely. BGP is vulnerable to leaks and prefix hijacks, which can be mitigated by Resource Public Key Infrastructure (RPKI) [10]. The absence of RPKI validation or its incorrect implementation by a VPN provider network may allow an attacker to maliciously announce routes and disrupt traffic intended for the VPN server. To measure whether RPKI origin validation is properly implemented, we use Cloudflare's existing testing infrastructure used in `isbgpsafeyet.com`. Cloudflare provides two prefixes—one covered with a valid Route Origin Authorization (ROA) [11] and another with an invalid ROA. During this measurement, we make an HTTPS request to each of the sources. If the request to the invalid source fails while being able to fetch contents from the valid source, then it is an indication that RPKI validation is implemented in the network.

4.3.2 Misconfigurations and Leakages

AS Mismatch: This measurement is designed to detect possible IP leakage using Autonomous System (AS) information. For the user's public IPv4 and IPv6 addresses, we obtain the Autonomous System Numbers (ASNs) from `RIPEstat Data API's` "Network Info" data call [196]. In the *VPN case*, if the ASes of the IPv4 and IPv6 address (where available) do not belong to the same organization, it indicates that the traffic is being leaked to a "second" AS. If the AS(es) in the *VPN case* and *ISP case* are the same, it most likely indicates an IP packet leak, or it means that the VPN provider is using the same network as the ISP.

VPN Kill Switch Test: This measurement is designed to understand if the user's VPN has implemented an effective measure to protect users' traffic during tunnel failure. The kill switch feature, as it often called, is a fundamental security measure that is used to prevent any information leakage when VPN tunnel failure occurs. If the connection to the VPN server fails for any reason, this mechanism cuts off the user's connection to the Internet in

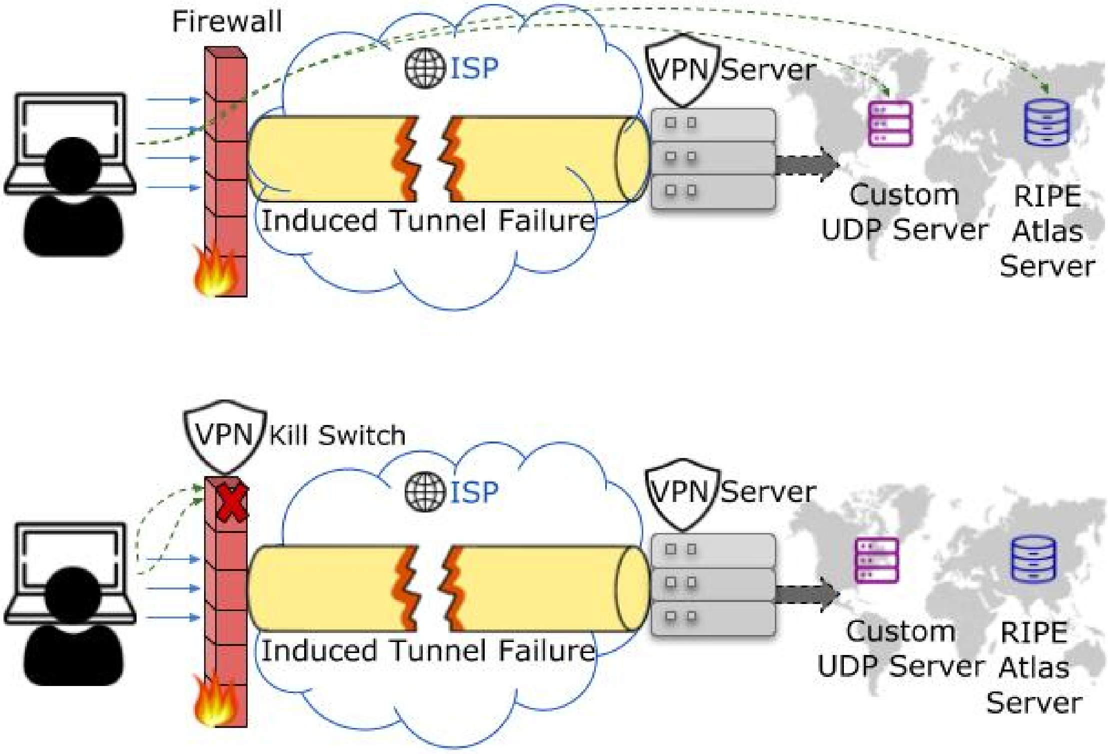


Figure 4.3: **VPN Kill Switch—** The top image depicts the scenario where the VPN kill switch feature is disabled, and the bottom image shows the kill switch feature enabled and working. Our measurement induces a tunnel failure by blocking all traffic not on our “allowlist”. When the VPN kill switch is disabled (top), the traffic to the allowlist hosts is allowed to pass through the ISP link. In the bottom case, the traffic to the allowlist is also blocked due to the VPN’s kill switch feature.

order to disallow unprotected access until the VPN is able to reconnect. The functioning of the kill switch feature is illustrated in Figure 4.3.

Conceptually, to test for failures in such a mechanism, we create an “allowlist” of certain destination hosts, and then cause a tunnel failure by blocking all traffic *except to and from hosts on the allowlist*. If the VPN kill switch is effective, the traffic to the hosts on the allowlist should *also be blocked*. While this seems straightforward, it demands complex platform-specific implementation which we do in three stages: bootstrapping, setting up firewall rules to induce tunnel failure, and finally a two-pronged approach to detect VPN kill switch failures.

In the bootstrap stage of our application, we seek administrative privileges necessary for making changes to the user's firewall. We then initiate sessions with our two custom UDP servers to begin receiving UDP heartbeats, and log the firewall's state. In the *VPN case*, prior to running any measurements, we set up the necessary platform-specific components and log the firewall state again. This setup code on Linux entails creating new chains for the `iptables`. On Windows, we log the version of PowerShell and details about the `NetSecurity` module that is used to interact with the machine's firewall. On MacOS, we check if the `pf` program allows us to create custom anchors, verify that `pfctl` functions as expected, and finally enable the packet filter and obtain a token that is necessary to revert changes made by the measurement on the user's machine.

Next, in the *VPN case* we create a "allowlist" to be added to our custom firewall rules. This allowlist contains the IPv4 and IPv6 addresses of `RIPEstat Whats My IP`, **one** of our custom UDP heartbeat servers, and authoritative nameservers and public DNS resolvers belonging to Cloudflare, Google, and OpenDNS which become necessary for both our two-pronged approach, and the upcoming *DNS leak test*. We then modify the user's firewall, apply our custom rules to induce tunnel failure *without* resetting existing rules, and log the state of the firewall again. Once applied, the firewall blocks all traffic (to any port) to and from all hosts **not on the allowlist**, meaning there can be no communication to and from the VPN server.

Lastly, after inducing tunnel failure we use a two-pronged approach to detect flaws in the VPN kill switch:

- First, for over 120 seconds, we periodically query the IPv4 and IPv6 endpoints of `RIPEstat Who am I` data call that is on our allowlist. We chose 120 seconds because OpenVPN has a timeout of 120 seconds to allow tunnel failure to be signalled [165] and the official OpenVPN Connect v3 client uses 60 seconds as the default connection timeout. If the kill switch feature is not enabled, the query will reach the endpoint and the IP returned will be the user's ISP IP address shown in Figure 4.3 (top). On the other hand, if the kill switch works effectively, the queries will time out, as shown in Figure 4.3 (bottom).

- Second, confirming induced tunnel failure requires another step of validation in case the VPN connection is governed by a stateful firewall, which typically preserves connection state once established. To that end, we use our custom UDP heartbeat servers to check if the firewall states are indeed preserved by a stateful firewall. Starting from the bootstrapping stage, our custom UDP servers, called Server$_A$ and Server$_B$, continuously send UDP heartbeats to the application and receive acknowledgements. Next, in the allowlist, we only add Server$_A$ and not Server$_B$. After the custom rules are applied, the heartbeat traffic from Server$_B$ should be blocked. We can conclude that our custom rules have induced tunnel failure (i.e. no stateful firewall) when traffic from Server$_A$ is also blocked.

In reporting the result of our *kill switch measurement*, we report leaks when the UDP heartbeat servers indicate successful tunnel failure and the user's ISP IP leak is confirmed from the `RIPEstat` endpoint's response.

Though the technique of using custom firewall rules to trigger tunnel failure has been used before, we identify a number of factors that can interfere with measurement results. Factors such as ordering of anchors in the main firewall, and VPNs inserting dynamic rules on the fly affect how the firewall rules are parsed, and hence may lead to false positives results if not handled carefully. Unfortunately, these factors were overlooked by previous studies. Specifically Khan et al. [103] triggered their blocking by resetting the test machine's firewall configuration with their own rules according to the code on their project's GitHub repository [104]. In doing so, they *possibly overrode* rules added by the VPN application, thereby effectively turning off many VPNs' kill switch mechanisms *before testing them*, leading to potential false positive results. Hence, we developed our own two-pronged approach to try and overcome such issues.

DNS Leak during tunnel failure: This measurement investigates if VPN providers' protection mechanisms during tunnel failure leak DNS traffic. Some VPN providers allow DNS queries to bypass their kill switch or firewall rules possibly to resolve domain names of their servers to attempt reconnection during tunnel failure. However, this behavior introduces privacy risks that can expose the user. For instance, during tunnel failure any third-party application on the device can send DNS queries to obtain the ISP IP of the user. Moreover, the ISP could learn the sites visited by the user via the 'A' and 'AAAA' queries made. These risks can be avoided by implementing more-secure reconnection methods. For instance, Wireguard resolves necessary DNS names during bootstrap [247].

To test for this DNS leak, we first resolve and add to our allowlist all the IPv4 and IPv6 IPs of the authoritative nameservers and recursive resolvers belonging to popular public DNS services such as Cloudflare, Google, and OpenDNS. Once the allowlist is applied as explained in the *VPN Kill Switch Test*, over the period of 120 seconds, we send two `whoami` DNS probes: one to a public recursive resolver and another to an authoritative nameserver, which are repeated every 100 milliseconds. Each probe is sent once over TCP and UDP for each round. These `whoami` probes are queries of the form: `dig +noedns -t txt -c chaos whoami.cloudflare. @one.one.one.one.` and `dig +noedns -t txt whoami.cloudflare.com. @ns3.cloudflare.com.` (similar queries to services of the other two providers) which return the public IP address in the response.

If no response is received for any of the queries, the test declares no DNS leak. Otherwise, we corroborate the received responses with collected packet captures to confirm that the user's ISP IP is returned, which we then consider a DNS leak.

4.3.3 Security and Privacy Essentials

The VPN ecosystem is largely unregulated and as a result does not have standardized requirements of security and privacy practices. In this section, we contribute a list of security and privacy essentials that we believe are basic guarantees. These are not meant to be a comprehensive checklist of features but rather a list of measurable, fundamental aspects that VPN providers should be able to fulfill.

Port Scan: Open ports on a VPN server can be used by malicious actors to identify running services that can be exploited. This measurement scans and discovers the different services running on the VPN server from our custom backend. The application first contacts the backend and establishes a token based on the `UUID`, and the obtained token is then sent to the backend server with a request to scan. Our backend upon receiving the token begins an `Nmap` scan towards the source IP address. In order to restrict the scanning time and target ports of interest, we trigger an `Nmap` scan of a total of 64 (43 TCP, 21 UDP) ports curated using knowledge from the `routersecurity.org` blog. In the *VPN case*, the result of this measurement detects the ports open/filtered and the services running on the VPN server.

Router Scan: This measurement is designed to ascertain if the user's home (ISP) router's management interface is reachable while connected to the VPN. Recent reports have found that multiple models of routers are vulnerable to Remote Code Execution attacks through the router web management interface [38, 37] both pre- and post-authentication. To prevent such web exploits against the user, the router interface should be blocked by the VPN provider.

To detect this, in the bootstrapping stage, we first retrieve the router addresses from the local routing table with platform specific code. In the *VPN case* we run a TCP banner grab, check for HTTP and HTTPS servers on the router's ports 80 and 443 and log the TLS certificate (if available), and send DNS queries to port 53.

DNS Discovery: This measurement is designed to discover all possible DNS resolvers available to the user, separately, in the *VPN case* and *ISP case*. This enables us to check whether the VPN uses either public DNS resolvers or user's ISP resolvers that exposes the users to privacy risks.

In each *VPN case* and *ISP case*, to discover all resolvers, we begin by obtaining the DNS resolvers list from the runtime `Node.js` DNS configuration. We then sequentially query public DNS `whoami` helpers operated by Cloudflare, Akamai, and Google using *each* of the available resolvers. These helpers are designed to respond to ‘A’ or ‘TXT’ record queries with the unicast IP address of the recursive resolver that queried the authoritative name server.

Using the discovered resolvers, we do basic liveness checks to ensure they are responsive. These liveness checks are done by sequentially requesting the A, AAAA, SOA, NS records of root name servers. If a resolver is able to obtain a `NOERROR` response for *at least one* of the requests, then it is marked as responsive. This gives us separate lists for *VPN case* and *ISP case* of the responsive DNS resolvers from the user’s machine, which is subsequently used for all following DNS-related tests.

Presence of DNS Proxy: This measurement is designed to identify if the VPN has a DNS proxy that targets all the DNS resolvers in the *VPN case*. A DNS proxy is typically used to mitigate DNS leaks and aims to prevent third-parties from potentially monitoring the user’s DNS queries. To measure this, we compile a list of public DNS resolvers belonging to Cloudflare, Google, Neustar, OpenDNS, Quad9, VeriSign, and Yandex. Using two randomly selected resolvers from this list, we query the public DNS `whoami` helpers operated by Cloudflare, Akamai, and Google. If the user’s VPN connection supports both IPv4 and IPv6, then both endpoints of the `whoami` helpers are queried. This yields the external IP addresses of the DNS resolver(s) making this query.

If the VPN implements a DNS proxy, the ASNs corresponding to the two external IPs should overlap, and the *overlapping ASN should match the ASN of discovered DNS resolvers in VPN case*. Note that we use public DNS resolvers with the assumption that their AS information do not overlap, but even if they do, we do not consider it a DNS proxy unless it also matches the ASN of the resolvers of the *VPN case*. A recent example of public DNS servers’ AS overlapping is Neustar’s acquisition of Verisign’s public DNS service in November of 2020[155].

Even in the presence of a DNS proxy, the *DNS discovery measurement* can still find DNS leaks if the discovered resolvers belong to an AS other than the VPN’s AS. This may happen due to a number of reasons, for instance, in the presence of an IPv6 leak or a malfunctioning DNS proxy.

Support for DNSSEC: This measurement is designed to test if an available resolver validates DNSSEC signatures, a DNS extension to ensure data integrity [86]. For this measurement, we set up a custom authoritative nameserver for a domain under our control. We create a well-configured subdomain with a valid DNSSEC signature, and a subdomain with an invalid DNSSEC signature.

First, the parent zone, say `testdomain.com` is signed, followed by signing of a well-configured subdomain (`good.testdomain.com`) and adding a valid Delegation Signer (DS) record to the parent zone. The DS record is typically generated using the key signing key (KSK) that is created for each signed zone. For `bad.testdomain.com`, we sign the zone correctly but add a malformed DS record to the parent zone. By doing so, the DNSSEC chain of trust is broken between the parent zone and the child zone, and therefore resolvers that validate DNSSEC signatures would throw a SERVFAIL error. Contrarily, a resolver that does not validate DNSSEC signatures would return records successfully for both queries.

Use of Query Name Minimization: This measurement is designed to test if an available resolver implements query name minimization (qmin). Regular DNS queries reveal more information than necessary, and qmin was introduced [88] to limit the query to only include relevant information to a DNS name server. We follow the method introduced by de Vries et al. [41] and set up our own custom test domains. The detection relies on the fact that non-qmin resolvers miss any delegation that happens before the terminal label of a query.

We set up a test domain like `test-qm.testdomain.com` similar to de Vries et al. and set up an authoritative nameserver for that domain. This nameserver *NS* will return a TXT record for the full query `x.y.test-qm.testdomain.com` containing the text “NO, QNAME minimization is not enabled!”. But, this nameserver also delegates queries to the second-to-last label, i.e. queries to `y.test-qm.testdomain.com`, to a separate nameserver say NS^*, which when queried for the full domain will return a TXT record containing the text "YES, QNAME minimization is enabled!". Qmin-enabled resolvers will find the record on NS^* whereas non-qmin resolvers that only look for terminal label will find the record on NS.

To ensure robustness, we query both our custom domain and the public qmin test service `qnamemintest.internet.nl` and report results based on both queries.

Lack of support for DoH and Presence of DNS64 Resolver: This measurement is designed to test if an available resolver intentionally signals that the network is unsuitable for DNS-Over-HTTPS (DoH). DoH is designed to encrypt the DNS queries to increase user privacy and prevent eavesdropping. Due to the implications on user privacy and discussions surrounding DoH, we are interested in detecting if the VPN and ISP networks disable DoH.

Firefox has now enabled DoH in their browsers by default for US-based users. They use a canary domain [145], namely `use-application-dns.net`, to mitigate compatibility problems for those users who get the DoH feature enabled by default. We include a measurement to query this canary domain and verify the response. A non-`NOERROR` response or the lack of 'A' or 'AAAA' records signals a lack of support for DoH.

A valid exception is when a DNS64 resolver signals lack of support for DoH. NAT64 and DNS64 technologies are generally used by networks to provide IPv4 connectivity for IPv6-only nodes. DNS64 uses "IPv6 address synthesis" to create local IPv6 addresses for IPv4-only services, thereby allowing communication between DNS-using IPv6-only nodes and IPv4-only services. Importantly, DoH standards have declared DNS64 out of their scope. In IPv6-only networks using DNS64/NAT64, third-party DoH is expected to be incompatible and hence would be a viable reason for signalling unsuitability for DoH.

To that end and since such a setup in VPNs is interesting to measure, we test for DNS64 resolvers taking inspiration from RFC 7050 [87], we use the well-known special use IPv4-only domain name `ipv4only.arpa`. All DNS resolvers are supposed to respond with a positive response for a DNS `A` resource record query for the domain and resolve it according to the specification in RFC 7050. In contrast, while requesting a DNS `AAAA` resource record for the same domain, only DNS64 resolvers reply with one or more `AAAA` resource records indicating that they utilize IPv6 address synthesis. Our measurement reports if the available resolvers are DNS64 resolvers and thus signal lack of support for DoH.

TLS Interception: This measurement is designed to identify the presence of TLS interception. VPNs are usually in a privileged position to perform TLS interception as shown by previous work [89]. These interceptions are made possible by the VPN application asking users to install their own, often self-signed, root CA certificates and later generating custom certificates for each TLS connection on the fly. Intuitively, we can measure this by comparing the certificates we fetch through the *VPN case* and the *ISP case*. However, a mismatch does not necessarily mean that the TLS connection has been intercepted because sites like Google or Facebook employ load-balancing with multiple servers containing different certificates.

Taking this factor into account, our *TLS interception test* is designed as follows. We compile a list of three domains—GitHub, Google, and Facebook—that were previously targeted by interception events [219, 97]. These domains were chosen in order to keep the measurement short but at the same time maximize the likelihood of seeing any interception occurrence. We set up a backend at our university and configure it to fetch *all* non-expired certificates for these domains on a daily basis from Certificate Transparency Logs. These certificates are trimmed down to contain only SHA256 signatures and their issuers' CNs, and are updated on our configuration repository daily. During the measurement, these "pre-loaded" certificates are fetched by the application and compared with the certificates we receive from visiting these domains directly. In addition to the three domains mentioned above, we also make two requests to a domain under our control with the SNI header modified to `google.com` or `facebook.com`. This is done based on knowledge from previous work, which showed that adversaries may intercept a connection based on a trigger keyword in the SNI [219]. This measurement identifies TLS Interception if at least one certificate's SHA256 signature and the issuer's CN does not match its counterpart in the pre-loaded set.

TLS Fingerprinting: This measurement is designed to detect the presence of a TLS-terminating endpoint. Previous studies have shown that the wide range of features supported in TLS makes it possible to fingerprint each TLS implementation and to use mismatching fingerprints to detect TLS Man-in-the-middle attacks [121, 59].

To conduct this measurement, we set up a custom fingerprinting backend hosted at our university which listens on TCP port 443. The measurement triggers a TLS handshake with the backend server and sends its experiment `UUID`. Upon receiving a new handshake request, the server will extract the ClientHello from the handshake and hash it to produce a SHA256 fingerprint. We use six fields and extensions of the TLS ClientHello to produce the fingerprint, including server name, supported curves, supported points, signature schemes, supported application protocols, and supported versions. The backend returns this fingerprint back to the application. After querying through both ISP and VPN links, the application will compare the two fingerprints and report any mismatch. A mismatch suggests the presence of a TLS-terminating endpoint in the network path.

4.4 Data Collection

Using *VPNalyzer* for our *data collection*, we aim to investigate VPN providers as well as highlight the value of our application. To demonstrate that *VPNalyzer* works under a variety of conditions, we collect experiments from different types of VPNs (free, paid, and self-hosted) selected for their popularity and market share. These VPNs use different protocols including OpenVPN, Wireguard, IPSec, and IKEv2.

Overall, we have 80 providers: 58 are paid, premium services (includes our University VPN), 18 are free services, and the remaining four are leading self-setup VPN solutions—Outline [96], OpenVPN Access Server [167], Algo [223], and Streisand [218], as illustrated in Table 4.2. The free services are selected from the top free VPNs from the MacOS App

VPN Type	Name of VPN Provider
Free Providers (18)	1.1.1.1 + Warp Cloudflare, Best VPN, Free VPN by Free VPN.org, K2VPN, Psiphon, Riseup, Star VPN: Unlimited WiFi Proxy, Touch VPN, Urban VPN desktop, VeePN, VPN Hotspot - Unlimited Proxy, VPN Owl, VPN Plus, VPN Pro, VPN Proxy Master, VPN Super: Best VPN Proxy, VPNBook, VPNLite
Self-hosted (4)	Algo*, OpenVPN Access Server*, Outline*, Streisand*
Paid Providers (58)	AirVPN, Anonine, Astrill VPN*, Atlas VPN *, Avast Secureline, Avira Phantom, AzireVPN, Betternet, BolehVPN, Bullguard*, Cactus VPN, Cryptostorm, CyberGhost *, Encrypt.me *, ExpressVPN *, F-Secure Freedome *, FastestVPN*, Goose VPN*, Hide My Ass! *, Hide.me*, HideIPVPN, Hotspot Shield*, IP Vanish*, IVPN *, Ivacy VPN*, Kaspersky*, KeepSolid/VPN Unlimited *, LeVPN, Mozilla VPN*, Mullvad VPN*, Namecheap*, NordVPN*, Norton Secure VPN*, OVPN.com, PandaVPN, Perfect Privacy *, Private Internet Access*, Private Tunnel, PrivateVPN*, ProtonVPN*, PureVPN*, Speedify *, Steganos, Strong VPN*, SurfEasy, SurfShark *, TorGuard, Trust.Zone*, TunnelBear*, Turbo VPN, University VPN*, Unspyable, VPN.ac, VPNUK, Vypr*, Windscribe*, ZenMate, ZoogVPN

Table 4.2: **VPN list—**Names and types of all 80 tested VPN providers. The 39 VPN providers included in our custom "secure" configurations case study are marked with an asterisk.

Store. For the paid services, we chose the highest tier of service (sometimes called "pro" or "advanced") offered by the provider where applicable.

All 80 providers are tested in their default, "out of the box" security configurations. Since self-hosted VPNs are dependent on client-side software, we tested each of them with their recommended software. We tested the OpenVPN Access Server with the official OpenVPN Connect Client [166], Streisand's OpenVPN offering with Tunnelblick on MacOS and OpenVPN Connect Client on Windows, and Algo and Streisand's Wireguard offering with the official Wireguard application. Outline offers its own client software.

Furthermore, we collect experiments from a subset of the VPN providers by configuring them in a "custom secure mode". VPN providers often offer extra privacy and security functionalities such as blocking third-party DNS, and kill switch. But these features are not *enabled by default* to favor performance or usability. We conduct a case study by zooming in on the 34 most popular VPN providers (all paid services) selected by combining the top VPN recommendation sites based on search engine results, listed in Table C.1. We augmented this list with the four self-hosted VPN solutions and our institutional university VPN. For this case study, we use custom secure configurations where we enabled all relevant security and privacy settings available on the VPN client software. We repeat our testing on both Windows and MacOS (not all VPN providers had client software for Linux). We report our findings from this case study in §4.5.8.

In total, including our case study, we analyzed 230 experiments from 80 VPN providers, all run before July 24, 2021. The names of all the 80 providers is presented in Table 4.2.

4.5 Findings

VPNalyzer uncovers several previously unreported issues from the 80 tested VPN providers. We note that this is only a snapshot of the tested VPN providers. There have already been changes in ownership, and operation of some providers between the time of testing and publication, and the fact that the VPN ecosystem is constantly evolving and changing highlights the need for a tool such as *VPNalyzer* to continue to test and monitor them. We have responsible disclosures in progress regarding our findings.

In this section, we describe our findings in-depth and extract key takeaways. Moreover, considering that the implications of our findings may vary based on the readers' threat model, we provide a condensed visualization in Figure 4.4 and our raw findings in Table C.2 in the Appendix. For instance, if a reader wants to identify how many tested paid providers have traffic leaks during tunnel failure and do not satisfy *some* security and privacy essentials, the second intersection in Figure 4.4 shows the answer is seven paid VPN providers.

4.5.1 Support for IPv6

Majority of VPN providers and servers do not support IPv6. Alarmingly, five VPN providers do not block IPv6 traffic thereby leaking user data to their ISP IPv6 link

Overall, we are the first to show that a majority of VPN providers do not offer support for IPv6—only 11 providers out of 80 tested had IPv6 connectivity. Our data collection consists of multiple experiments for a majority of the VPN providers, and we note that no provider had explicit labelling of servers having IPv6 capability. The state of IPv6 adoption around the world has been increasing steadily, with ≈30% of Alexa Top 1,000 sites being

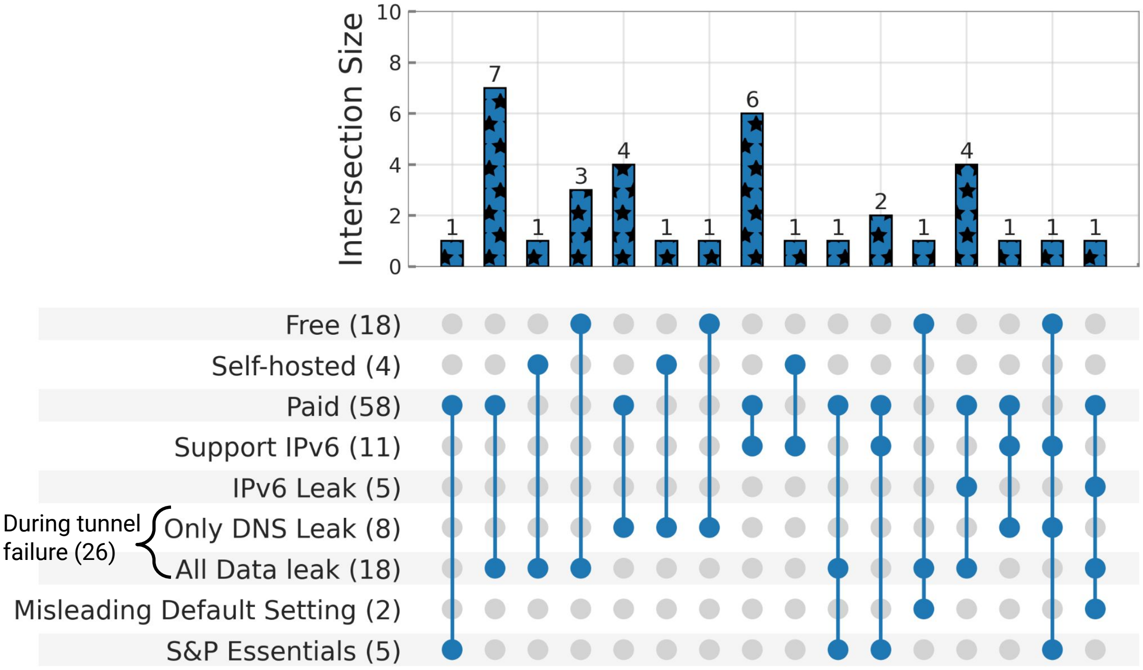


Figure 4.4: **Summary of Results**—The number of providers with intersecting findings (as indicated by the blue circles) are illustrated. S&P Essentials includes a positive result for DNSSEC, Qmin, RPKI validation, and supports DoH.

reachable over IPv6 [248] and over 36% of Google users accessing their services over IPv6 as of July 2021 [67]. However, this trend is not reflected in the VPN providers tested.

Alarmingly, our *AS Mismatch* measurement discovered that five VPN providers leak IPv6 traffic. These VPN providers—Astrill VPN, Norton Secure VPN, Turbo VPN, SurfEasy VPN, and our university VPN—do not block the user's IPv6 connectivity and thus *leak IPv6 data to the ISP*. We filed responsible disclosures for this issue to the providers, and our university VPN has already fixed this IPv6 leak by blocking IPv6 connectivity.

4.5.2 Traffic Leakages

Twenty-six VPN providers leak user data to the ISP during tunnel failure; eight of which leak DNS traffic alone, and the remaining 18 in addition leak other types of traffic

Our *VPN kill switch test* discovers 18 VPN providers leak all user traffic during tunnel failure, suggesting a missing kill switch feature or a faulty implementation. Out of these 18,

All Traffic Leak	Name of VPN Provider
Free Providers (4)	Free VPN by Free VPN.org, Psiphon, Urban VPN desktop, VPN Proxy Master
Self-hosted (1)	OpenVPN Access Server
Paid Providers (8)	Encrypt.me, Hide My Ass!*, IPVanish*, Ivacy VPN, Pure VPN, Speedify, Trust.Zone, Strong VPN*
Paid & Leaks IPv6 (5)	Astrill VPN*, Norton Secure VPN, SurfEasy, Turbo VPN, University VPN
Only leaks DNS traffic during tunnel failure (8)	1.1.1.1+Warp, Avira Phantom VPN, Betternet, Hotspot Shield*, Private Internet Access*, Streisand (on OpenVPN Connect v3), TunnelBear, VPN Owl

Table 4.3: **Providers with traffic leakages**—26 providers leak traffic during tunnel failure. * indicates those with traffic leaks even when their "kill switch" feature is enabled, see §4.5.8.

four are free VPN providers, 13 are paid providers, and one is self-hosted, as reported in Table 4.3. The 13 paid providers also include five that leak IPv6 traffic, described previously in §4.5.1. The kill switch feature is commonly disabled by default in the VPN application often citing that it interferes with the user's browsing experience and lowers usability. In case of tunnel failure, a missing or misconfigured kill switch can lead to privacy and security issues such as leaking sensitive user data. Unfortunately, these tunnel failures are easy to induce, e.g. by simply dropping packets to VPN server, especially by active adversaries such as governments and ISPs.

Our *DNS leak during tunnel failure* measurement finds 26 providers leaking DNS traffic, which includes the above 18, and eight other providers including top ones such as TunnelBear and Private Internet Access. As a result of extensively studying different kill switch implementations, we are the first to measure and discover DNS leaks during tunnel failure which occurs due to VPN providers allowing DNS queries to bypass the tunnel, possibly in an effort to facilitate reconnection. This implementation decision opens up a serious vulnerability which can be avoided in a plethora of ways such as diagnosing issues with and reordering the machine's firewall, and caching VPN-related DNS names to attempt reconnection.

Of the eight providers that leak only DNS traffic, one is a self-hosted VPN–Streisand's OpenVPN configuration on Windows, two are free providers, and five are paid providers. Note that our measurement only reports "definitive" leaks, and is a conservative lower bound. Nevertheless, our findings show that a significant number of popular VPN providers are affected by leakages during tunnel failure, potentially exposing user data to adversaries.

4.5.3 Security and Privacy Essentials

Adoption of practices we consider security and privacy essentials is not uniform across VPN providers

Findings from our *Support for DNSSEC* and *Use of Query Name Minimization* measurements show that many VPN providers do not implement DNS security and privacy measures. We find that only 54 VPN providers use resolvers that have DNSSEC validation enabled, which is important to ensure DNS data integrity. We find that an even fewer number of VPN providers, only 26, use resolvers that support query name minimization(qmin), which was specifically introduced in RFC 7816 to improve DNS privacy.

From our *RPKI validation* measurement, we find that 35 VPN providers have servers in networks where the RPKI validation is enabled. Note that since VPN servers are usually distributed widely in different networks, our result only means that at least one of the tested servers of the provider validates RPKI. Although the proportion of providers performing RPKI validation is higher than Cloudflare's global estimates (20% of Internet) in late 2020 [30], VPN providers should be ahead of the curve especially to protect against issues like BGP hijacking.

We also discover that 14 VPN providers have configured their network to disable DNS-over-HTTPS for Firefox users via their canary domain [145] without the presence of a DNS64 resolver. The Mozilla canary domain is a deliberate signal to convey that the network is unsuitable for DoH, and will result in DoH being disabled for those users who had it enabled by default on Firefox. We believe that doing so without reason, or without informing users is poor practice [146, 144]. To learn if they have any operational reasons, we contacted these 14 VPN providers as part of our responsible disclosure.

On a positive note, from our *Router Scan* measurement, we find that five VPN providers—

Cactus VPN, Encrypt.me, IVPN, Mullvad VPN, and Windscribe—block access to the local ISP router interface while connected to the VPN server. For instance, Mullvad VPN always blocks traffic going to and from the ISP router (and local network) while connected. We recommend that all VPN providers block access to the router interface to protect their users from potentially being vulnerable to a class of deanonymization attacks through the router management web interface and/or the stub DNS resolver.

From our *Port Scan* measurement, we see that VPN servers have port 443 open in 46 VPN providers, port 80 open in 23 VPN providers, ports 53 and 8443 open in 11 providers each, and port 8080 open in 10 providers. While these are not necessarily security risks, they can be used by malicious actors to identify and conduct active probing against VPN servers and can be exploited in a number of ways [78].

In all of the above findings, our results for a particular provider over different experiments and servers in Windows and MacOS are highly concordant.

4.5.4 Sharing of VPN Infrastructure

Many VPN providers use the same underlying infrastructure

Using our *AS Mismatch* measurement, we find that the servers of 27 VPN providers belong to a single AS (AS 9009- M247 Ltd). While previous work found one shared IP block in this AS [103], *VPNalyzer* finds that 14 VPN providers share four IP blocks listed in Table 4.4. The other 13 providers' servers were distributed across the IP space belonging to AS 9009. Additionally, we find an IP block shared by two providers in AS 60068 (Datacamp Limited). Such shared IP blocks are easier for censors or other adversaries to identify and block.

Colocation of VPN servers could be the result of bi-lateral partnerships [103], such as Mozilla VPN using servers "powered by Mullvad" [147]. From our measurement, we find that IP blocks in AS 16509 (Amazon) infrastructure are shared across Norton Secure VPN and SurfEasy VPN, which are both different brands under the NortonLifeLock Product

IP Block	VPN Providers
AS 9009:	
37.120.128.0/17	IPVanish, Touch VPN, Nord VPN, Norton Secure VPN
217.138.192.0/18	Boleh VPN, Ivacy VPN, Hide.me, Mozilla VPN, Goose VPN, Windscribe
5.181.234.0/24	Pure VPN, OVPN
95.174.64.0/22	Fastest VPN, Atlas VPN
AS 60068:	
84.17.48.0/21	CyberGhost VPN, NordVPN

Table 4.4: **VPN Providers with shared infrastructure**—IP blocks shared by different providers in AS 9009 and 60068.

family [159]. Our results are only a lower bound, as we did not connect to every single server available from each provider, since studying this was not our main goal. Furthermore, sharing of infrastructure could also be due to small VPN providers outsourcing or renting resources from a large cloud/hosting provider, or a single parent company owning multiple VPN brands that share infrastructure [234].

4.5.5 Deceptive and/or Malicious Behavior

Malicious and deceptive behaviors such as traffic interception are not widespread but are not non-existent

We find that malicious behavior such as TLS interception is less widespread, to an even lesser extent than previously reported [103, 89]. This could be due to the fact that we test more popular, premium VPN providers that are used by a large population of users. Even so, we do still find evidence of manipulation, previously unreported, in at least two providers—Betternet (on both MacOS and Windows) and Turbo VPN (Windows)—as well as deceptive geolocation claims by four free providers.

We find two instances of abnormal TLS responses and one provider serving non-genuine response for DNS queries. Betternet returns an RFC 6598 Carrier-grade NAT address for all DNS queries. Although this could be a design choice for optimization, we still label it unexpected behavior. In our *TLS interception* measurement, when we request our custom domain with the SNI header modified to `google.com`, we see that both Betternet and

Turbo VPN return the certificate belonging to Google whereas we expect to see our custom domain's certificate. This could be due to an internal policy of handling requests to entities such as Google [57] but we still report it as abnormal behavior. We have reached out to the providers as part of our responsible disclosure.

From our *Geolocation test*, we find instances of "deceptive" geolocation in four out of the 18 free VPNs—namely Free VPN by Free VPN.org, VPN Owl, VPN Hotspot - Unlimited Proxy, and VPN Super. In these providers, the VPN-advertised server location and the geolocation determined by our Cloudflare endpoint do not match. They range from nearby countries (United Kingdom and Germany), to different hemispheres (Japan and Australia). This corroborates findings from Weinberg et al. in [240] and has potential for further future work using the active geolocation measurement data collected using *VPNalyzer*.

4.5.6 Use of Public DNS Services

Twenty-nine VPN providers (including paid, premium ones) configure clients to use public DNS servers

From our *DNS Discovery* measurement, we observe that 29 VPN providers have configured their client applications to use public DNS resolution services, such as Google Public DNS, Cloudflare DNS, OpenDNS, and Quad9, as illustrated in Figure 4.5. Users must be informed that their DNS queries are being routed to third-party organizations such as Google in addition to their VPN provider, especially when users pay for the VPN service [204, 190]. In general, our goal is to report to the user all the entities that handle their DNS queries. Depending on the user's threat model and VPN use case, this information may be critical.

Of these 29 providers, we find two interesting behaviors. We observe that Windscribe uses its own DNS servers, but from our *DNS discovery* we see that when the recursion is done over IPv6, it uses Cloudflare DNS. On the other hand, Trust.zone always uses Google Public DNS on Windows but allows a fallback to using user's default DNS through the VPN on MacOS due to a configuration failure on Tunnelblick.

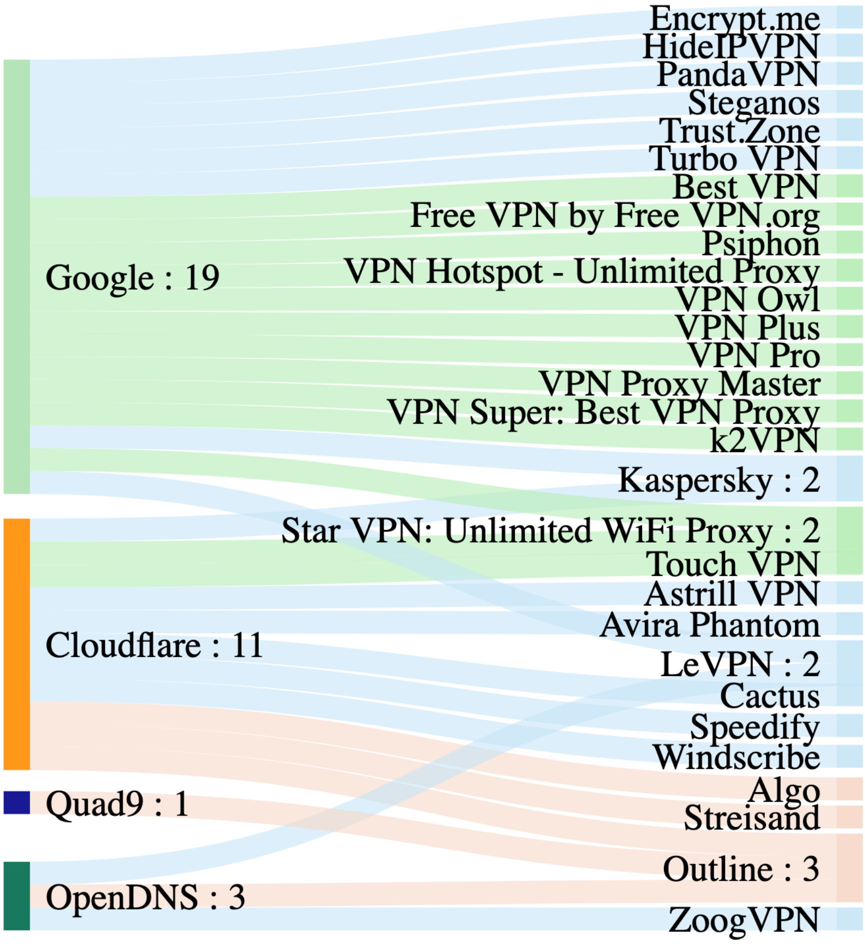


Figure 4.5: **Use of Public DNS Services**—List of 29 VPN providers that use popular public DNS services. Free VPNs are colored in green, paid in blue, and self-hosted in orange.

Among the free VPN providers, 12 out of 18 configure their users to use public DNS servers with Google Public DNS being the most popular choice, followed by Cloudflare DNS. Additionally, we find that three of the self-hosting VPN solutions, Algo, Streisand, and Outline configure the client to use public DNS services. In the case of the self-hosted solutions, having no option to easily override the use of the public DNS services could prove problematic in countries where access to these public DNS services is blocked.

Some VPN providers host their "own" DNS service on hosting providers such as Amazon, Linode, and Digital Ocean, which is different from where their corresponding VPN servers are hosted. While this is different from simply outsourcing user queries to public

DNS services, we still note that users' DNS queries are being served through different hosting providers.

4.5.7 Use of DNS proxies and Mitigating DNS Leaks

Twelve VPN providers have implemented DNS proxies which help mitigate DNS leaks

From our *DNS Proxy* measurement, we find 12 VPN providers employing custom DNS proxies in their network in order to mitigate DNS leaks. On a positive note, of these 12, we also find some VPN providers such as Mullvad VPN, explicitly blocking queries to public DNS resolvers and only allowing queries sent to their own DNS servers. We believe that this is a good security practice, as it is done in an attempt to ensure that the user's DNS queries are not leaked to other third-party DNS resolvers.

4.5.8 Case Study: Testing Custom Secure Configurations

Ten VPN providers out of 39 tested leak traffic even in a more secure setting, six even had a "kill switch" enabled

VPN providers often optimize for performance and usability over security because features like blocking third-party DNS and kill switch can interfere with user experience. To measure if VPN providers offer meaningful, configurable security features in their application, we conduct this case study of testing the 39 most popular VPN providers and find that a multitude of issues persist even in the secure mode.

From our measurements, we find traffic leaks in 10 providers tested in their secure mode, six of which *even had a kill switch setting enabled*, as detailed in Table 4.3. Furthermore, we also observe two providers—Astrill VPN, Norton Secure VPN—which were reported in §4.5.1 to leak IPv6 traffic, also do so in their most secure configuration. This case study highlights egregious implementation failures as these 10 providers suffer from information leakage during tunnel failure even in their most secure configuration, shown in Table C.2. While all 10 providers in their secure configuration, including top providers like TunnelBear, leak DNS traffic during tunnel failure, six of them *also leak all traffic*.

We find that IPv6 support is an “advanced” feature that needs to be turned on by the user in at least two providers—IVPN and Mullvad VPN. Furthermore, we noticed that VPN providers have different names for their mechanism to protect users’ data from leaking during tunnel failure (e.g. VigilantBear, Kill switch, Disable network access, Firewall always on *etc*), making it difficult for users to identify and enable this feature.

We find that four VPN providers—Bullguard VPN, F-secure Freedome, KeepSolid, and ProtonVPN—block access to ISP routers *only in the “secure” mode*. For example, ProtonVPN allows access to the ISP router interface under the default configuration but blocks the access when “Netshield” is enabled.

Alarmingly, we find that two of the 80 VPN providers have misleading default settings. The Astrill VPN and Psiphon applications are configured to tunnel only browser traffic *by default* and hence, for all our findings in the work, Astrill VPN was run using both its OpenVPN and Wireguard protocol (also part of this case study in its secure modes), and Psiphon was run using its L2TP/IPsec protocol. We note that their default configuration is unsafe, as it poses potential security and privacy risks for users who may transmit sensitive information using non-browser apps, under the assumption that the VPN application encrypts all their traffic. Typically, VPN products that tunnel only browser traffic are downloaded and configured as browser extensions. Given that users have to download an app to install and run Astrill VPN and Psiphon, it may mislead them to think that all traffic is tunneled by default.

4.6 Discussion

Our evaluation of 80 popular VPN providers with *VPNalyzer* uncovers several important issues with VPN providers and shows the usefulness of *VPNalyzer* as a tool to investigate the VPN ecosystem at scale. While one experiment on *VPNalyzer* currently takes ≈20 minutes to run and conducts 15 measurements, the modular design supports the efficient addition of new tests and upgrades to existing measurements. *VPNalyzer* can be updated continuously to address the evolving nature of threats in the VPN ecosystem. Although we do not focus on testing all available servers for each VPN provider, our findings are highly consistent across the multiple servers tested for a provider. We highlight the results of most measurements in §4.5, but we intentionally choose not to report the bandwidth measurements collected by *VPNalyzer*. While the individual bandwidth results are interesting to users and are displayed on the app, a large-scale comparison of bandwidth between VPN providers is subject to many compounding factors, such as time of measurement and traffic capacity of ISPs.

Our findings shed light on the lack of standardization and regulation by highlighting the varying levels of security and privacy we see offered by the VPN providers. We discover that the mechanism to protect user's traffic during tunnel failure (i.e. kill switch, firewalls, shields), IPv6 connectivity, blocking of ads and tracking, and smart/secure DNS are all features often disabled in the "default" mode of VPN applications. While we recognize that this is a conscious decision from the VPN provider, these settings should be made more accessible and user-friendly. There is no standard jargon for these features, and the names and capabilities of simple settings are often exaggerated by VPN providers. For instance, terms like "military grade encryption", and "smart connect" are often proffered with little explanation accompanying them.

There have been attempts by certain VPN providers to form coalitions like the VPN Trust Initiative (VTI) [83] that take steps towards regulation and setting industry best practices. However due to vested interests of a handful of VPN providers, these efforts are typically not adopted by other popular providers in the VPN ecosystem. Moreover, anecdotal evidence suggests that VPNs in these coalitions do not follow all its basic principles. For instance, Ivacy VPN which is part of VTI, advertises "anonymity" prominently on its website [94] whereas "Never claim VPNs guarantee anonymity" is one of the basic principles of the coalition [83].

There is a need for independent, unbiased parties to put forward standards based on systematic, data-driven studies such as the one provided in this work. To that end, we partnered with *Consumer Reports* and served as panelists on a workshop about VPNs organized by them which was attended by over 1,500 users. Thereafter, *Consumer Reports* used our *VPNalyzer* tool as the first in a line of systematic investigation to help evaluate a set of popular providers for a recommendation article on their website. The simplicity and usability of our *VPNalyzer* tool helped *Consumer Reports* to also test providers to evaluate and recommend.

Responsible disclosure of issues found by *VPNalyzer* are already helping VPN providers improve and fix issues with their service. For instance, our university VPN has already fixed their IPv6 leak, and made their kill switch implementation more secure. For our next steps, we plan to release the *VPNalyzer* tool for a wider audience in the coming months and we hope that *VPNalyzer* benefits users and helps the general public choose better VPN providers.

4.7 Broader Impact

Our testing methodology and the VPNalyzer tool presented in this work was used by Consumer Reports as the first line of systematic investigation to evaluate a set of popular VPNs.

This work was featured in a white paper [72], and our insights were quoted in two articles written by Consumer Reports [70, 71]. Based on our results in this work, we filed several responsible disclosures with VPN providers, and five of whom have had discussions with us and have started fixing the issues. One provider awarded us a bug bounty as well. Our work with Consumer Reports was cited by members of Congress urging the FTC to call for regulation in the VPN ecosystem [55].

4.8 Conclusion

To our knowledge, we are the first research study to build a system, *VPNalyzer*, and develop a tool with automated tests and functionality to empower researchers and users to assess the service provided to them by VPN providers. We demonstrate that the *VPNalyzer* application is able to find important issues by analyzing 230 experiments from 80 popular and diverse desktop VPN providers. We find several notable issues and vulnerabilities, including IPv6 leaks, kill switch leaks, DNS leaks during tunnel failure, and the lack of adoption of certain security and privacy essentials. We plan to conduct a public release of the *VPNalyzer* tool in the coming months, which will only further increase our measurement scope and findings. We hope that *VPNalyzer* benefits researchers and users alike, helps the general public make more-informed decisions about which providers to use for their particular needs, and ultimately fosters stronger security and privacy practices in the VPN ecosystem.

CHAPTER V

Leveraging Cross-Layer Network Latency Measurements to Detect Proxy-Enabled Abuse[1]

Online advertising revenue in the U.S. reached over $209.7 billion in 2022, almost three times as much as the revenue from U.S. TV advertisements [215, 180]. Companies and service providers increasingly thrive on collecting, processing, and sharing large amounts of data on users, and monetize this data by enabling targeted ads and tracking. However, with the growing awareness and demand for privacy among the general public, emerging tech companies are aiming to democratize the ad delivery ecosystem by building services that are privacy-focused and even share ad revenue with users. Creative solutions that promote user agency by allowing users to opt-in to viewing and earning rewards from unobtrusive, privacy-preserving ads are gaining popularity [3]. On a technical level, these "ad reward networks" make use of browsing activity to generate ads but all the computational processes occur locally on the user's device, without any data being exfiltrated to the companies. They then use privacy-preserving protocols to confirm ad event activity and reward users in cryptocurrency based on ad providers and region-specific pricing.

One critical threat to these ad reward networks is attackers actively game the reward system by leveraging VPNs or proxies. These attackers fabricate their geolocation and get access to more high-reward ads, and falsify ad event activity to earn disproportionate

[1] This chapter is based on: Reethika Ramesh, Philipp Winter, Sam Korman, and Roya Ensafi. CalcuLatency: Leveraging Cross-Layer Network Latency Measurements to Detect Proxy-Enabled Abuse.

rewards. Ensuring the adoption and long-term viability of these privacy-focused, user-first business models critically hinges on the availability of cost-effective and easily deployable techniques to differentiate regular users from those that use VPNs or proxies, for further monitoring for suspicious activity. This prevents attackers from taking advantage of the system while allowing benign users to continue using VPNs if they do so for privacy reasons.

Balancing abuse-prevention techniques with rigorous privacy-requirements is a hard challenge, especially if the service provider seeks to uphold their privacy-focused business model. The state of the art in VPN and proxy detection uses a series of heuristics developed by surveilling client traffic at large. Especially deployed by large platforms, these services use the data collected to build IP reputation metrics, generate user-specific metrics, and even use black box services that claim to detect "suspicious" users and IP addresses. But to get access to these metrics, the service provider will need to compromise on protecting the privacy of their users. In short, we seek to answer: *Can such service providers build a system using minimum connection features, such as latency, to infer VPN or proxy use, without jeopardizing user privacy or the need for data collection?*

In this work, we build and evaluate our solution to this question, CalcuLatency, that leverages cross-layer network latency measurement techniques to differentiate users who are using a remote, long-distance VPN or proxy from those who are not. We leverage the fact that when a user connects through a proxy, the application-layer latency will be measured from end-to-end (service provider to browser), whereas, any measurement on the network-layer only reaches the proxy (service provider to VPN server), and the round-trip time (RTT) difference between these two metrics can be a reliable indicator. To this end, we combine existing techniques such as WebSocket RTT, TCP handshake RTT, and ICMP ping, as well as implementing and evaluating a modified traceroute method (0trace), which has not been done before. 0trace conducts hop enumeration from within an *existing, established TCP connection* such as a WebSocket session. Using CalcuLatency, we can even differentiate between network-layer and application-layer proxies.

To evaluate CalcuLatency, we not only have to evaluate each of the measurement techniques on their own but also evaluate the system as a whole by using a comprehensive (geo)diverse set of clients. The former is necessary to characterize and quantify, for each technique, the effects of network jitter and the reliability of the methods. For the latter, we implement the system as a web service and conduct a two-pronged evaluation of CalcuLatency as a whole: (i) We perform an in-depth testbed evaluation where we maximize the number of VPN products, servers, protocols, and browsers tested, from four different user geolocations; (ii) we expand this to include a real-world, crowdsourced evaluation to collect and analyze more diverse user geolocations. To this end, we rally user participation on Twitter and personal contacts, using the authors' accounts. We have participants from 37 different countries located in all (six) continents and 144 autonomous systems, and collect a large crowdsourced dataset. Due to the value such data provides, we will open-source our data to aid future research.

We find empirically that a viable threshold to consider a particular client as a remote VPN or proxy connection is 50 milliseconds. In 98% of *all direct measurements* from both sets of evaluation, we find that the RTT difference is below this threshold of 50ms. Conversely, 89.1% of *all VPN measurements* in the testbed evaluation and 63.9% of the crowdsourced evaluation have an RTT difference of *above 50ms*. Through location analysis, we were able to attribute a majority of the remaining 10.9% and 36.1% VPN measurements to the fact that the VPN server and the user were located very close to each other.

Investigating the VPN measurements whose RTT difference was below 50ms, we find that in a majority of such cases (66.2%) the VPN server is located close to the user (within 650mi). This can be equated to the straight line distance between Washington DC and Boston, MA (635 mi) or the straight line distance between Mountain View, CA, and San Diego, CA (685 mi). Surprisingly, this indicates that we can reliably detect proxy use when a user and the VPN or proxy are just 650 miles apart or more.

Overall, we consider 50ms to be the RTT difference threshold for a connection to be labeled as a remote VPN or proxy, our system has a *false negative rate of 2.9%* (28/964) and a low *false positive rate of 0.95%* (2/210).

Adopting a more conservative analysis strategy, if we only consider measurements that have a successful ICMP ping and 0trace reaches the client or at least the same network as the client (i.e. 96% of all measurements), we find that the RTT difference is below 50ms for all of the 210 direct measurements, and none were wrongly flagged using our system. These measurements include 137 unique user IPs from 84 different Autonomous Systems (ASes), from six different continents and over 34 different countries. We also reduce our false negatives to 2.77% (26/937).

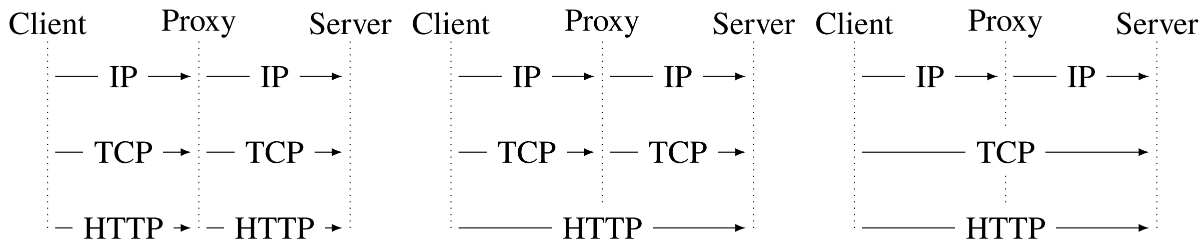
While CalcuLatency cannot detect all proxy use especially if the user and VPN are close to each other, CalcuLatency is an easy-to-deploy, open-source solution that can serve as an inexpensive defense against proxy-enabled abuse for server-side operators. We incorporate existing methods to calculate network latency and implement a modified traceroute method to overcome the challenges of our probes getting blocked due to stateful firewalls or NATs. Though this traceroute concept was first discussed over two decades ago, we are the first to implement and deploy in a real-world system and evaluate its performance. Our system CalcuLatency is soon going to be deployed into production with Brave Software's suite of anti-fraud techniques.

5.1 Background

Below, we discuss how network proxies differ in their architecture and we explain each of our measurement methods.

5.1.1 Architecture of Network Proxies

Figure 5.1 illustrates how a proxy's type affects the underlying protocol stack. We divide the protocol stack into three layers as per the OSI model: the application layer (HTTP), the transport layer (TCP), and the network layer (IP). Application-layer proxies terminate



(a) Application-layer proxies, e.g., proxies that work via HTTP `CONNECT`.

(b) Transport- and Sessions-layer proxies, e.g., SOCKS, Tor, and SSH.

(c) Network-layer proxies, e.g., VPNs like OpenVPN and WireGuard.

Figure 5.1: **Architecture of Network Proxies**—We distinguish between three types of proxies that differ based on the layer at which they terminate client connections. Clients connecting to application-layer (5.1a) and transport-layer proxies (5.1b) first establish a TCP connection with the proxy and initiate the proxying using protocol-specific messages. The proxies create new TCP connections to the web server, and then establish the tunnel. Whereas, network layer (5.1c) proxies authenticate the client and establish a tunnel, after which all packets including TCP SYN packets are encapsulated and proxied through the server.

the application protocol, e.g., to inspect content for malware (Figure 5.1a). This includes Web servers that support the HTTP `CONNECT` method. Transport-layer proxies like Tor and SSH (both build on top of the SOCKS protocol) pass through the application protocol but terminate the client's TCP connection (cf. Figure 5.1b). Clients use SOCKS's signaling mechanism to tell the proxy what destination to connect to. Finally, network-layer proxies like OpenVPN or WireGuard perform NAT but pass through the client's TCP connection (cf. Figure 5.1c).

5.1.2 The 0trace Technique

In a traceroute, a client sends multiple packets to a server, with each packet containing an incrementing time-to-live value (TTL) in the IP header. These packets are stateless "stray" packets, meaning that they do not belong to an open network connection. Stateful firewalls may reject such packets, terminating the traceroute.

In 2007, Zalewski invented a traceroute technique that can get past stateful firewalls [256]. This technique—called 0trace—creates trace packets that match the five-tuple of an already-established TCP connection. Unable to tell apart 0trace packets from packets belonging to this TCP connection, stateful firewalls will let 0trace's trace packets pass. 0trace achieves its goal by manipulating network packet headers and does so by crafting its own network packets using Linux's raw socket API. However, this technique comes at the cost of potentially corrupting the TCP connection: the receiving host may terminate the TCP connection upon receiving an unexpected trace packet. Care must therefore be taken when selecting a TCP connection as 0trace may disrupt it.

While 0trace's purpose is hop enumeration in the presence of firewalls, we use it for RTT measurements. The go-to technique for RTT measurements is ICMP echo requests (colloquially called "pings") but we found that many residential ISPs block pings (§ 5.2.4.1). While 0trace does not always work in such settings, it does allow for more accurate RTTs in the presence of firewalling as we show in Section 5.3.3.

5.1.3 The WebSocket API

The WebSocket protocol is an application layer protocol on top of TCP, which offers Web applications a bidirectional socket for communication. While distinct from HTTP, the WebSocket protocol is compatible with HTTP and uses the HTTP `Upgrade` header in its handshake. This allows Web servers to handle both HTTP and WebSocket connections on the same port. Once a WebSocket connection is established, the client and server exchange binary data. Web applications can use WebSockets by taking advantage of the JavaScript WebSocket API that's supported by all modern browsers [2]. Later in this work, we use WebSockets to determine the application-layer round trip time between a client and server.

5.2 Method

Our aim is to measure round-trip times on the application, transport, and network-layers, and use a combination of these different measurements to reason about whether a particular connection is coming through a proxy server. We combine four cross-layer

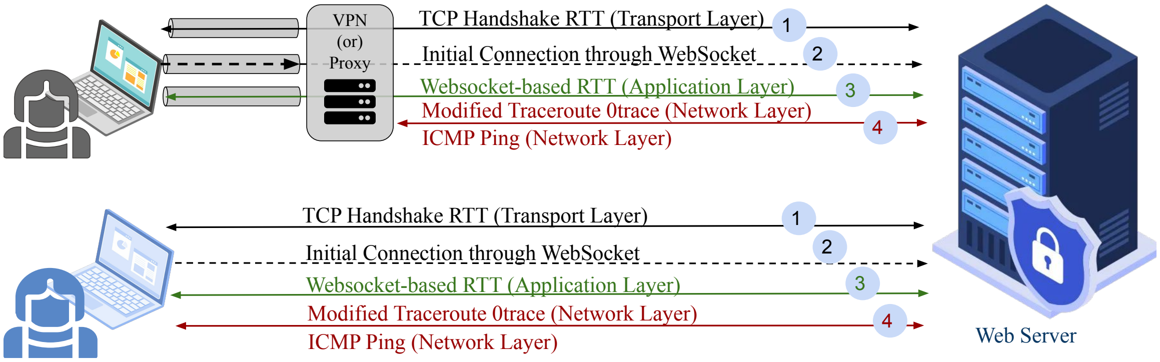

VPN
(or)
Proxy
TCP Handshake RTT (Transport Layer)
1
Initial Connection through WebSocket
2
Websocket-based RTT (Application Layer)
3
Modified Traceroute 0trace (Network Layer)
4
ICMP Ping (Network Layer)
TCP Handshake RTT (Transport Layer)
1
Initial Connection through WebSocket
2
Websocket-based RTT (Application Layer)
3
Modified Traceroute 0trace (Network Layer)
4
ICMP Ping (Network Layer)
Web Server

Figure 5.2: **Measurement Setup**—User connects to the Web server either via a VPN or Proxy (top) or directly (bottom). We illustrate the measurements done in both cases. The network layer measurements only reach the proxy server in case the user is using a proxy, but reaches all the way to the user's public IP if they are connecting directly.

latency measurement techniques into a single software service that we call CalcuLatency. Our system integrates well into existing service provider infrastructure, and we create and publish a pipeline to analyze the collected data.

5.2.1 System Architecture and Assumptions

A typical client-proxy-service scenario consists of three entities: (i) a service provider that makes available one or more HTTP endpoints to its clients; (ii) clients that use the service provider's services; and (iii) proxy servers that some clients use to disguise their topological (i.e., IP address) and physical (i.e., their home country) location. While most clients are honest, there are some malicious ones that seek to defraud the service provider while using network proxies to disguise their location. For instance, the ad reward network provides different rewards for different geolocations, and users may seek to earn disproportionate rewards using proxies. The service provider needs an inexpensive and practical solution that can tell apart clients that use remote proxies to mask their geolocation from those that do not. We introduce CalcuLatency to fill this gap.

Broadly, our technique allows for the detection of the three proxy types illustrated in Figure 5.1. CalcuLatency's key insight is that the use of a proxy affects the layers in the OSI model differently. If we find a non-trivial difference between the application-layer

RTT (Δ_{AL}) and the transport-layer RTT (Δ_{TL}) and/or the network-layer RTT (Δ_{NL}), we can conclude that the client is using a proxy server.

Consider a concrete example: A client in India is using a VPN server in Canada to make an HTTP connection to our CalcuLatency server in the U.S. Upon accepting the connection, our server now determines three types of round-trip times to the client, as shown in Figure 5.2. First, it upgrades the HTTP connection to WebSocket and sends several pings (using JavaScript) to determine the application-layer RTT. Next, our server inspects the (previously recorded) TCP handshake to infer the transport-layer RTT. Finally, our server determines the network-layer RTT by sending ICMP echo requests and by running a 0trace measurement, which piggybacks onto the already-established HTTP connection.

Having determined all RTTs, our server notes that the WebSocket RTT is $250\,\mathrm{ms}$—the time it takes for the WebSocket pings to travel from the U.S. to Canada, to India, and back again. The ICMP echo responses however exhibit an RTT of only $35\,\mathrm{ms}$—the time it takes to go from the U.S. to Canada, and back again. This difference stems from the fact that the WebSocket ping was answered by *the client's browser in India* while the network-layer ping was answered by *the proxy's network stack in Canada*. Confronted with the RTT difference of $250 - 35 = 215\,\mathrm{ms}$, the service provider concludes that the client is using a proxy.

Assumptions: First, CalcuLatency requires an HTTP connection between the client and the server. We chose HTTP because of its ubiquity but other application-layer protocols work equally well. Second, the client's proxy must not be geographically close to the user. This assumption is reasonable in the case of service providers, such as ad reward networks, that seek to identify users who spoof their country of origin. Third, we assume that clients do not control the network behavior of the proxy and therefore cannot have it delay selected network packets. Instead, we assume that clients either use open proxies or rent them, e.g., as part of a VPN subscription. Even if the client does control the proxy, it is not guaranteed to evade CalcuLatency's detection as our 0trace component can estimate the RTT to the

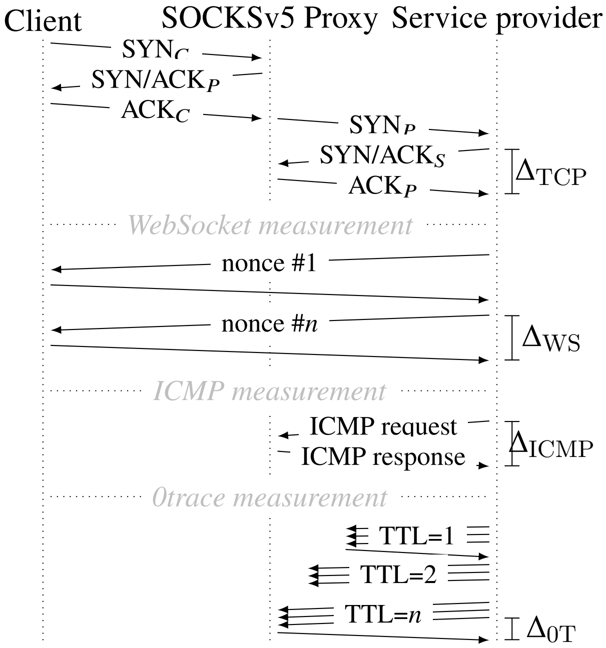


Figure 5.3: **Measurement Flow**—In this example, the client uses a SOCKSv5 proxy. The service provider determines four round-trip times (on three layers) to infer what kind of proxy the client uses.

client using adjacent hops on the path that are outside the client's control.

While not universally applicable, we believe that these assumptions are both realistic and commonplace. What's more, CalcuLatency does not suffer from the shortcomings of existing, established proxy detection techniques like IP reputation blocklists that just use IP-to-Geolocation databases [241, 63, 174].

5.2.2 Determining the Application-layer RTT

How can the service provider determine the application-layer RTT to the client? We chose WebSocket for this task: As of June 2023, WebSocket is supported by all major browsers [1], it is compatible with HTTP, and it is purpose-built for real-time communication, making it a natural choice for determining round-trip times. As mentioned above, CalcuLatency is not dependent on WebSocket and could employ alternatives, like WebRTC, XMPP, or HTTP page load times [238, § II.B].

Specifically, a client establishes a WebSocket connection with the service provider.

The server then sends n WebSocket-based pings to the client as illustrated in Figure 5.3. Section 5.3.1 suggests that $n \geq 10$ is useful to make the measurement more robust against transient networking issues or asymmetric routing, both of which interfere with our measurement. In our CalcuLatency system, we send 100 WebSocket-based pings to the client to minimize the odds of interference. Upon receiving a WebSocket ping, the client's browser simply echoes back each ping to the server. Upon receiving the echo response, the server determines the WebSocket RTT on the Application Layer (Δ_{AL}) between itself and the client as follows:

$$\Delta_{\mathrm{AL}} = \min\{\Delta_{\mathrm{AL}_1}, ..., \Delta_{\mathrm{AL}_n}\}.$$

Importantly, our technique must work in an adversarial environment because malicious clients seek to disguise their use of a proxy. These clients control the server-provided JavaScript, and can therefore choose to either delay the echo or send an "anticipatory" echo before having received the corresponding request. The former is against a malicious client's interest because it would increase the odds of CalcuLatency concluding that the client is using a proxy. Malicious clients are therefore incentivized to respond as quickly as possible. To thwart the "anticipatory response" attack, we include a unique nonce in every echo request. The client has to embed the nonce in its echo response, preventing it from sending responses before having received the request.

5.2.3 Determining the Transport-layer RTT

We take advantage of the fact that in our setting, the client establishes a TCP connection with our server, meaning that we are in control of the TCP three-way handshake. We estimate the RTT between client and server by calculating the time difference between the server responding with a SYN/ACK segment and the client acknowledging receipt with an ACK segment, as discussed by Ding and Rabinovich [43]. We refer to this RTT as Δ_{TL}. Again, clients have no incentive to delay their ACK segment and are unable to send an "anticipatory" segment because of the difficulty of predicting TCP sequence numbers. We find that Δ_{TL} is reliable and straightforward to determine but we have *only a single sample* per connection, which makes this technique susceptible to transient conditions like network congestion. Indeed, Høiland-Jørgensen et al. [82] showed that "20% of the clients experience increased delays of more than about $80\,\mathrm{ms}$, at least 5% of the time."

We considered taking advantage of the TCP timestamp option to measure RTT. We chose not to because TCP timestamps are not universally used [120, § 6] and often exhibit poor granularity—Veal et al. showed in their PAM'05 paper that the majority of 500 popular web servers used a granularity of either $10\,\mathrm{ms}$ or $100\,\mathrm{ms}$ [231, § 3].

5.2.4 Determining the Network-layer RTT

Finally, we focus on the lowest layer for which we determine the round-trip time: the network layer. Determining the RTT to an IP address is challenging because it is difficult to reliably get a remote network stack to respond to unsolicited IP packets. Prior work reported that it is increasingly rare for hosts to respond to ICMP echo requests, or send a TCP RST segment in response to unsolicited SYN segments [14, § 4]. To maximize success, we draw on two separate techniques to estimate Δ_{NL}, our network-layer RTT: ICMP and 0trace.

5.2.4.1 ICMP RTT

CalcuLatency sends five ICMP echo requests to the client. We refer to this measurement as Δ_{ICMP}. Prior work had found that some proxies don't answer to ICMP: 90% of VPN servers tested by Weinberg et al. [241] reportedly ignored ICMP requests. However, they only tested servers belonging to seven undisclosed VPN providers. Due to ICMP's potential usefulness to CalcuLatency, we revisit this topic by asking the following research question:

Do VPN servers respond to ICMP pings? To answer this question, we sent ICMP requests to IP addresses belonging to 80 VPN providers that past work studied [251, 185]. These VPN providers host servers that run various VPN protocols including OpenVPN, WireGuard, and other proprietary protocols. We augment our list of IP addresses with addresses present in the same /24 network as the VPN IP addresses, because IPs adjacent to known VPN servers on hosting provider(s) are likely to be rented by VPNs as well, according to previous work [185, 103]. Thus, our final list contained 492 unique IP addresses. We chose to ping addresses adjacent to known VPN servers because a response from an adjacent address is likely to have a near-identical RTT to that of the actual proxy, which may not respond to ICMP requests. Our ICMP requests to these addresses resulted in 369 addresses (75%) that responded—a higher percentage than what past work found.

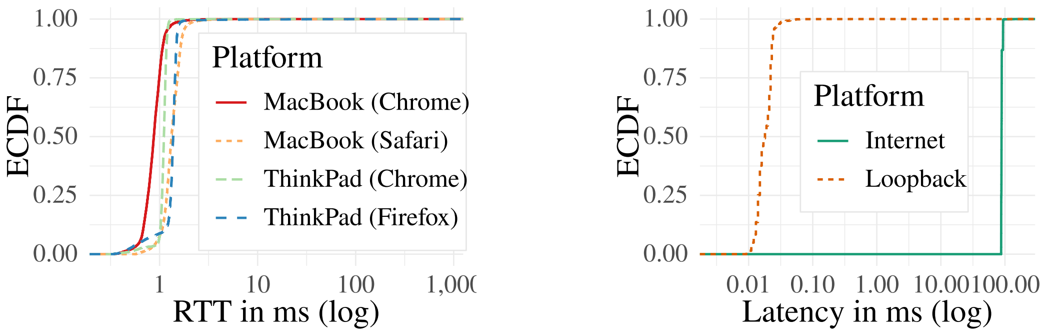
VPN servers appear likely to respond to ICMP requests but what about a random sample of the IPv4 population? To answer this question, we ran a ZMap scan [255] targeting a randomly selected set of 1 million IP addresses. This resulted in 10.52% of IP addresses that responded to our ICMP requests. This is in line with prior work by Bano et al., which found that approximately 10% of IP addresses respond to ICMP pings [14, § 4.1]—making the odds of a response low.

We conclude that VPN servers are significantly more likely to respond to ICMP requests than randomly selected IP addresses, which includes residential clients that don't use a proxy. We therefore decide to augment our ICMP measurement with 0trace, which we discuss in the next section.

5.2.4.2 0trace RTT

We use Zalewski's 0trace technique [256] for the purpose of determining the round-trip time between the server (where the 0trace measurement is initiated) and a connecting client. Again, we take advantage of the fact that our setting has the client establish a WebSocket connection to the server, meaning that we have an already-established TCP connection that 0trace can use for its TTL and RTT measurement.

We developed a Go package that implements 0trace. This package uses Linux's raw socket API to send manually crafted TCP segments with incrementing IP time-to-live (TTL)



(a) RTT distribution of WebSocket measurements on four platforms.

(b) Latency distribution of TCP handshake measurements over two networks.

Figure 5.4: **Building block evaluation of different techniques used in CalcuLatency**

values. We carefully craft 0trace packets to match the TCP five-tuple of the WebSocket connection,[2] so that firewalls between client and server will not interfere. Our 0trace implementation keeps incrementing the TTL until either (i) the responding IP address is identical to the client's IP address; or (ii) until TTL 32 is reached. For each TTL, we send three redundant probes to account for potential packet loss. If we don't receive a response to any of our three probe packets within five seconds, we mark the given TTL as unresponsive. The RTT of our 0trace measurement ($\Delta_{0\mathrm{T}}$) is the round-trip time of the probe packet *that made it the farthest* to the client, i.e., the probe packet with the largest TTL.

Being equipped with four techniques that measure the RTT across three OSI layers, we now devise a decision tree that helps us collapse all our measurements into a single verdict: does the client use a proxy?

5.2.5 Implementation and Deployment

CalcuLatency combines our application, transport, and network-layer measurements into a single service. We implemented CalcuLatency in both Go and JavaScript (for the WebSocket pings). The Go service consists of (i) a Web server with an endpoint that speaks WebSocket, (ii) a Go package that records TCP handshakes to extract their round trip times,

[2] The five-tuple consists of IP source and destination addresses; TCP source and destination ports; and the IP protocol.

and (iii) a package that runs a 0trace measurement to the client.

While developing our 0trace Go package, we take advantage of the ICMP error requirements stated in RFC 1812 and RFC 792, which state that the returned ICMP error message will contain (parts of) the original datagram's data [85, 84]. We find that the ICMP error message reliably contains the IP header of the original datagram in the ICMP datagram. Hence, we design our server to keep track of the IP ID for each TTL-limited probe it sends out, and it uses the IP ID obtained from the IP header of the original datagram from the received ICMP error packet to correspond the hop IP to the particular TTL value. Using the packet sent time and packet received time, the server calculates the RTT for each hop that responds with the ICMP error. It repeats the process until (if) it reaches the VPN IP, or until the maximum TTL value. All our code is available online under a free software license but we omit the URL to preserve our anonymity.

Currently, we are collaborating with a company to deploy our work in production. We plan to deploy CalcuLatency as part of a CAPTCHA: Some of this company's users are occasionally served a CAPTCHA to verify if the user resides in the country they claim to reside in. The CAPTCHA is non-interactive and consists of HTML and JavaScript, which initiates a WebSocket connection to the CalcuLatency server. In the first phase, CalcuLatency sends 100 WebSocket pings to the client to estimate the application-layer RTT. After CalcuLatency determines the application-layer latency, we calculate the RTT of the TCP handshake, send ICMP pings, and use 0trace to determine the network-layer RTT, as discussed above. Clients that CalcuLatency deems to be using a proxy are flagged for manual inspection.

5.2.6 Confirming the Use of Proxies

The result as determined by CalcuLatency is meant to be used as one "signal" and augmented with other signals from different peripherals of the network in order to make a determination on fraudulent connections. The service provider can use probes designed by Xue et al. [251] and Maghsoudlou et al. [119] to conduct active measurements towards suspected VPN and proxy servers to confirm that these servers run the respective tunneling protocols.

5.3 Building Block Evaluation

Before evaluating CalcuLatency in its entirety, we study its building blocks in isolation.

5.3.1 Reliability of WebSocket Pings

Unlike our network-layer RTT methods, which are typically handled by the kernel's network stack, our application-layer method—WebSocket—traverses not only the kernel's network stack but also the client's browser, and is therefore subject to more jitter. To characterize and approximate this jitter, we built an HTTPS-based Web service that sends 10,000 sequential WebSocket echo requests to the client and measured the round-trip time $\Delta_{\mathrm{WS}_1}, \ldots, \Delta_{\mathrm{WS}_{10{,}000}}$ for each request. We made these measurements over the loopback interface to eliminate networking delays and isolate processing delays.

Figure 5.4a illustrates the results for two consumer laptops (ThinkPad X1, MacBook Pro) running two browsers.[3] The median RTT for all distributions is less than $1.4\,\mathrm{ms}$, and 99% of RTT measurements completed in less than $2.4\,\mathrm{ms}$. All distributions exhibit a long tail, presumably caused by transient load spikes. *CalcuLatency must account for this by running multiple measurements, spaced out over time.*

5.3.2 Reliability of TCP Handshake RTT

We conduct an experiment to understand the reliability of the TCP handshake in measuring RTT. We used Go to build an HTTP server that served a static index page. In parallel to serving this index page, the server uses libpcap to record the TCP handshake;

[3] The ThinkPad is a X1 Carbon 8th generation equipped with a i5-10210U CPU, and ran Chrome 111 and Firefox 111. The 2023 MacBook Pro is equipped with an M2 Pro CPU, and ran Safari 16.3 and Chrome 111.

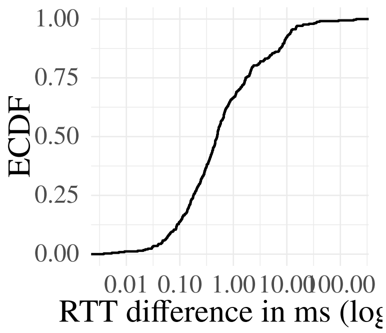


Figure 5.5: Differences between 0trace-determined RTT and RIPE Atlas-determined RTT.

in particular the time delta between the server responding to the client's SYN with its SYN/ACK segment and the subsequent receipt of the client's ACK. We built a shell script that establishes 86,400 TCP connections: one connection per second for 24 hours. We ran this experiment in two network settings: in the "Loopback" setting, client and server run on the same machine, communicating over the loopback interface. By excluding the Internet, this setting highlights computational latency. In the "Internet" setting, client and server run on separate machines, communicating over the Internet. The client ran on a VPS in Ohio, U.S. while the server ran on a VPS in Paris, France.

Figure 5.4b illustrates the results of this experiment. In the "Internet" setting, we see a minimum, median, and maximum latency of 87, 88, and 171 ms, respectively. 99% of handshakes exhibit a latency of less than 94 ms—only 7 ms more than the minimum. The "Loopback" setting, we observe a median latency of 18 µs. 99% of handshakes exhibit a latency of less than 34 µs. As expected, we observe a long tail in both settings and account for outliers by running repeated measurements.

The TCP RTT is readily available and straightforward to calculate, but we only get a single measurement per TCP connection. One could work around this limitation by having the client establish multiple TCP connections to the server.

5.3.3 Reliability of 0trace Pings

To evaluate 0trace, we take advantage of the RIPE Atlas network. First, we set up an HTTPS server that runs our 0trace code whenever a client connects to the server. Next, we select 726 RIPE Atlas probes that had the "home" tag, indicating these probes are likely connected to residential ISPs, which makes for a more realistic evaluation. We stratified our probe selection to cover all continents: Africa (23), South America (43), Australia (84), Asia (125), Europe (201), and North America (250). We then use these probes to run two measurements targeting our HTTPS server:

1. An "SSL" measurement that instructs the probe to fetch our HTTPS server's certificate. The purpose of this is to trigger the service's 0trace measurement toward the client.
2. A "Ping" measurement that sends ICMP echo requests to our server. The purpose of this measurement is to obtain RTT ground truth.

For each RIPE Atlas probe, we measure two RTT values: one for the 0trace measurement and one for the ping measurement. We then determine the absolute difference between the 0trace RTT and the ping RTT, which serves as ground truth.

Our results show that in 470 measurements (64.73%), 0trace was able to send trace packets all the way to the RIPE Atlas probe's IP address. But, we find that only about 47.5% of all probes successfully ran the ping experiments, possibly due to RIPE Atlas toolkit issues. Hence, our analysis only considers the 345 measurement runs where 0trace manages to reach the final hop *and* where the probe's ICMP request succeeded. Among these 345 measurements, we find that in 256 experiments (74.2%), 0trace was able to reach the probe directly, and in 98.4% of these cases, the difference in RTT between the two measurements (0trace and ICMP ping) is below $5\,\mathrm{ms}$, and 100% of the cases had an RTT difference of less than $20\,\mathrm{ms}$, with the range of differences being $0.003\,\mathrm{ms}$–$8.36\,\mathrm{ms}$. The RTT difference for all 345 measurements is presented in Figure 5.5.

In 56 more experiments (16.2%), 0trace reached the same network (Autonomous System) as the probe itself. In this case, we find that the RTT differences is less than $20\,\mathrm{ms}$ in 94.6% of the cases, with the range of RTT differences being $0.14\,\mathrm{ms}$–$92\,\mathrm{ms}$.

Of the remaining 33 (9.6%) where 0trace was not able to reach the probe or its network, we find that the RTT difference between the last hop of 0trace and ICMP ping RTT in 90.9% of the cases is below $40\,\mathrm{ms}$, and the range of RTT differences are $0.10\,\mathrm{ms}$–$196\,\mathrm{ms}$, with a mean of $20.53\,\mathrm{ms}$.

In summary, of the total 345 experiments, **74.2% of the time (256/346) 0trace reaches the probe IP directly**. Considering the RTT difference to be between 0trace's last successful hop RTT and the probe IP's ICMP RTT, we find that 92.2% of experiments had less than $10\,\mathrm{ms}$ of RTT difference, and over **97.4% of experiments had an RTT difference less than** $20\,\mathrm{ms}$.

5.4 Evaluating CalcuLatency in Practice

We conduct a thorough evaluation of CalcuLatency using a two-pronged approach. First, we conduct an evaluation with a controlled testbed where we control the webserver, the client, and test our system with a variety of different VPN products and proxy servers, that run multiple different protocols. We test the system with mobile connections, various different user locations, without any proxies connected, and then by connecting to various VPN and tunnelling protocols. Next, we extend on this experiment by conducting a large evaluation with volunteer Internet users. This mimics a real-world deployment test for CalcuLatency because we draw on a geographically diverse population that's representative of the average Internet user. For this experiment, we set up our measurement server at a university network and advertised our crowd-source measurement page via social media channels.

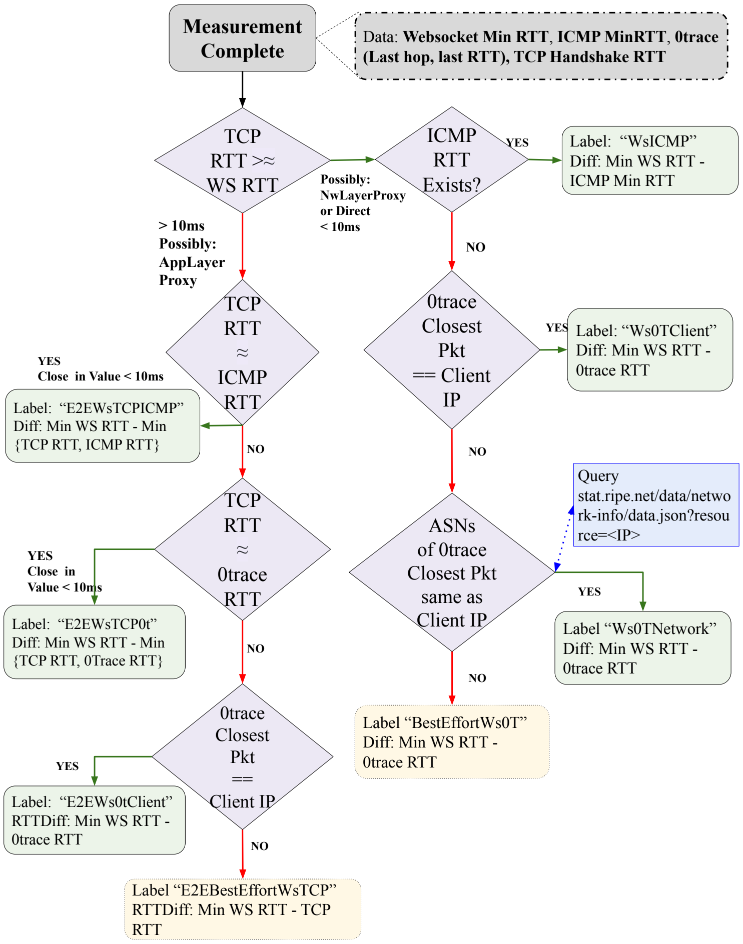

Measurement Complete
Data: Websocket Min RTT, ICMP MinRTT, 0trace (Last hop, last RTT), TCP Handshake RTT
TCP RTT >≈ WS RTT
Possibly: NwLayerProxy or Direct < 10ms
ICMP RTT Exists?
YES
Label: "WsICMP" Diff: Min WS RTT - ICMP Min RTT
NO
> 10ms Possibly: AppLayer Proxy
TCP RTT ≈ ICMP RTT
YES Close in Value < 10ms
Label: "E2EWsTCPICMP" Diff: Min WS RTT - Min {TCP RTT, ICMP RTT}
NO
0trace Closest Pkt == Client IP
YES
Label: "Ws0TClient" Diff: Min WS RTT - 0trace RTT
NO
Query stat.ripe.net/data/network-info/data.json?resource=<IP>
TCP RTT ≈ 0trace RTT
YES Close in Value < 10ms
Label: "E2EWsTCP0t" Diff: Min WS RTT - Min {TCP RTT, 0Trace RTT}
NO
ASNs of 0trace Closest Pkt same as Client IP
YES
Label "Ws0TNetwork" Diff: Min WS RTT - 0trace RTT
NO
Label "BestEffortWs0T" Diff: Min WS RTT - 0trace RTT
0trace Closest Pkt == Client IP
YES
Label: "E2EWs0tClient" RTTDiff: Min WS RTT - 0trace RTT
NO
Label "E2EBestEffortWsTCP" RTTDiff: Min WS RTT - TCP RTT

Figure 5.6: **Decision Tree—**We create a decision tree that uses the round-trip times measured using our four different techniques. Based on their results, we use a combination of these values to decide whether a particular experiment can be labeled as a possible proxy connection. The two yellow nodes refer to cases that lack some of the measurement results and are hence, labeled "best effort" calculations.

5.4.1 Control Testbed Evaluation

We conduct a controlled evaluation of our CalcuLatency service by creating a testbed and collecting various measurements ourselves. In our experiment, we controlled the webserver that conducted the measurements. We had team-members run tests from their devices from fifteen different home and mobile networks using four different popular browsers—Chrome, Mozilla Firefox, Safari, and Brave. Our network locations include multiple cities in the United States, Canada, the United Arab Emirates, and India. We conduct both direct measurements from these networks and also run measurements while connecting to various different VPN and proxy servers around the world. We instrumented an automation script using Selenium to trigger measurements from multiple different browsers on a computer at the same time. In both the direct connection case and when we turned on a VPN, we use this automation script to conduct multiple experiments from different browsers towards the same server. The VPNs and proxies used to conduct these experiments include ten popular, commercial VPN providers including Astrill VPN, CyberGhost VPN, ExpressVPN, IPVanish, IVPN, Mozilla, Mullvad, NordVPN, Private Internet Access, Surfshark. These VPN providers offer multiple different protocols such as Wireguard, OpenVPN, and other proprietary protocols such as OpenWeb and Lightway. We also instrumented our own SOCKS5 proxy servers and tested them as well. We do not measure or reason about security and privacy aspects of the tested commercial VPN providers; our aim is to simply connect to various servers they offer and evaluate our CalcuLatency measurement service.

5.4.1.1 Data Characterization

In total, we conducted 891 unique experiments with 354 unique client IPs. Our measurements came from 337 unique VPN or proxy server IPs belonging to 82 different autonomous systems (ASes), and 17 unique direct, client IPs belonging to 12 different ASes. Overall, we collected data from four different countries.

5.4.1.2 Results

We calculate the RTT difference for each experiment using a decision tree that we devised a decision tree (Figure 5.6).

First, we check if the TCP handshake RTT (Δ_{TL}) $\gtrapprox$ websocket RTT (Δ_{AL}), i.e. if they are within ± $10\,\mathrm{ms}$. If so then the experiment is a network layer proxy or a direct measurement. In this case, the Δ_{TL} cannot be used for calculating the RTT difference. Hence, we use the ICMP RTT, if it exists, and if not, we will use 0trace RTT as Δ_{NL}. In order to differentiate and identify which value was used as Δ_{NL} and to record the conditions

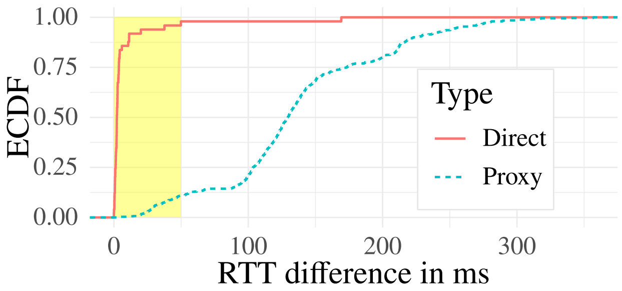


Figure 5.7: **ECDF of RTT difference for the 891 control, testbed evaluation measurements**—From our measurements, we find that over 98% of direct measurements and less than 11% of VPN measurements have an RTT difference below $50\,\mathrm{ms}$ (shaded in yellow).

that are met according to the decision tree, we assign labels for each end node of the tree. For instance, in this case, if ICMP RTT does not exist (i.e. ICMP was not supported by the client IP), we will use 0trace RTT and assign different labels based on whether 0trace reached the client IP (*0TClient), 0trace did not reach client but reached the same ASN as the client (*0TNetwork), and label it a "best effort" if none of these conditions is met but, we only have the 0trace RTT from the last successful hop to use for the calculation.

On the other hand, if $\Delta_{\mathrm{TL}} \lessapprox \Delta_{\mathrm{AL}}$, then the experiment could be that of an application layer proxy. In this case, we check if $\Delta_{\mathrm{TL}} \approx$ to either of the Δ_{NL} (ICMP RTT, 0trace RTT). If not, we follow the same procedure as above and use ICMP RTT if it exists, and if not, fallback to 0trace if it reaches the client IP or its ASN. Finally, if none of those conditions are met, we label it a "best effort" and use the TCP RTT to calculate the RTT difference.

In our control testbed evaluation, **we find that in 98.0% of the direct measurements (48 of 49) the calculated RTT difference is less than** $50\,\mathrm{ms}$, as shown in Figure 5.7. Upon investigating, the anomalous direct measurement belonged to an Indian mobile network provider, where the ICMP failed and 0trace was unable to reach even the same network, hence it was a "best effort" calculation. We will revisit the "best effort" cases in §5.4.3.

Next, we investigate the VPN measurements, we find that the RTT difference is above $50\,\mathrm{ms}$ in 89.1% of all VPN measurements (750 of 842), also shown in Figure 5.7. Among the rest of the 10.9% of VPN experiments, we find that in 60.9% of them (56 of 92), the

VPN server is close to the user geographically, i.e. the straight line distance between user and VPN server is less than 650 miles. This is equal to the straight line distance between Washington DC and Boston, MA (635 miles) or the straight line distance between Mountain View, California and San Diego, California (685 miles). This is expected, as specified in our assumptions, our technique primarily seeks to identify longer-distance remote VPN users.

We investigated the 14 unique VPN server IPs belonging to 36 of 92 measurements (39.14%) where the user to VPN distance was above 650 miles. We found that six of those 14 IPs belonged to AS9009 (corresponding to 17 of 36 measurements), and for all these IPs, the VPN provider claims their location to be in a Latin American country (Mexico, Costa Rica, The Bahamas, Venezuela, Chile, Argentina). However, when we investigated their ICMP RTTs as measured from our server in the U.S. Midwest region, we find that all their RTTs ranged between $28\,\mathrm{ms}$–$52\,\mathrm{ms}$. We used the speed of the Internet approximation provided by Katz-Bassett et al. [100] which is $\frac{4}{9}$ c, where c is the speed of light traveling in a vacuum, and find that for all of these experiments, **the locations of the VPN server as reported by the VPN provider is an impossibility**. Their true geolocation appears to be much closer than that what is reported.

For the remaining 19 experiments with eight unique VPN server IPs, we were not able to disprove the advertised location of the server with the RTT measurements. However, five VPN servers belonged to the same VPN provider which may have a policy to inflate ICMP values. The other two IPs did not have a reliable ICMP value for us to confirm or disprove their advertised location.

Overall, we find that $50\,\mathrm{ms}$ is a viable threshold for the RTT difference to distinguish a direct measurement from one coming through a remote VPN or proxy. We expand on this evaluation by conducting a real-world evaluation to increase the diversity of our “direct” connections and user locations.

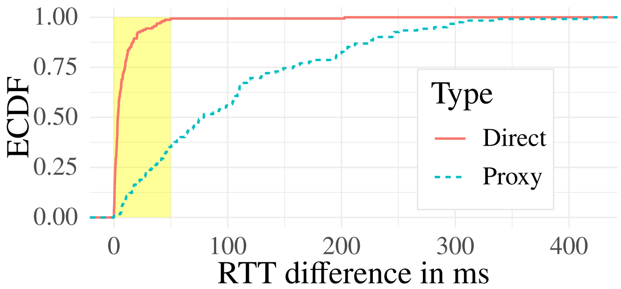


Figure 5.8: **ECDF of RTT difference for the 283 public crowdsourced evaluation measurements**—From our collected measurements, we find that over 98% of direct measurements and 36.1% of VPN measurements have an RTT difference below 50 ms (shaded in yellow).

5.4.2 Real-world Crowdsourced Evaluation

Next, we conduct an evaluation of our system in practice, using real-world measurements. We developed the CalcuLatency system, created a web server, and deployed it on a subdomain of our university. To obtain ground truth about the measurements, we create a form that users fill out to give us information regarding their setup. We ask users to optionally provide us with an email with which we can reach them, whether they are connecting to us though a mobile or a desktop, who their internet service provider is, and their current location (to reason about the measured latencies and physical distance). We then ask users if they are connecting to us via a VPN, or directly. If they are using a VPN, we request them to enter their VPN server location if known, and any details about their VPN provider.

We recruited users by publicizing a call for participation on Twitter using the authors' twitter handles and collected data for this experiment over a period of 15 days.

5.4.2.1 Data Characterization

We collected 283 unique experiments from 252 unique client IPs, with (self-reported) 161 direct measurements and 122 VPN measurements. Our 161 direct measurements came from 145 unique client IPs belonging to 93 different autonomous systems (ASes). Our 122 VPN measurements came from 109 unique VPN IPs belonging to 51 different ASes.

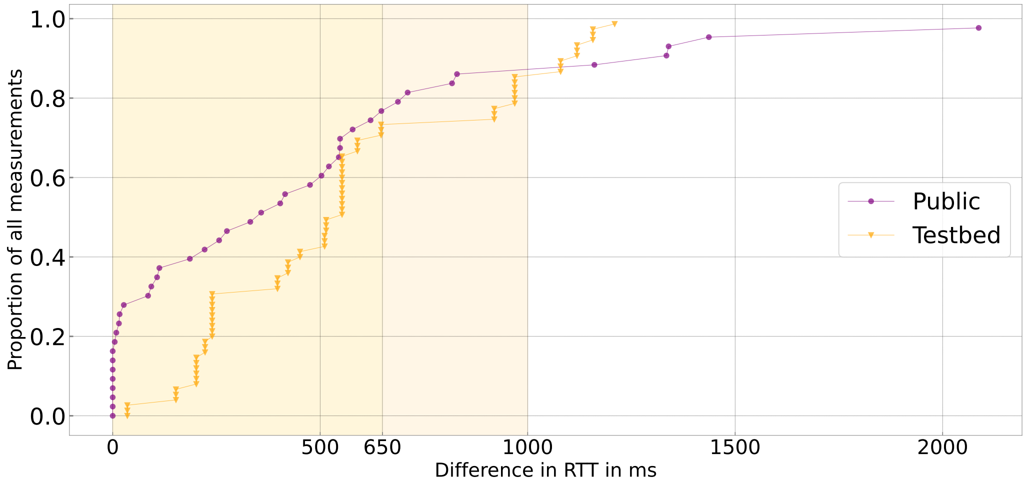


Figure 5.9: **CDF of the distance between user and VPN when RTT difference < 50ms—** Majority of VPN experiments with a low RTT difference (below 50ms) are located close to the user. 66.2% of these experiments, the proxy is only upto 650 miles away from the user (shaded in yellow), and 89% of these experiments, the proxy is less than 1000 miles from the user (shaded in orange).

Overall, we collected data from over 37 different countries from all (six) continents.

5.4.2.2 Results

Based on the same decision tree explained in §5.4.1 and Figure 5.6, we calculate the RTT difference and label each experiment with the appropriate label from the tree. **We find that 98.8% of all direct measurements have an RTT difference below** $50\,\mathrm{ms}$ (159 of 161) as shown in Figure 5.8, following the same trend as our controlled testbed evaluation, despite a large increase in the variety and diversity of our collected direct measurements. Upon investigating the remaining 1.2% (2 of 161) measurements, we find that one measurement was conducted through Safari and indicates that the IP is an iCloud Private Relay IP, we conclude that the user may have overlooked the fact that private relay was turned on, and hence this is a "proxy" measurement. The other experiment had an abnormally large 0trace RTT value, which is a measurement anomaly. This experiment was also labeled a "best effort" calculation (We explore this in §5.4.3).

Next, we investigate the VPN measurements, the RTT difference in 63.9% of measurements (78 of 122) are above the threshold of 50 ms, also shown in Figure 5.8. Of the 36.1% VPN measurements whose RTT difference is below 50 ms, we find that in 77.3% of the cases (34 of 44), the user and the VPN server are below 650 miles apart, which we consider as the minimum threshold for it be a "remote proxy", as shown in Figure 5.9. We use the locations provided by the user in our form and also compare the VPN IP-geolocation. Another 2.2% (1 of 44) was an experiment mislabeled as a VPN measurement, although the client IP indicates it is in the same location as the user. Of the remaining 20.5% of measurements (9 of 44), we did not see any proof to disprove the apparent location of the VPN.

5.4.3 Results Combining the two Evaluations

Overall, if we consider 50 ms to be the RTT difference threshold for a connection to be labelled as a remote proxy, our method has a low *false positive rate of 0.95%* (2/210 direct measurements). On the other hand, we find that 14.1% (136 of 964) VPN measurements had an RTT difference below this threshold. However, only 2.9% (28/964) of VPN measurements are located over 650 miles away from the user, and are legitimate (i.e. excluding VPNs with fake locations, and mislabeled experiments). Thus, we consider 2.9% our method's *false negative rate*.

What percentage of VPN servers respond to ICMP pings? In our evaluations, we collected a total of 434 unique VPN IPs, and we find that over 94.2% of these IPs (409/434) responded to ICMP pings. A majority of the VPN server IPs that did not respond to ICMP pings belong to Astrill VPN (14/25), which may have a policy for some/all of its servers to not respond to ICMP pings. Of the rest, four VPN servers belonged to personal OpenVPN VPN servers, three other servers others belonged to SOCKS5 proxies, and the remaining four were miscellaneous VPN services including iCloud Private Relay.

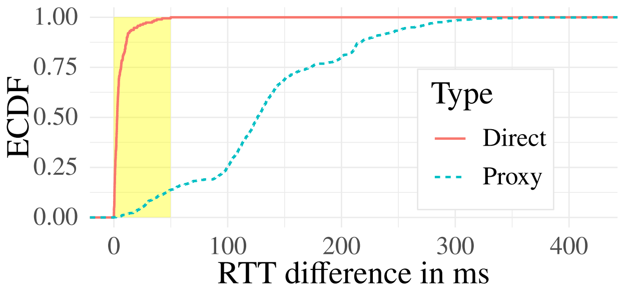


Figure 5.10: **ECDF of RTT difference of 1127 reliable measurements—**This analysis only contains measurements that did not have a “best effort” calculation label from our decision tree. 100% of all direct measurements and 86.2% of all VPN measurements have an RTT difference below $50\,\mathrm{ms}$ (shaded in yellow).

Does removing all best-effort calculations improve our method? Yes! Combining both sets of evaluation, we see that exactly 4.0% of all measurements (47/1174) was labeled “best effort calculations”, which means that the experiment did not contain successful ICMP measurements, and that TCP RTT measurement (Δ_{NL}) is too close in value to Δ_{AL}, and 0trace did not reach the client or even the same network as the client IP. Since these anomalous measurements only comprise 4% of all measurements, we do an investigation into what our analysis would look like if we removed these measurements. In production, we can achieve this by simply labeling all “best effort” experiments as requiring a re-run and have the client conduct another round of measurements.

Figure 5.10 illustrates this investigation, we see that a 100% of all direct measurements have an RTT difference less than $50\,\mathrm{ms}$, and 86.2% of VPN measurements (808 of 937) have an RTT difference greater than $50\,\mathrm{ms}$. Of the 13.8% (129 of 937) VPN measurements that have an RTT difference below $50\,\mathrm{ms}$, we find that 66.7% of measurements are less than 650 miles away from them user and another 13.2% advertise fake VPN server locations as we found in §5.4.1. The remaining 26 VPN experiments whose RTT difference is less than $50\,\mathrm{ms}$ but distance from the user is more than 650 miles are false negatives, and the average distance between the VPN server and user in these cases is 1089 miles.

In summary, in a more conservative analysis setting where we only consider the experiments where we obtain all measurement data points (96% of all collected measurements), *we do not find any false positives and false negative rate is 2.77% (26/937).* In production systems, the experiments that we exclude in this analysis, i.e. those that are labelled "best-effort" can be marked for deeper inspection and for re-running of the experiment to remove transient issues.

5.5 Ethics

Before running a public, crowd-sourced evaluation of our system detailed in § 5.4.2, we contacted our institution's IRB and were informed that our project is exempt from regulation. Our crowdsourcing evaluation web service presents a input form and did not trigger any measurements until the user reads, inputs data about the measurement, and consent to the measurements. The index page also outlines the measurements conducted, data collected, and that any latency data collected will be published as an open-source data set to help future research, after anonymizing the last octet of each visitor's IP address. From the web service, we measure the user's latency using four different methods and any data collected is only accessible to the authors of this paper.

5.6 Limitations

CalcuLatency cannot reveal proxy users in all cases. We can only detect proxy users if the proxy is sufficiently far enough from the user, from our measurements this distance appears to be at least 650 miles or more. Otherwise if the user and proxy are closer, the RTT difference may be too small to detect the proxy. This may be a prohibitive limitation for some service providers, and a non-issue for others because users may deliberately choose proxy servers that are geographically far away from them, to disguise their physical location. For example, users defrauding ad rewards networks want access to higher-reward locations, or users of streaming services often use VPNs to appear to be in another country, to get access to geo-restricted content. Each service operator can decide based on their business needs if they are conservative (treat ambiguous users as direct users) or be more aggressive (treat ambiguous users as proxy users). The service provider can also choose to subject ambiguous users' connections to further tests to confirm observations.

Our technique becomes less accurate in the presence of highly restrictive firewalls that discard all ICMP traffic. For perfect accuracy, we need to receive ICMP error packets from the client itself. The farther away we are from the client, topologically, the less accurate our RTT measurements become. While such restrictive firewalls do exist, our results suggest that they are the exception rather than the rule. We find from our evaluation that 94.2% of all VPN or proxy server IPs respond to ICMP pings, and over 56% of direct connections respond to ICMP pings, which is much higher than reported before. Additionally even if direct ICMP pings are disallowed, 0trace can function as long as the firewall does not discard ICMP error responses.

5.7 Related Work

We provide an overview of related work on latency measurement techniques and its applications, Internet liveness tests, and proxy detection methods.

5.7.1 Latency Measurements

In a 2021 blog post, Tschacher writes that using in-browser measurements and on the server side by measuring RTT of the incoming TCP/IP handshake, website owners can infer that a visitor is using a proxy or VPN using the latency difference [224]. However, the blog post conducts a limited evaluation and relies only on one incoming TCP/IP connections and the handshake to identify RTT which may be affected by packet loss and other transient factors.

Weinberg et al. [241] used ping-time measurements to hosts in known locations to estimate the locations of 2,269 proxy servers. They also found that over 90% of VPN servers they tested ignore ICMP pings and the first hop router for the VPN servers drop

ICMP pings and do not send time exceeded packets.

Hopper et al. [77] estimate that an Internet host (in 2007) can be uniquely identified by knowing its RTT to 5 other randomly chosen hosts. Only up to the RTT equivalence though (Same subnet RTT is indistinguishable etc).

Pelsser et al. [171] studied whether ICMP ping provides a good estimation of delay and they find that from an application perspective, ICMP ping actually gives a very poor estimate of the delay and jitter as it can vary between flows, and is dependent on the flow identifier. They propose an adaptation of paris-traceroute which is not biased by per-flow network load balancing.

Hoogstraaten [76] explored several server-side VPN detection methods, such as using existing IP information databases (WHOIS, rDNS), fingerprinting TCP options like advertised MSS, and even timing based measurements. But they only propose limited latency based measurements to identify internet-layer proxies, and conducted a short proof of concept. They measure the round-trip time of the three-way TCP handshake initiated by the client and compare it to an ICMP ping sent towards the proxy server. Apart from the limitations of their measurement and evaluation, their experiments were not extensive enough to confirm the reliability of this technique.

Jiang and Dovrolis propose in their CCR'02 article two techniques (one based on the TCP handshake) that a passive in-path monitor can use to measure the TCP round-trip time [95]. Unlike this work, CalcuLatency has access to bidirectional flows and can therefore analyze all three segments of the TCP handshake.

5.7.2 Internet Liveness

In their CCR'18 paper, Bano et al. perform Internet-wide scans to study the population of addresses that respond to probes [14]. They found ICMP probes are the most effective to discover alive hosts but TCP probes can further add to the population of alive hosts. A combination of ICMP, TCP, and (to a lesser extent) UDP results in the most complete picture of liveness. Bano et al.'s results inform how CalcuLatency attempts to determine an address's network-layer or transport-layer latency.

5.7.3 Proxy Detection

Much work has focused on detecting proxies; mostly on detecting Web proxies, VPNs, and Tor.

Before the pervasive deployment of HTTPS, in-path Web proxies would sometimes modify payload in flight. Reis et al. propose in their NSDI'08 paper to detect such proxies by augmenting Web pages with JavaScript that can detect modifications [195].

In a similar vein, Weaver et al.'s PAM'14 paper studied the prevalence of in-path Web proxies by sending controlled application-layer measurements between clients and a server, controlled by the researchers using Netalyzr [236]. Unlike these papers, we focus on network-layer proxies and rely on round trip times rather than application-layer measurements.

In an S&P'19 paper, Mi et al. studied the emerging ecosystem of end user systems that serve as proxies to others—often without the knowledge of the owners of the proxies [137]. Some proxy detection techniques assume that a proxy's IP address is owned by a data center but Mi et al. show that this assumption is increasingly incorrect. Mi et al. further show in an NDSS'21 paper that these residential proxy services also include mobile devices.

Internet censorship research made an effort to locate censorship devices, which are typically network proxies. In their CoNEXT'22 paper, Sundara Raman and Wang et al. propose a traceroute method that assists with identifying what hop on the path between client and server is conducting censorship.

The closest related work is by Webb et al. [238] who also proposed detecting proxies and VPNs based on traffic timing and latency. They measure the RTT for each connecting IP address and flag anomalies in the distribution of these RTTs as proxies. However, their method must be trained for each IP address, and hence is not practical.

Both Singh et al. and Lin et al., in their work, infer the use of a proxy by taking advantage of the latency patterns caused by either the presence or absence of Nagle's algorithm on the proxy [116, 207].

5.8 Discussion and Conclusion

In this paper, we present and evaluate our system CalcuLatency that incorporates several network RTT measurement techniques and leverage the application-layer and network-layer differences in roundtrip-times when a user connects to the service using a proxy (which are absent when the user connects to the service directly). We implement and evaluate each building block of our system individually: WebSocket RTT measurement on the application-layer, TCP handshake RTT recorded on the transport-layer, and ICMP ping and 0trace hop-enumeration RTT on the network-layer. We integrate all these techniques into one system called CalcuLatency and conduct a two-pronged evaluation: a control testbed environment where we test multiple different VPN products, proxy protocols, and server locations with over 337 unique VPN or proxy server IPs tested. To expand the diversity of our evaluation, we rally users and collect a public, crowdsourced, real-world evaluation of our system, which gained us 283 measurements from 37 different countries in all six continents.

Our evaluations reveal that a round-trip time difference of 50 milliseconds between the application and network-layer latencies is a viable, empirical threshold to consider a particular client as a remote VPN or proxy connection.

CalcuLatency provides a preliminary, labeling technique for service-providers, and must not serve as the only signal for detecting "malicious proxy traffic". Not all VPN users are attackers and not all VPN or proxy traffic is abusive. Hence, our method only serves as one of the signals, and service providers must leverage business-specific logic and make decisions on flagging connections as possible abuse traffic. For example, ad reward networks can flag certain connections as malicious and use such signals over time to detect and curb abusive ad consumers.

Ensuring the adoption, success, and sustainability of such novel, user-first business models relies heavily on the availability of computationally cost-effective and easily deployable techniques. These novel efforts to ‘re-calibrate the Internet” need to strike a delicate balance between implementing robust abuse-prevention mechanisms and maintaining a rigorous commitment to privacy. With CalcuLatency, we provide an open-source technique that can be readily deployed as a software service that can support efforts that prioritize user interests and their privacy.

CHAPTER VI

Conclusion

In this dissertation, I bring together various lines of inquiry to study privacy harms caused by commoditized filtering technology and the evaluate the ecosystem surrounding one of the most popular privacy-enhancing technologies—the commercial VPN ecosystem from security, privacy, and usability perspectives.

First, I presented work that highlights the growing issue of the commoditized deep packet inspection technology proliferating privacy harms by enabling censorship. We illustrate that Russia's censorship architecture enabled by commoditized deep packet inspection technology is a blueprint and even a forewarning of what national censorship regimes could look like in many countries that have similarly diverse ISP ecosystems to Russia's.

Next, I conducted empirical studies of VPN users and VPN providers using quantitative and qualitative methods to answer fundamental questions about the VPN ecosystem from the point of view of two of its most important stakeholders. We augment the survey of 1,252 U.S. VPN users with nine VPN technology developers' perspectives to holistically identify key areas of concern and bridge gaps in these ecosystems. Through this study, we put forth recommendations for the Internet freedom community, security and privacy advocates, consumer protection agencies.

Concurrently, I presented work investigating leading privacy-enhancing VPN tools products from a security and privacy standpoint. We built a systematic, rigorous, and semi-automated tool to conduct testing of existing tunneling VPN products. The test suite that we built contains 15 novel measurements that includes tests for aspects of service, security and privacy essentials, misconfigurations, and leakages including whether the VPN has implemented an effective mechanism to protect users during tunnel failure. Our investigation into 80 desktop VPNs using this tool reveals several novel findings such as evidence of traffic leaks during tunnel failure, IPv6 leaks, and more implementation shortcomings.

Finally, I conducted a study of VPN misuse from serverside operators' perspective. I developed and presented a system, CalcuLatency, that leverages cross-layer network latency measurement techniques to identify malicious actors that use a remote, long-distance VPN or proxy. I conducted a two-pronged evaluation, and provided an empirical latency-difference threshold (between application-layer latency and network-layer latency) to identify potential VPN and proxy traffic. This system is soon going to be deployed into production to add to Brave Software's suite of anti-fraud, abuse-prevention techniques.

Through my dissertation work, I strive to make the commercial VPN ecosystem less opaque, improve our understanding of VPN users and VPN providers and bridge the gaps between them, develop a safer approach centered on networking to detect misuse of VPNs for malicious activities, empower users around the world to test and evaluate their VPN providers with our tool, and facilitate their access to the right tool for their specific needs.

APPENDICES

APPENDIX A

Appendix for Emerging Censorship Techniques: A Case Study of Russia

A.1 Validating the Russian Blocklist

To validate our source of historical blocklists at [193], we obtained access to a small set of blocklists digitally signed by Roskomnadzor through a few different anonymous sources. To get the corresponding historical blocklists from the Zapret source, we searched it for the date and timestamp closest to that in the anonymously-supplied blocklists. None of the date and timestamps were a perfect match between the two sources, leading us to believe that the Zapret information has a different source than the small set of blocklists we obtained from our anonymous sources.

Using the closest version of the Zapret source, we pre-processed the contents of both blocklists. This included extracting all IP addresses and all domains in each of the files, resulting in sets of IP addresses and domains to compare between the two different sets of blocklists.

We compared the two sources' sets of IP addresses and domains using the Jaccard index of similarity, which is calculated by taking the size of the intersection of the two sets and

Date	IPs Only	Domains Only	IPs & Domains
2017-06-13	1.0	0.99998	0.99999
2018-04-27	0.99994	0.99867	0.99805
2018-05-13	0.99733	0.99998	0.99996
2018-11-08	0.99996	0.99999	0.99997

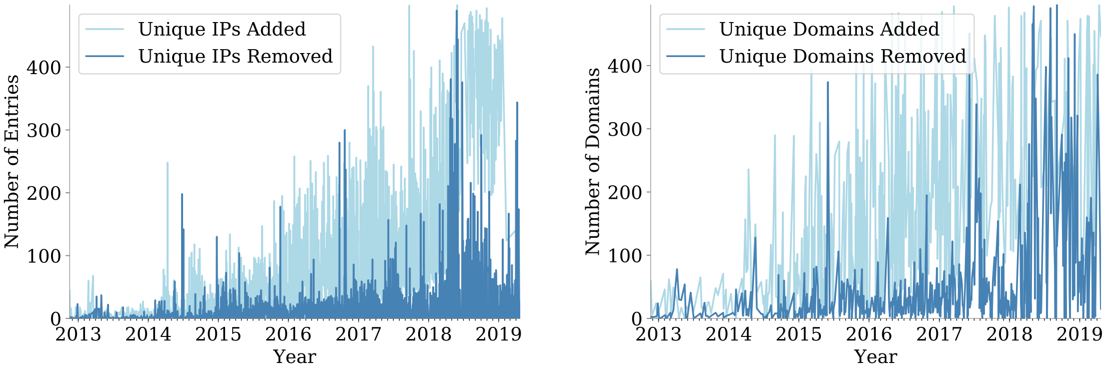
Table A.1: **Zapret-supplied blacklists' similarity to anonymously-supplied blacklists signed by Roskomnadzor, using the Jaccard index for each category as mentioned in the column name.**

dividing by the size of the union of the two sets. The Jaccard index is a number between 0.0 and 1.0, where 0.0 represents no similarity and 1.0 represents completely similar sets.

Applying the Jaccard index to our blocklists from different sources (which was signed by Roskomnadzor), we found that the Zapret blocklists are *extremely* similar to the signed blocklists. Our results are shown in Table A.1. We analyzed the similarity of the sets of IP addresses, domains, and the entire set of all IP addresses and domains combined. All sampled blocklists have a similarity greater than 0.99 for any given content type (IP, domain, or IP & domain). Based on these findings, we conclude that the Zapret source of blocklists is representative of the list produced by Roskomnadzor, and is both correct and complete, and thus sufficient for our analysis in this work.

A.2 Analysis of $RUBL$

Figure A.1a shows how the number of unique IPs added per day outpaces the number of unique IPs removed per day, further displaying the rapid growth of $RUBL$. In addition, the time series plot shows significant volatility. Significant events such as court rulings restricting a certain service will lead to a spike in number of IPs added. On the other hand, media traction of specific collateral damage instances will lead to a spike in number of IPs removed. Regardless, days without any significant activity is prevalent, leading to many downturns in the graph. Since the blacklisting of specific sites require no court ruling, IP address additions rarely fall to zero. The opposite is true for address removals, which often



(a) IPs Added vs. IPs Removed.

(b) Domains Added vs. Domains Removed.

Figure A.1: **Blocklist volatility over 7 years**—The two subfigures shows the volatility of the blocklist, with many spikes and downturns in response to real world events.

fall to zero due to the degree of time and difficulty involved for content owners to initiate an official removal procedure. Similar to Figure 2.2, Figure A.1a also shows a rise in addition of unique IPs and decrease in removal of unique IPs in 2019, suggesting that the blocklist is being handled more carefully recently. We observe the same trends with domains in Figure A.1b, although there is more variability.

A.3 Blockpages observed through DNS Poisoning

Figure A.3 shows three block pages received at Probe 9, Probe 14, and VPS 6 due to DNS poisoning, as discussed in § 2.5.2.

A.3.1 $RUBL_{sub}$ Measurements

As we mentioned in § 2.4.1, the $RUBL_{sub}$ consists of 39 subnets, ranging from /16s to /24s. 31 out of 39 of these subnets contain at least one IP reachable to one of our controls. Of the remaining eight subnets completely unreachable from our controls, seven belong to Telegram and all eight are geolocated to Moscow.

Figure A.2 shows the percentage of blocking in each of the 31 subnets in $RUBL_{sub}$ that were reachable in our controls, where percentage of blocking is the number of IPs unreachable out of total number of IPs in the reachable subnets. Two subnets, Subnet 16

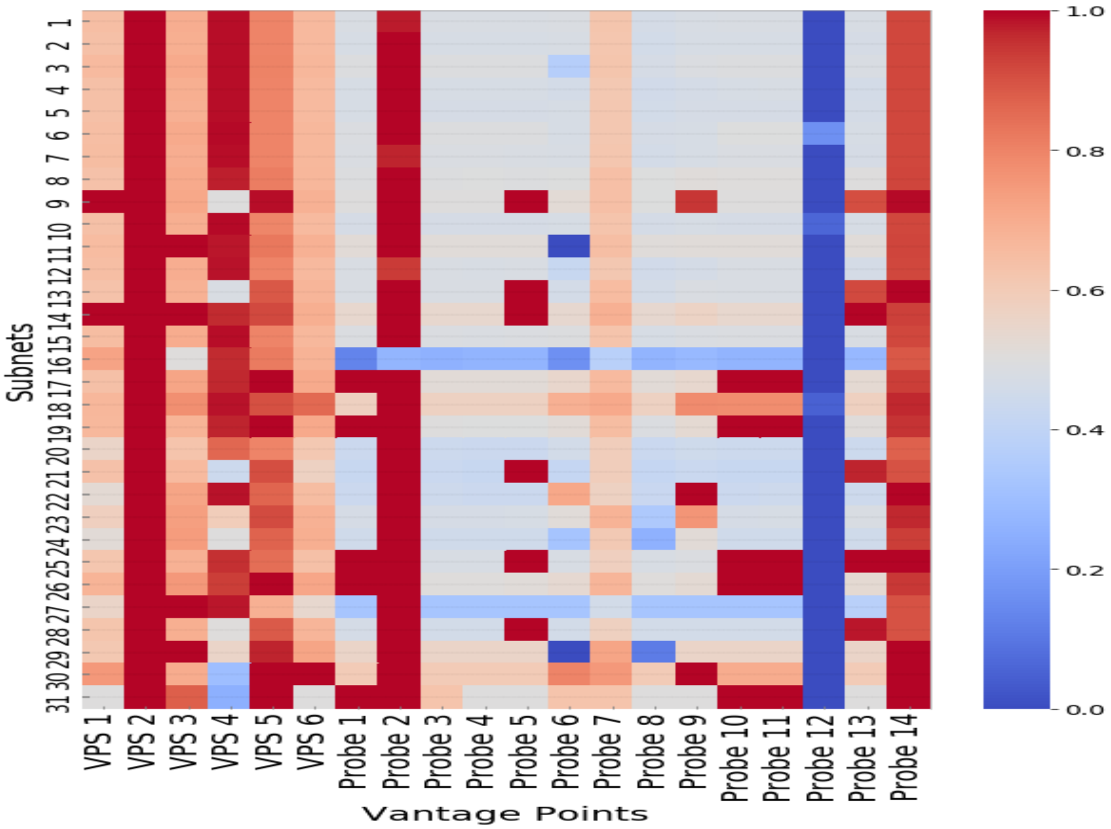
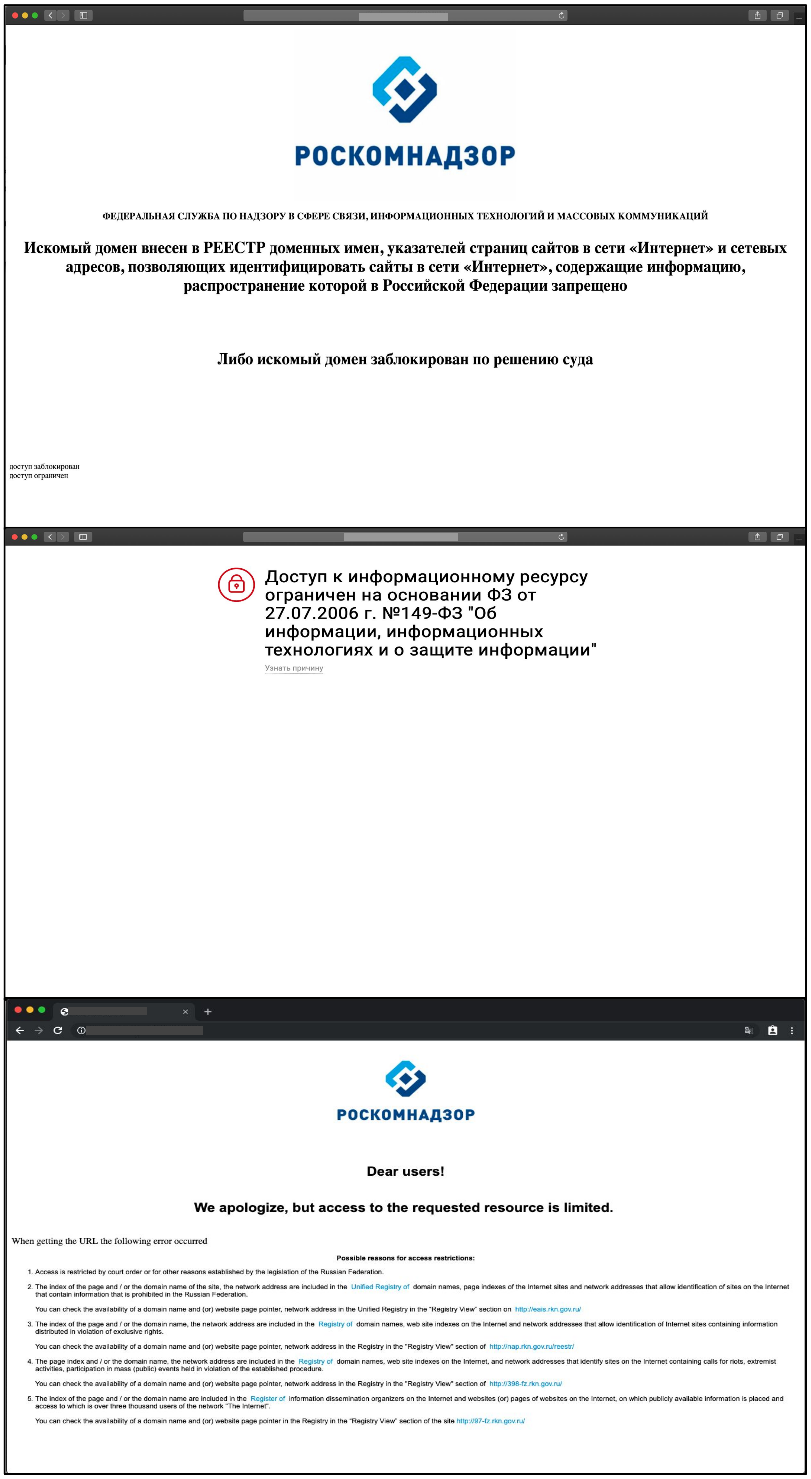


Figure A.2: **Blocking per subnet when testing $RUBL_{sub}$ on VPSes and Probes**— Datacenter vantage points observe a large percentage of blocking in almost all subnets. Residential vantage point comparatively block intensively in fewer subnets.

and 27, see much lower rates of blocking at most of our probes. These subnets belong to Cloud South, a U.S.-based hosting provider, and UK2, a UK-based hosting provider. Several of the other subnets belong to providers such as DigitalOcean, so it is unclear why these two subnets see less residential blocking, though it might pertain to collateral damage associated with blocking them. Another interesting feature of this analysis is that blocking appears to be correlated with the size of the subnet: larger subnets are blocked more, by both probes and VPSes.



[redacted]

РОСКОМНАДЗОР

ФЕДЕРАЛЬНАЯ СЛУЖБА ПО НАДЗОРУ В СФЕРЕ СВЯЗИ, ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И МАССОВЫХ КОММУНИКАЦИЙ

Искомый домен внесен в РЕЕСТР доменных имен, указателей страниц сайтов в сети «Интернет» и сетевых адресов, позволяющих идентифицировать сайты в сети «Интернет», содержащие информацию, распространение которой в Российской Федерации запрещено

Либо искомый домен заблокирован по решению суда

доступ заблокирован
доступ ограничен

[redacted]

Доступ к информационному ресурсу ограничен на основании ФЗ от 27.07.2006 г. №149-ФЗ "Об информации, информационных технологиях и о защите информации"

Узнать причину

[redacted]

РОСКОМНАДЗОР

Dear users!

We apologize, but access to the requested resource is limited.

When getting the URL the following error occurred

Possible reasons for access restrictions:

1. Access is restricted by court order or for other reasons established by the legislation of the Russian Federation.
2. The index of the page and / or the domain name of the site, the network address are included in the Unified Registry of domain names, page indexes of the Internet sites and network addresses that allow identification of sites on the Internet that contain information that is prohibited in the Russian Federation.

   You can check the availability of a domain name and (or) website page pointer, network address in the Unified Registry in the “Registry View” section on http://eais.rkn.gov.ru/
3. The index of the page and / or the domain name, the network address are included in the Registry of domain names, web site indexes on the Internet and network addresses that allow identification of Internet sites containing information distributed in violation of exclusive rights.

   You can check the availability of a domain name and (or) website page pointer, network address in the Registry in the "Registry View" section of http://nap.rkn.gov.ru/reestr/
4. The page index and / or the domain name, the network address are included in the Registry of domain names, web site indexes on the Internet, and network addresses that identify sites on the Internet containing calls for riots, extremist activities, participation in mass (public) events held in violation of the established procedure.

   You can check the availability of a domain name and (or) website page pointer, network address in the Registry in the "Registry View" section of http://398-fz.rkn.gov.ru/
5. The index of the page and / or the domain name are included in the Register of information dissemination organizers on the Internet and websites (or) pages of websites on the Internet, on which publicly available information is placed and access to which is over three thousand users of the network "The Internet".

   You can check the availability of a domain name and (or) website page pointer in the Registry in the “Registry View” section of the site http://97-fz.rkn.gov.ru/

Figure A.3: **Three example blockpages.**

APPENDIX B

Appendix for Empirical Understanding of VPN users and VPN providers

B.1 Difficulty in VPN discovery (RQ1)

Population	VD/SD/NE/SE/VE	**V. Difficult%**	**S. Easy %**
High expertise	19/139/158/ 108 /87	3.7↓	21.1↑
Moderate expertise	44/180/198/116/93	7.0	18.4
Limited expertise	13/22/39/12/23	11.9	11
Population	Diff/Neither/Easy	**Difficult%**	**Easy %**
Paid/Premium	337/319/334	34	33.7
Free	64/49/44	40.8	28.0
Other (Uni./other)	16/26/60	15.7	58.8↑

Table B.1: Number and % of users from different user groups indicate how difficult it was decide on a VPN to use (from VD-Very Difficult to VE-Very Easy). Symbols indicate ↑ more, and ↓ less likely than the other rows in the column.

Finding a suitable VPN is typically not a trivial task and we report in §3.4.1 that user response to the query of whether they had difficulty in selecting a VPN provider, is almost evenly spread over the difficulty scale. However, we did find differences between users with varying security and privacy expertise shown by a χ^2-test (p = 0.004206, with N=1251). As mentioned in 3.2.1, we perform pairwise z-tests (α=0.05) with FDR-BH correction to find how different user groups relate to each other.

We also find significant difference between users that use different subscription types (free, paid/premium, other) also shown by a χ^2-test (p = 0.000005, with N=1249). We report on these differences in §3.4.1 and these findings are detailed in Table B.1.

B.2 Emotional connection with VPN for different user expertise (RQ3)

As shown in 3.4.3, in general, users indicate they feel unsafe without a VPN. We find that there are no significant differences between users with varying expertise levels and their perception of safety without VPNs, as shown by a χ^2-test (p = 0.085, N=1252). We notice that less limited expertise users indicate that they feel at least somewhat safe without a VPN (only 20%, 22 of 110 as compared to 30.3% of the high- and 29.5% of the moderate expertise users).

While this is not a statistically significant difference, we explored the reason they do not feel safe without a VPN by analyzing their textual response immediately after this question. Limited expertise users who responded (98 of 110) mainly express worry (about hacking, tracking, and more), and confusion about what VPN offers, and explain scenarios where they feel unsafe. S99 says:

> "One never knows when either the so-called good guys or the bad guys are lurking about, just waiting to pounce. In my book, I want to be safe rather than sorry[...]"

In general, users indicate they feel safer browsing the Internet with a VPN. In a different section of the survey, we ask them about the perception of safety while using a VPN. We find significant differences between users with varying expertise levels and their perception of safety with VPNs as well, shown by a χ^2-test (p=0.003, N=1252). While large sections of all populations feel somewhat or very safe (86.7%, 1,086) using a VPN, limited expertise users are *significantly less* likely to indicate they felt safe using a VPN (75.5%, 83 of 110) compared to 88.3% (557 of 631) of moderate- and 87.3% (446 of 510) of

Population	Safety without VPN		Safety with VPN	
Expertise	VS/SS/NO/SU/VU	**S%**	VS/SS/NO/SU/VU	**S%**
High	(22/133)/48/231/77	30.3	(202/244)/30/21/13	87.3
Moderate	(19/167)/44/324/77	29.5	(179/378)/38/27/9	88.3
Limited	**(5/17)**/13/52/23	**20.0**↓	**(27/56)**/18/8/1	**75.5**↓

Table B.2: Number and % of users with different security and privacy expertise and their feeling of safety when browsing without and with a VPN (from VS-Very Safe to VU-Very Unsafe). Symbols indicate ↑ more, and ↓ less likely than the other rows in the column. Highlighted values indicate that they contribute to the relevant percentage.

high expertise users), summarized in Table B.2. We also find that instead limited expertise users were significantly more likely to indicate they had no opinion on safety while using a VPN (16.4%, 18 of 110), possibly due to confusion on what a VPN provides. S1153 says, who indicated no opinion says:

> "I feel both somewhat unsafe and somewhat safe"

B.3 Codes from qualitative survey responses

B.3.1 Reasons for use

Privacy from ISP (22), **Privacy:** Privacy (17), from tracking (10), from tracking and ads targeting (5), surveillance (3), securing browsing history (3), hiding location (2), from ads (2), selling my data (2), hacking (2), from attribution (1), banking (1), during searching (1), ISP and large companies (1), **Security:** during banking (4), hackers (4), as a principle (2), paranoia (1), confidential/sensitive data (2), OpSec (1), hackers/surveillance and bad actors (1), protection (1), **Offered the service:** by Norton (4), free with other service (3), with router (1), by ISP (1), for low price (1), with device (1), from employer (1), **While travelling:** surveillance countries (2), protection from local actors (2), censoring countries (2), in general (2), don't trust hotels (1),**Anonymity** (3), **Access geo-restricted content** (2), **Work with tech** (1), **Safeguard device** (1), **No-log VPN** (1), **for IPTV** (1), **For work/uni** (1), **Browsing from different locations** (1)

B.3.2 Other Resources Used

Part of Software/Security Suite (102), **Trusted service provider** (27), **Prior Experience** (15), **Reviews:** Consumer Reports (13), Offered the service for free (13), Introduced as part of my job (8), thatoneprivacyguy (7), **Own testing** (7), Trying the trial option (7), **Word of Mouth:** from technical staff (6), Recommended by service (2), Computer Clubs (2), Colleague (2), University (1), Indiegogo (1), Meetings (1), Geek Squad (1), Friend/Family (2), Computer services company (1) **Trust:** the Mac App Store (5), the Google play store (1), Tech YouTubers (1), Leo Laporte (1), privacytools.io (1), Bloggers, Apple News+ (1), Expert reviewer (1), **No choice in VPN provided** (4), **Company Announcements** (5), **Reviews:** Reviewer Kim Komando (3), Specific to MacOS (3), PCMag (3), News Articles (3), Recommendation Sites (2), Trusted sources (2), NYT (2), Local tech advisor (2), testmy.net (1), ZDNet (1), Recommendation on YouTube (1), Print Magazines (1) **Advertising:** Ads on trusted podcast (3), Promos (3), in specific site (1) **No Research** (3), **Provider's website** (1)

B.3.3 Feeling of Safety without VPN: Limited expertise users

Worry: hacking (13), tracking (6), exposed personal details including IP and location (10), unsafe in today's world (4), bad actors (3), dark web/net (2), ISP access data (1), open to exploitation (1), malware (1), less protected (1), happened to others (1), breaches (1), ID theft (1), prevention (1), fear threats and financial data (1), **Confusion:** don't understand (10), what does ISP do with data (1), service stopped working (2), **Safety:** no prior issues (5), I'm careful (3), I have anti-virus (2), using VPN makes me safer (2), use trusted provider (3), my device is safe (1), I am trusting (1), added protection (1), **Scenario:** only unsafe in public networks (4), I use it if I have it (1), no reason (1), HTTPS isn't always available (1), **No worry:** I feel okay (2), **Understanding:** with research (1), its supposed to hide me (1), anyone can see my traffic (1), **Specific needs** (1), **Needs:** trade-offs (1), harder for hackers (1), make me safer (1).

B.3.4 High-expertise users response to mental model

DNS Provider (7), **Site:** if entered personal info (6), has access to cookies (4), might know (5), if insecure protocol (1), **VPN Provider:** logging (5), depending on service (4), surely knows (3), alone cannot hide you (2), audit trail (1), **ISP knows if DNS leaked** (4), **Other actors:** example site's partners (3), large companies like Google/Facebook (3), Browser (2), search engine (2), ad networks (2), IDSs (1), badly implemented tech (1), **Threat actors:** tracking (3), government agencies (2), third party cookies (2), browser fingerprinters (2), hackers (1), **Idk:** nobody if no logging (2), any hop in between VPN and site (1).

B.4 User Survey

We first display the survey landing page containing the consent form and introduce the survey. Then, we display the following questions:

Demographics & General Questions about Internet Usage

In this section, we ask a few general questions to collect demographic information, Internet habits, and self-reported security and privacy knowledge rating.

Q1. Gender*

○ Woman ○ Man ○ Non-binary ○ Prefer not to disclose ○ Prefer to self-describe

Q2. Age Range*

○ 18–25 ○ 26–35 ○ 36–45 ○ 46–55 ○ 56–65 ○ >65 ○ Prefer not to disclose

Q3. Country of Residence*

Dropdown list of countries

Q4. Highest Level of Education*

○ Some education, but no high school diploma or equivalent ○ High school degree or equivalent (e.g. GED) ○ Some college, no degree ○ College or university degree (for example a bachelor's or associate's degree) ○ Post-graduate education (for example a master's or a doctorate degree) ○ Other (Please elaborate below) ○ Prefer not to disclose

Q5. Is your field of study connected to computer science and/or technology?*

○ Yes ○ No ○ Prefer not to disclose

Q6. On a general day, how many devices (e.g. mobile phone, laptop, tablets) do you use to browse the internet?*

○ 1 ○ 2 ○ 3 ○ 4 ○ 5 ○ 6 ○ 7 ○ 8 ○ 9 ○ 10+

Q7. How would you rate your knowledge about privacy & security on the Internet?*

○ 1 - No knowledge - I do not have any knowledge of privacy or security concerns pertaining to the Internet ○ 2 - Mildly knowledgeable - I've heard of things such as VPNs, proxies, or Tor, but don't really understand how they work and/or have limited experience using them; I cannot describe specific dangers to privacy or security online but I know they exist. ○ 3 - Moderately knowledgeable - I can name some of the dangers to privacy online and am wary of them in general; I know about privacy tools and have tried using some of them. ○ 4 - Knowledgeable - I have a technical understanding of the threats that exist online and what tools can be used to mitigate them ○ 5 - Expert - I conduct research or work in a field related to Internet privacy and security

VPN Usage

In this section, we aim to understand the reasons why you use a VPN and your typical VPN usage patterns. Our definition of "VPN product" is any VPN service that you may have used *for personal use. The service may have been free, on a trial basis, a subscription based service, or a one-time fee type service. We also include school/university provided VPNs in our definition. It does not include: Corporate (workplace specific) VPN configurations, or VPNs set up by an individual (such as using Algo, Outline, or Streisand)*

Q8. Have you ever used a commercial VPN service as defined above?* Examples include products like NordVPN, ExpressVPN; Free services like Hotspot Shield Free, TunnelBear Free, Psiphon, and Lantern; University VPNs provided to you (accessible using your university credentials).
○ Yes ○ No

Q8A. What type of subscription do you generally use?
○ Free/trial version of a VPN service ○ Paid/premium version of a VPN service ○ Does not apply (University or Custom VPN) ○ Other (Please elaborate below)

Q9. Why do you use a VPN product? (Choose all that apply)
□ To access school or work networks remotely □ To protect myself from various threats/adversaries □ To access content blocked in my network (eg. due to censorship) □ My IP address is blocked from certain websites □ I'm interested in the technology behind it □ To make public networks safer to use □ For file sharing (e.g. torrents) □ To access region-specific content e.g. on Netflix, Hulu, BritBox, News □ It was a free service offered to me □ Other (Please elaborate below)

Q10. How frequently do you use a VPN?
○ Occasionally ○ Every week ○ Every day ○ All the time/Always on ○ Other (Please elaborate below)

Q11. To make sure that you’re still paying attention, please select only “Yes, more than once" in the options below

○ Yes, once ○ Yes, more than once ○ No ○ I don’t know

VPN Discovery and Choosing a VPN Product

In this section, we aim to understand how you discover VPN products, and the decisions and trade-offs that you may have made while choosing a VPN provider.

Q12. How difficult was it to decide which commercial VPN product/app to use?

○ Very difficult ○ Somewhat difficult ○ Neither easy nor difficult ○ Somewhat easy ○ Very easy

Q13. Please explain the reasons for your choice above.

Q14. There are many resources that are aimed at helping users choose a VPN provider. We are interested in learning about all the resources that you used in your journey to select a VPN provider. What resources did you use to select your VPN provider? (Choose all that apply)

□ Actively researching on the Internet e.g. using a search engine □ Recommendations from friends and family □ Reading the VPN provider websites □ Digital Training Workshops □ I randomly encountered them while browsing the web, through advertisements □ Recommendation websites e.g. TechRadar, CNET, Top10vpn □ User review posts e.g. Reddit, YouTube □ My work or school/university provides me a VPN □ Conferences and events □ Other (Please elaborate below)

The following question is displayed for each resource selected in Q14.

Q14A. In hindsight, how would you rate the level of credibility & trustworthiness of the resources you selected for the previous question?

(Options available for each resource are ○ Not Trustworthy ○ No Opinion ○ Moderately Trustworthy ○ Extremely Trustworthy)

Q15. Please rate the importance of the following criteria while selecting a VPN provider.*
(Options available for each criteria are ○ Not preferred/Indifferent ○ Preferred, but not a dealbreaker ○ Required, a dealbreaker)
→ Speed (Quality of service) → Price of the service → Easy to understand/use app (GUI) → Well-documented features → Ability to change location to access media on websites e.g. Netflix, Hulu, or News → Variety/number of servers located in different countries around the world → Clear explanation of logging and data practices

The following question is displayed for each criteria selected in Q15.
Q15A. Please move the tiles to rank the importance of the criteria you rated "Preferred" or "Required" in the previous question (1 being most important).*

Q16. Please select the names of all the commercial VPN providers you have tried/used before.
□ NordVPN □ StrongVPN □ ExpressVPN □ Hide.me □ IPVanish □ Speedify □ Hotspot Shield □ Psiphon □ TunnelBear □ Calyx VPN □ Windscribe □ Mullvad VPN □ Private Internet Access (PIA) □ PrivateVPN □ CyberGhost VPN □ TorGuard □ HideMyAss □ Hola Free VPN □ ProtonVPN □ Astrill VPN □ Norton Secure VPN □ VyprVPN □ SurfShark □ Mozilla VPN □ PureVPN □ Others (please separate by commas)

Q17. Is there anything else about the process of VPN discovery that you wish to share with us?

Mental Model of VPNs and Your Personal Threat Model *In this section, we ask questions about how you think VPNs work and your understanding of their data collection practices.*

Q18. How safe or unsafe do you feel when browsing the internet without a VPN?*
○ Very unsafe ○ Somewhat unsafe ○ No opinion ○ Somewhat safe ○ Very safe

Q18A. Please elaborate on why you feel that way when browsing without a VPN.

Q19. Imagine you are using a VPN and you open http://www.example.com, who do you believe knows that you have visited http://www.example.com? (Choose all that apply)
□ My Internet service provider (ISP) □ My VPN provider □ The owner of example.com □ Nobody □ I don't know □ Other (Please elaborate below)

Q20. Do you use a VPN for the purpose of securing or protecting your browsing activity?*
○ Yes ○ No

Display the following question if Q20: = Yes
Q20A. Since you selected that you use a VPN to secure or conceal your browsing activity, who do you want to protect it from? (Choose all that apply)
□ My Internet service provider (ISP) □ My School/Employer □ Friends and family □ Advertising companies □ Hackers/Eavesdroppers on open WiFi networks □ My government □ Other governments □ I do not use a VPN for this purpose □ Other (Please elaborate below)

Q21. To make sure that you're still paying attention, please select only "Somewhat safe" in the options below

○ Very unsafe ○ Somewhat unsafe ○ No opinion ○ Somewhat safe ○ Very safe

Q22. What data do you think is being collected about you by your VPN provider? (Choose all that apply)
□ My geolocation □ Timestamps of when VPN is in use □ VPN servers that I connect to □ Websites visited □ Interests/Preferences for ads □ Demographics and account holder information □ Device types □ Private messages □ Audio/Video collected from my device □ Keystrokes recorded from my keyboard □ I am not sure □ Other (Please elaborate below and separate by commas)

Q23. Why do you think your VPN provider collects this data about you? (Choose all that apply)
□ Internal analytics and quality of service reasons □ Advertising □ User tracking □ Political motives (government mandated) □ Crime investigation (for law enforcement) □ Selling information to third parties □ I am not sure

Q24. Please use this text box to share your views on data collection by VPN providers

Expectations about VPN service

In this section, we aim to discover what you expect from your VPN provider in terms of quality of service, privacy, and security. We aim to understand how VPN providers can build trust with you as a user.

Q25. How safe or unsafe do you feel while using your favorite VPN as compared to browsing without the VPN?

○ Very unsafe ○ Somewhat unsafe ○ No opinion ○ Somewhat safe ○ Very safe

Q25A. Please elaborate on why you feel that way when browsing with your favorite VPN.

Q26. Please rate the level of concern you would have about the following VPN-related issues.
(Options available for each concern are ○ Not at all concerned ○ Slightly concerned ○ Moderately concerned ○ Very concerned ○ Extremely concerned)
→ VPN provider logging my activity → VPN provider selling my activity data → VPN servers' geographic location not as advertised → Lack of transparency or documentation → VPN client software containing malware → Misconfigured VPN servers leaking some data → VPN not communicating to me that the connection has dropped (VPN tunnel failure)

Q27. Which of these efforts would increase the trust you have towards a VPN provider?
(Options available for each effort are ○ No Opinion ○ Not at all important ○ Slightly important ○ Moderately important ○ Very important ○ Extremely important)
→ Independent Security Audits → Clear Logging Policy → Response to Legal and Law Enforcement Requests (e.g. Warrant Canary) → Security Protocols and Disclosure of Breaches → Endorsements from NGO and/or academics

End of Survey

Q28. Finally, is there anything related to commercial VPNs that you wish to share with us?

B.5 VPN Provider Interview Questionnaire

1. *[Biggest Challenges]* Briefly tell us, what are the biggest problems that you see in the VPN ecosystem?

2. *[User Base]* Who is your main user base? There are many reasons why people use

VPNs like circumvention, privacy from ISP etc, do you know why your users are using your VPN?

3. *[Features]* What features do you think users care about the most? What are the features you focus most development efforts on?

4. *[Mental Models]* Do you think users have an accurate/good mental model of how VPNs work? Where do usability issues and frustration with VPNs usually come from?

5. *[Pricing]* From a user's perspective, we've also seen that pricing is a key criteria. How does the pricing work at your VPN? How does pricing in the industry work, generally?

6. *[Marketing/Recommenders]* Do you actively reach out to VPN recommenders to try your product? What strategies do you use to market your product?

7. *[Trust]* What are your efforts to build trust with users? Do you have third party security audits, why do you think they help? What are other efforts to build trust with users?

8. *[Dark Patterns]* What are some dark patterns and shady practices in the VPN ecosystem?

Miscellaneous (Extra questions, when we have time):

X1. How do you think these issues can be fixed? How to combat these issues in the future?

X2. Are there technical challenges in implementation, like IPv6 for example? What are some of the barriers for these challenges?

X3. What are your bug disclosure models?

X4. Do you have special circumvention technologies and obfuscation developed or implemented for censorship circumvention users?

X5. How do you reconcile with laws of the land where you have your servers?

APPENDIX C

Appendix for VPNalyzer

C.1 Recommendation Websites

Table C.1 shows the 25 recommendation websites used to create the set of popular VPN providers, which we augment with the four self-hosted VPNs and our institutional VPN.

Recommendation Websites Used
https://www.security.org/vpn/best/
https://www.techradar.com/vpn/best-vpn
https://www.cnet.com/news/best-vpn/
https://www.tomsguide.com/best-picks/best-vpn
https://www.pcmag.com/picks/the-best-vpn-services
https://thebestvpn.com/
https://www.wired.co.uk/article/best-vpn
https://www.zdnet.com/article/best-vpn/
https://www.cloudwards.net/best-vpn/
https://www.internetsecurity.org/compare/usa
https://www.top10vpn.com/best-vpn-for-usa/v/d/
https://bestvaluevpn.com/usd/best-vpn/?utm_campaign=ggls-en-usa-gen
https://www.nytimes.com/wirecutter/reviews/best-vpn-service/
https://cybernews.com/best-vpn/
https://vpnoverview.com/best-vpn/top-5-best-vpn/
https://www.guru99.com/best-vpn-usa.html
https://www.crazyegg.com/blog/best-vpn-services/
https://www.forbes.com/advisor/business/software/best-vpn/
https://blog.flashrouters.com/vpn/
https://vpnpro.com/best-vpn-services/
https://bestvpn.org/best-vpns-for-the-usa/
https://www.safetydetectives.com/best-vpns (formerly thatoneprivacyguy)
https://www.tomsguide.com/best-picks/best-free-vpn
https://www.top50vpn.com/best-vpn
https://www.top10vpn.com/best-vpn/
Total: 25

Table C.1: **Websites Used**—The top 25 websites used to create a set of 34 of the most popular VPN providers. Buffered VPN was among the 34 but is currently non-operational.

C.2 Full Results

Table C.2 shows a complete summary of our findings.

			Leaks during tunnel failure		Security & Privacy Essentials					
#	VPN Provider	IPv6 Support	All Data Leak	DNS Leak	DNSSEC Validation	Qmin Support	RPKI Validation	DoH Disabled?	Does it use Public DNS?	DNS Proxy
1	1.1.1.1 + Warp Cloudflare	🟩	Leak not detected	Yes, leaks DNS	🟩	🟩	🟩	🔴	🔴	🔴
2	AirVPN	🟩	Leak not detected	Leak not detected	🔴	🔴	🔴	🟩	🔴	🔴
3	Algo	🟩	Leak not detected	Leak not detected	🟩	🟩	🟩	🟩	🟩	🔴
4	Anonine	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🔴
5	Astrill VPN	Leaks to ISP	Leaks traffic**	Yes, leaks DNS**	🔴	🟩	🔴	🔴	🟩	🔴
6	Atlas VPN	🟩	Leak not detected	Leak not detected	🟩	🟩	🟩	🔴	🔴	🔴
7	Avast Secureline	🔴	Leak not detected	Leak not detected	🔴	🔴	🔴	🔴	🔴	🔴
8	Avira Phantom	🔴	Leak not detected	Yes, leaks DNS	🟩	🟩	🔴	🔴	🟩	🔴
9	Azire VPN	🟩	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🔴	▲
10	BestVPN	🔴	Leak not detected	Leak not detected	🟩	🔴	🟩	🔴	🟩	🔴
11	Betternet	🔴	Leak not detected	Yes, leaks DNS	🔴	🔴	🔴	🔴	🔴	🟩
12	BolehVPN	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🔴	🔴
13	Bullguard	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🔴	🟩
14	Cactus VPN	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🟩	🔴
15	Cryptostorm	🔴	Leak not detected	Leak not detected	🔴	🔴	🔴	🔴	🔴	🔴
16	CyberGhost	🔴	Leak not detected	Leak not detected	🔴	🟩	🟩	🔴	🔴	🔴
17	Encrypt.me	🔴	Leaks traffic**	Yes, leaks DNS**	🟩	🔴	🔴	🔴	🟩	🟩
18	ExpressVPN	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🟩	🔴	▲
19	F-Secure Freedome	🟩	Leak not detected	Leak not detected	🟩	🟩	🟩	🔴	🔴	🟩
20	FastestVPN	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	▲
21	Free VPN by Free VPN.org	🔴	Leaks traffic	Yes, leaks DNS	🔴	🔴	🔴	🔴	🟩	🔴
22	Goose VPN	🔴	Leak not detected	Leak not detected	🟩	🟩	🟩	🔴	🔴	🔴
23	Hide My Ass!	🔴	Leaks traffic	Yes, leaks DNS**	🔴	🔴	🟩	🔴	🔴	🔴
24	Hide.me	🟩	Leak not detected	Leak not detected	🟩	🔴	🔴	🟩	🔴	▲
25	HideIPVPN	🔴	Leak not detected	Leak not detected	🟩	🔴	🟩	🔴	🟩	🔴
26	Hotspot Shield	🟩	Leak not detected	Yes, leaks DNS**	🔴	🔴	🟩	🔴	🔴	🟩
27	IP Vanish	🔴	Leaks traffic**	Yes, leaks DNS**	🔴	🔴	🟩	🟩	🔴	🟩
28	IVPN	🟩(in custom)	Leak not detected	Leak not detected	🟩	🔴	🟩	🔴	🔴	🔴
29	Ivacy VPN	🔴	Leaks traffic**	Yes, leaks DNS**	🟩	🔴	🟩	🔴	🔴	🔴
30	K2VPN	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🟩	🔴
31	Kaspersky	🔴	Leak not detected	Leak not detected	▲	🟩	🔴	🔴	🟩	🔴
32	KeepSolid VPN Unlimited	🔴	Leak not detected	Leak not detected	🔴	🔴	🔴	🔴	🔴	🔴
33	LeVPN	🔴	Leak not detected	Leak not detected	🟩	🔴	🟩	🟩	🟩	🔴
34	Mozilla VPN	🟩	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🟩
35	Mullvad VPN	🟩(in custom)	Leak not detected	Leak not detected	🟩	🔴	🟩	🔴	🔴	▲
36	Namecheap	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🟩	🔴	🟩
37	NordVPN	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🔴	🔴
38	Norton Secure VPN	Leaks to ISP	Leaks traffic**	Yes, leaks DNS**	🔴	🔴	🔴	🔴	🔴	🔴
39	OVPN	🟩	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🔴	🔴
40	OpenVPN Access Server	🔴	Leaks traffic	Yes, leaks DNS	🔴	🔴	🟩	🔴	🔴	🔴

			Leaks during tunnel failure		Security & Privacy Essentials					
#	VPN Provider	IPv6 Support	All Data Leak	DNS Leak	DNSSEC Validation	Qmin Support	RPKI Validation	DoH Disabled?	Does it use Public DNS?	DNS Proxy
41	Outline	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🟩	🔴
42	Panda VPN	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🟩	🔴
43	Perfect Privacy	🟩	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🔴
44	Private Internet Access	🔴	Leak not detected	Yes, leaks DNS**	🔴	🟩	🟩	🔴	🔴	▲
45	Private Tunnel	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🟩
46	Private VPN	🔴	Leak not detected	Leak not detected	🔴	🟩	🟩	🔴	🔴	🔴
47	Proton VPN	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🟩	🔴	🔴
48	Psiphon	🔴	Leaks traffic	Yes, leaks DNS	🟩	🔴	🟩	🔴	🟩	▲
49	Pure VPN	🔴	Leaks traffic	Leak not detected	🟩	🔴	🔴	🔴	🔴	🔴
50	Riseup	🔴	Leak not detected	Leak not detected	🟩	▲	🟩	🔴	🔴	▲
51	Speedify	🔴	Leaks traffic	Yes, leaks DNS	🟩	🟩	🟩	🔴	🟩	🔴
52	Star VPN	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🟩	🔴
53	Steganos	🔴	Leak not detected	Leak not detected	🟩	🔴	🟩	🔴	🟩	🔴
54	Streisand	🔴	Leak not detected	Yes, leaks DNS	🟩	🟩	🔴	🟩	🟩	🔴
55	Strong VPN	🔴	Leaks traffic**	Yes, leaks DNS**	🟩	🔴	🟩	🟩	🔴	🔴
56	SurfEasy	Leaks to ISP	Leaks traffic	Yes, leaks DNS	🔴	🔴	🟩	🔴	🔴	🔴
57	SurfShark	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🔴	🔴	🔴
58	TorGuard	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🔴
59	Touch VPN	🔴	Leak not detected	Leak not detected	🔴	🟩	🟩	🔴	🟩	🔴
60	Trust.Zone	🔴	Leaks traffic	Yes, leaks DNS	🟩	🔴	🟩	🔴	🟩	🔴
61	TunnelBear	🔴	Leak not detected	Yes, leaks DNS**	🔴	🔴	🟩	🔴	🔴	🟩
62	Turbo VPN	Leaks to ISP	Leaks traffic	Yes, leaks DNS	🟩	🔴	🔴	🔴	🟩	🔴
63	University VPN	Leaks to ISP	Leaks traffic	Yes, leaks DNS	🟩	🔴	🟩	🟩	🔴	🔴
64	Unspyable	🔴	Leak not detected	Leak not detected	🔴	🔴	🟩	🟩	🔴	▲
65	Urban VPN Desktop	🔴	Leaks traffic	Yes, leaks DNS	🔴	🔴	🟩	🔴	🔴	🟩
66	VPN Hotspot - Unlimited Proxy	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🟩	🔴
67	VPN Owl	🔴	Leak not detected	Yes, leaks DNS	🔴	🔴	🔴	🔴	🟩	🔴
68	VPN Plus	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🟩	🔴
69	VPN Pro	🔴	Leak not detected	Leak not detected	🟩	🔴	🟩	🔴	🟩	🔴
70	VPN Proxy Master	🔴	Leaks traffic	Yes, leaks DNS	🟩	🔴	🟩	🔴	🟩	🔴
71	VPN Super: Best VPN Proxy	🔴	Leak not detected	Leak not detected	🟩	🔴	🟩	🔴	🟩	🔴
72	VPN.ac	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🟩
73	VPNBook	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🔴
74	VPNLite	🔴	Leak not detected	Leak not detected	🔴	🔴	🔴	🔴	🔴	🔴
75	VPNUK	🔴	Leak not detected	Leak not detected	🔴	🔴	🔴	🔴	🔴	🔴
76	VeePN	🔴	Leak not detected	Leak not detected	🔴	🟩	🔴	🔴	🔴	🔴
77	Vypr	🔴	Leak not detected	Leak not detected	🟩	🔴	🔴	🔴	🔴	🔴
78	Windscribe	🔴	Leak not detected	Leak not detected	🟩	🔴	🟩	🟩	🟩	🔴
79	ZenMate	🔴	Leak not detected	Leak not detected	🔴	🔴	🟩	🔴	🔴	🔴
80	ZoogVPN	🔴	Leak not detected	Leak not detected	🟩	🟩	🔴	🟩	🟩	🔴

Table C.2: **VPN Providers & Results**—A red circle indicates “false”, and a green square indicates “true” for each test mentioned in the column. A yellow triangle denotes an inconclusive result. Under leaks during tunnel failure, findings also observed in the secure mode (§4.5.8) are marked with two asterisks**. For RPKI validation, “true” implies at least one experiment of the provider shows evidence that RPKI validation is enabled.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Can I use WebSocket? https://caniuse.com/?search=websocket.

[2] The WebSocket API (WebSockets). https://developer.mozilla.org/en-US/docs/Web/API/WebSockets_API.

[3] BAT Ecosystem Growth, 2023. https://basicattentiontoken.org/growth/.

[4] Tami Abdollah. Study: Russia's web-censoring tool sets pace for imitators. https://apnews.com/article/internet-service-providers-politics-iran-china-technology-2cee9a8f8b234f5a86987eec835f3c55.

[5] Giuseppe Aceto and Antonio Pescapé. Internet censorship detection: A survey. *Computer Networks*, 83, 2015.

[6] Yaman Akdeniz. Internet content regulation: UK government and the control of Internet content. *Computer Law & Security Review*, 2001.

[7] Omer Akgul, Richard Roberts, Moses Namara, Dave Levin, and Michelle L Mazurek. Investigating Influencer VPN Ads on YouTube. In *IEEE Symposium on Security and Privacy*, 2022.

[8] Alexander Khinshtein: Verification of users on the Internet is a matter of time, 2021. https://tass.ru/interviews/11032409.

[9] Darren Allan. CyberGhost owner buys PIA for $95.5m to create VPN giant. TechRadar, 2019. https://www.techradar.com/news/cyberghost-owner-buys-pia-for-dollar955m-to-create-vpn-giant.

[10] ARIN. Resource Certification (RPKI), 2021. https://www.arin.net/resources/manage/rpki/.

[11] ARIN. Route Origin Authorizations (ROAs), 2021. https://www.arin.net/resources/manage/rpki/roa_request.

[12] Simurgh Aryan, Homa Aryan, and J Alex Halderman. Internet Censorship in Iran: A First Look. In *FOCI*, 2013.

[13] Shelly Banjo. VPN Downloads Surge in Response to Hong Kong Security Law. Bloomberg, 2020. https://www.bloomberg.com/news/articles/2020-05-22/vpn-downloads-surge-in-response-to-hong-kong-security-law.

[14] Shehar Bano, Philipp Richter, Mobin Javed, Srikanth Sundaresan, Zakir Durumeric, Steven J. Murdoch, Richard Mortier, and Vern Paxson. Scanning the internet for liveness. *ACM CCR*, 48(2), 2018.

[15] Beautiful soup documentation. https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

[16] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal statistical society: series B (Methodological)*, 1995.

[17] Veroniek Binkhorst, Tobias Fiebig, Katharina Krombholz, Wolter Pieters, et al. Security at the end of the tunnel: The anatomy of VPN mental models among experts and non-experts in a corporate context. In *31st USENIX Security Symposium (USENIX Security 22)*, 2022.

[18] David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003.

[19] Что делать если сайт заблокирован провайдером. http://blogsisadmina.ru/internet/chto-delat-esli-sajt-zablokirovan-provajderom.html, November 2015.

[20] Aria Bracci and Lia Petronio. New research shows that, post net neutrality, internet providers are slowing down your streaming. https://news.northeastern.edu/2018/09/10/new-research-shows-your-internet-provider-is-in-control/.

[21] Virginia Braun and Victoria Clarke. Using thematic analysis in psychology. *Qualitative research in psychology*, 2006.

[22] Thanh Bui, Siddharth Rao, Markku Antikainen, and Tuomas Aura. Client-Side Vulnerabilities in Commercial VPNs. In *Nordic Conference on Secure IT Systems*. Springer, 2019.

[23] Censored Planet. https://censoredplanet.org/.

[24] Abdelberi Chaabane, Terence Chen, Mathieu Cunche, Emiliano De Cristofaro, Arik Friedman, and Mohamed Ali Kaafar. Censorship in the wild: Analyzing Internet filtering in Syria. In *Proceedings of the 2014 Conference on Internet Measurement Conference*, 2014.

[25] Farah Chanchary and Sonia Chiasson. User perceptions of sharing, advertising, and tracking. In *Eleventh Symposium On Usable Privacy and Security (SOUPS)*, 2015.

[26] Caleb Chen. America has killed Net Neutrality; Why you need a VPN service. Private Internet Access, 2018. https://www.privateinternetaccess.com/blog/america-has-killed-net-neutrality-why-you-need-a-vpn-service/.

[27] CDN providers blocked by China. https://www.cdnfinder.com/cdn-providers-blocked-china, 2014.

[28] Catalin Cimpanu. Data breach at Russian ISP impacts 8.7 million customers. ZDNet, 2019. https://www.zdnet.com/article/data-breach-at-russian-isp-impacts-8-7-million-customers/.

[29] Richard Clayton, Steven J Murdoch, and Robert NM Watson. Ignoring the great firewall of China. In *International Workshop on Privacy Enhancing Technologies*, 2006.

[30] Cloudflare. The Internet is Getting Safer: Fall 2020 RPKI Update, 2020. https://blog.cloudflare.com/rpki-2020-fall-update.

[31] Cloudflare. Configuring Cloudflare IP Geolocation, 2021. https://web.archive.org/web/20210816060903/https://support.cloudflare.com/hc/en-us/articles/200168236-Configuring-Cloudflare-IP-Geolocation.

[32] Cloudflare. How VPNs affect Internet speed, 2021. https://www.cloudflare.com/learning/access-management/vpn-speed.

[33] Jacob Cohen. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 1960.

[34] Consumer Reports. Security Planner, 2021. https://securityplanner.consumerreports.org/.

[35] Gareth Corfield. NordVPN rapped by ad watchdog over insecure public Wi-Fi claims. The Register, 2019. https://www.theregister.com/2019/05/01/nordvpn_tv_ad_rapped_advertising_standards_authority/.

[36] Jedidiah R Crandall, Daniel Zinn, Michael Byrd, Earl T Barr, and Rich East. ConceptDoppler: a weather tracker for internet censorship. In *ACM Conference on Computer and Communications Security*, 2007.

[37] CVE-2019-12103. https://nvd.nist.gov/vuln/detail/CVE-2019-12103.

[38] CVE-2019-19356. https://nvd.nist.gov/vuln/detail/CVE-2019-19356.

[39] Abdi Latif Dahir. Internet shutdowns are costing African governments more than we thought. https://qz.com/1089749/internet-shutdowns-are-increasingly-taking-a-toll-on-africas-economies/, 2017.

[40] Sarkis Darbinyan and Sergei Hovyadinov. World Intermediary Liability Map: Russia. https://wilmap.law.stanford.edu/country/russia.

[41] Wouter B de Vries, Quirin Scheitle, Moritz Müller, Willem Toorop, Ralph Dolmans, and Roland van Rijswijk-Deij. A First Look at QNAME Minimization in the Domain Name System. In *International Conference on Passive and Active Network Measurement*. Springer, 2019.

[42] Lucas DiCioccio, Renata Teixeira, and Catherine Rosenberg. Measuring home networks with homenet profiler. In *International Conference on Passive and Active Network Measurement*. Springer, 2013.

[43] Hao Ding and Michael Rabinovich. TCP stretch acknowledgements and timestamps: Findings and implications for passive RTT measurement. *ACM CCR*, 45(3), 2015.

[44] D. Dittrich and E. Kenneally. The Menlo Report: Ethical principles guiding information and communication technology research. Technical report, U.S. Department of Homeland Security, 2012.

[45] Jason A Donenfeld. Wireguard: Next generation kernel network tunnel. In *Proceedings of the Network and Distributed System Security Symposium (NDSS)*, 2017.

[46] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J Alex Halderman. A search engine backed by Internet-wide scanning. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015.

[47] Zakir Durumeric, Eric Wustrow, and J Alex Halderman. ZMap: Fast Internet-wide Scanning and Its Security Applications. In *USENIX Security Symposium*, volume 8, 2013.

[48] Agnieszka Dutkowska-Żuk, Austin Hounsel, Amy Morrill, Andre Xiong, Marshini Chetty, and Nick Feamster. How and why people use virtual private networks. In *31st USENIX Security Symposium (USENIX Security 22)*, 2022.

[49] Agnieszka Dutkowska-Zuk, Austin Hounsel, Andre Xiong, Molly Roberts, Brandon Stewart, Marshini Chetty, and Nick Feamster. Practicing Safe Browsing: Understanding How and Why University Students Use Virtual Private Networks. *arXiv e-prints*, 2020.

[50] Madeline Earp. Laws, cheap web filters arm Russia to block news, says Censored Planet. `https://cpj.org/2019/11/russia-internet-censorship-censored-planet/`.

[51] Electron. Build cross-platform desktop apps with JavaScript, HTML, and CSS, 2021. `https://www.electronjs.org/`.

[52] Steven Englehardt and Arvind Narayanan. Online tracking: A 1-million-site measurement and analysis. In *Proceedings of the 2016 ACM SIGSAC conference on computer and communications security*, 2016.

[53] Roya Ensafi, Jeffrey Knockel, Geoffrey Alexander, and Jedidiah R Crandall. Detecting intentional packet drops on the Internet via TCP/IP side channels. In *International Conference on Passive and Active Network Measurement*. Springer, 2014.

[54] Roya Ensafi, Philipp Winter, Abdullah Mueen, and Jedidiah R Crandall. Analyzing the Great Firewall of China over space and time. *Proceedings on privacy enhancing technologies*, 2015.

[55] Rep. Eshoo. Rep. Eshoo and Senator Wyden Urge FTC to Address Deceptive Data Practices by VPN Providers. eshoo.house.gov, 2022. `https://eshoo.house.gov/media/press-releases/rep-eshoo-and-senator-wyden-urge-ftc-address-deceptive-data-practices-vpn`.

[56] Lookman Fazal, Sachin Ganu, Martin Kappes, Anjur Sundaresan Krishnakumar, and Parameshwaran Krishnan. Tackling security vulnerabilities in VPN-based wireless deployments. In *Proceedings of IEEE International Conference on Communications (ICC)*, 2004.

[57] David Fifield, Jiang Jian, and Paul Pearce. SNI proxies. Bamsoftware, 2016. `https://www.bamsoftware.com/computers/sniproxy/index.html`.

[58] FortiNet. Fortiguard labs web filter. `https://fortiguard.com/webfilter`.

[59] Sergey Frolov and Eric Wustrow. The use of TLS in censorship circumvention. In *Network and Distributed Systems Symposium (NDSS)*, 2019.

[60] Genevieve Gebhart and Tadayoshi Kohno. Internet Censorship in Thailand: User Practices and Potential Threats. In *Security and Privacy (EuroS&P), 2017 IEEE European Symposium on*, 2017.

[61] Gensim Library — gensim 3.8.1. `https://pypi.org/project/gensim/`.

[62] David Gewirtz. Best VPN services for 2021: Safe and fast don't come for free. ZDNet, 2021. `https://www.zdnet.com/article/best-vpn-services-for-2021-safe-and-fast-dont-come-for-free`.

[63] Manaf Gharaibeh, Anant Shah, Bradley Huffaker, Han Zhang, Roya Ensafi, and Christos Papadopoulos. A look at router geolocation in public and commercial databases. In *Proceedings of the 2017 Internet Measurement Conference*. ACM, 2017.

[64] Phillipa Gill, Masashi Crete-Nishihata, Jakub Dalek, Sharon Goldberg, Adam Senft, and Greg Wiseman. Characterizing web censorship worldwide: Another look at the opennet initiative data. *ACM Transactions on the Web (TWEB)*, 9(1), 2015.

[65] Glasnost: Results from tests for bittorrent traffic shaping. `https://broadband.mpi-sws.org/transparency/results/`.

[66] Google. Google Cloud: Cloud Computing Services, 2021. https://cloud.google.com.

[67] Google. Google IPv6 Statistics, 2021. https://www.google.com/intl/en/ipv6/statistics.html.

[68] Gosuslugi. Получите электронный сертификат безопасности. (Translation) Get an electronic security certificate. https://www.gosuslugi.ru/tls.

[69] Yael Grauer. The Best VPN Service. New York Times Wirecutter, 2020. https://www.nytimes.com/wirecutter/reviews/best-vpn-service/.

[70] Yael Grauer. Should You Use a VPN? In *Consumer Reports Digital Lab*, 2021. https://www.consumerreports.org/vpn-services/should-you-use-a-vpn-a5562069524/.

[71] Yael Grauer. VPN Testing Reveals Poor Privacy and Security Practices, Hyperbolic Claims. In *Consumer Reports Digital Lab*, 2021. https://www.consumerreports.org/vpn-services/vpn-testing-poor-privacy-security-hyperbolic-claims-a1103787639.

[72] Yael Grauer and Steve Blair. Security and Privacy of VPNs Running on Windows 10. In *Consumer Reports Digital Lab*, 2021. https://digital-lab-wp.consumerreports.org/wp-content/uploads/2021/12/VPN-White-Paper.pdf.

[73] We monitor and challenge internet censorship in China. https://en.greatfire.org.

[74] Ariel Hochstadt. The death of net neutrality and the rise of VPNs. TechRadar, 2018. https://www.techradar.com/news/the-death-of-net-neutrality-and-the-rise-of-vpns.

[75] Rae Hodge. VPN use surges during the coronavirus lockdown, but so do security risks. CNET, 2020. https://www.cnet.com/news/vpn-use-surges-during-the-coronavirus-lockdown-but-so-do-security-risks/.

[76] Hans Hoogstraaten. Evaluating server-side internet proxy detection methods. Master's thesis, Leiden University, 2018.

[77] Nicholas Hopper, Eugene Y. Vasserman, and Eric Chan-Tin. How much anonymity does network latency leak? *ACM Transactions on Information and System Security*, 13(2), 2010.

[78] Michael Horowitz. RouterSecurity.org - TCP Ports to Test. https://routersecurity.org/testrouter.php#TCPports.

[79] Austin Hounsel, Prateek Mittal, and Nick Feamster. Automatically Generating a Large, Culture-Specific Blocklist for China. In *8th USENIX Workshop on Free and Open Communications on the Internet (FOCI 18)*. USENIX Association, 2018.

[80] Danny Yuxing Huang, Noah Apthorpe, Frank Li, Gunes Acar, and Nick Feamster. IoT Inspector: Crowdsourcing Labeled Network Traffic from Smart Home Devices at Scale. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, 2020.

[81] Tim Hume. How FBI caught Ross Ulbricht, alleged creator of criminal marketplace Silk Road. CNN, 2013. `https://www.cnn.com/2013/10/04/world/americas/silk-road-ross-ulbricht/index.html`.

[82] Toke Høiland-Jørgensen, Bengt Ahlgren, Per Hurtig, and Anna Brunstrom. Measuring latency variation in the internet. In *CoNEXT*. ACM, 2016.

[83] i2Coalition. The VPN Trust Initiative ("VTI"). `https://vpntrust.net/`, September 2020.

[84] IETF. RFC 792: Internet Control Message Protocol, 1981. `https://datatracker.ietf.org/doc/html/rfc792`.

[85] IETF. RFC 1812: Requirements for IP Version 4 Routers, 1995. `https://datatracker.ietf.org/doc/html/rfc1812`.

[86] IETF. RFC 4033: DNS Security Introduction and Requirements, 2005. `https://datatracker.ietf.org/doc/html/rfc4033`.

[87] IETF. RFC 7050: Discovery of the IPv6 Prefix Used for IPv6 Address Synthesis, 2013. `https://datatracker.ietf.org/doc/html/rfc7050`.

[88] IETF. DNS Query Name Minimisation to Improve Privacy, 2016. `https://tools.ietf.org/html/rfc7816`.

[89] Muhammad Ikram, Narseo Vallina-Rodriguez, Suranga Seneviratne, Mohamed Ali Kaafar, and Vern Paxson. An analysis of the privacy and security risks of android VPN permission-enabled apps. In *Proceedings of the 2016 Internet Measurement Conference*, 2016.

[90] Indonesia introduces new internet censorship system. `https://www.arabnews.com/node/1218011/world`.

[91] Internet Freedom Festival. Announcing the VPN Village!, 2020. `https://internetfreedomfestival.org/vpn-village/`.

[92] Internet usage statistics. `https://www.internetworldstats.com/stats.htm`.

[93] Internet Outage Detection and Analysis, CAIDA. `https://ioda.caida.org/ioda/dashboard`.

[94] Ivacy VPN. Ivacy VPN - Stay Anonymous Online with a VPN! `https://web.archive.org/web/20210716224231/https://www.ivacy.com/what-is-vpn/anonymous-vpn/`.

[95] Hao Jiang and Constantinos Dovrolis. Passive estimation of TCP round-trip times. *ACM CCR*, 32(3), 2002.

[96] Jigsaw. Outline VPN. https://getoutline.org.

[97] Martin Johnson. China, GitHub and the man-in-the-middle. GreatFire.org, January 30, 2013. https://en.greatfire.org/blog/2013/jan/china-github-and-man-middle.

[98] Ben Jones, Tzu-Wen Lee, Nick Feamster, and Phillipa Gill. Automated detection and fingerprinting of censorship block pages. In *Internet Measurement Conference (IMC)*. ACM, 2014.

[99] Jakob Kastrenakes. NordVPN reveals server breach that could have let attacker monitor traffic. The Verge, 2019. https://www.theverge.com/2019/10/21/20925065/nordvpn-server-breach-vpn-traffic-exposed-encryption.

[100] Ethan Katz-Bassett, John P John, Arvind Krishnamurthy, David Wetherall, Thomas Anderson, and Yatin Chawathe. Towards IP geolocation using delay and topology measurements. In *IMC*. ACM, 2006.

[101] Charlie Kaufman et al. Internet key exchange (IKEv2) protocol. Technical report, RFC 4306, 2005.

[102] David Kaye. Online propaganda, censorship and human rights in Russia's war against reality. *American Journal of International Law*, 2022. https://www.cambridge.org/core/journals/american-journal-of-international-law/article/online-propaganda-censorship-and-human-rights-in-russias-war-against-reality/359EF362F588AC8F601FE6C28260AD83.

[103] Mohammad Taha Khan, Joe DeBlasio, Geoffrey M Voelker, Alex C Snoeren, Chris Kanich, and Narseo Vallina-Rodriguez. An empirical analysis of the commercial VPN ecosystem. In *Proceedings of the Internet Measurement Conference (IMC)*, 2018.

[104] Taha Khan. VPN Tests. GitHub, 2018. https://github.com/tahakhan5/vpn_tests/blob/master/leakage_tests/tunnel_failure/run_test.py#L60.

[105] Sheharbano Khattak, Mobin Javed, Philip D Anderson, and Vern Paxson. Towards Illuminating a Censorship Monitor's Model to Facilitate Evasion. In *FOCI*, 2013.

[106] Aleksandar Kochovski. The Top 25 VPN Statistics, Facts & Trends for 2021. Cloudwards, 2021. https://www.cloudwards.net/vpn-statistics/.

[107] Christian Kreibich, Nicholas Weaver, Gregor Maier, Boris Nechaev, and Vern Paxson. Experiences from Netalyzr with engaging users in end-system measurement. In *Proceedings of the first ACM SIGCOMM workshop on Measurements up the stack*, 2011.

[108] Christian Kreibich, Nicholas Weaver, Boris Nechaev, and Vern Paxson. Netalyzr: Illuminating the edge network. In *Proceedings of the 2010 ACM SIGCOMM Conference on Internet Measurement*, 2010.

[109] Citizen Lab and Others. Url testing lists intended for discovering website censorship, 2014. https://github.com/citizenlab/test-lists.

[110] Lai Yi Ohlsen, Matt Mathis, Stephen Soltesz, Simone Basso. Introducing ndt7, 2020. `https://www.measurementlab.net/blog/ndt7-introduction/`.

[111] J Richard Landis and Gary G Koch. The measurement of observer agreement for categorical data. *Biometrics*, 1977.

[112] Langdetect Library — langdetect 1.0.7. `https://pypi.org/project/langdetect/`.

[113] Jonathan Lazar, Jinjuan Heidi Feng, and Harry Hochheiser. *Research methods in human-computer interaction*. Morgan Kaufmann, 2017.

[114] Fangfan Li, Arian Akhavan Niaki, David Choffnes, Phillipa Gill, and Alan Mislove. A large-scale analysis of deployed traffic differentiation practices. In *Proceedings of the ACM Special Interest Group on Data Communication*. SIGCOMM, 2019.

[115] Rensis Likert. A technique for the measurement of attitudes. *Archives of psychology*, 1932.

[116] Ruei-Min Lin, Yi-Chun Chou, and Kuan-Ta Chen. Stepping stone detection at the server side. In *Security in Computers, Networking and Communications*. IEEE, 2011.

[117] Graham Lowe, Patrick Winters, and Michael L Marcus. The great DNS wall of China. *MS, New York University*, 21, 2007.

[118] Lumen Privacy Monitor. `https://www.icsi.berkeley.edu/icsi/projects/networking/haystack`.

[119] Aniss Maghsoudlou, Lukas Vermeulen, Ingmar Poese, and Oliver Gasser. Characterizing the VPN ecosystem in the wild. In *PAM*. Springer, 2023.

[120] Gregor Maier, Anja Feldmann, Vern Paxson, and Mark Allman. On dominant characteristics of residential broadband internet traffic. In *IMC*. ACM, 2009.

[121] Measuring Active Listeners, Connection Observers, and Legitimate Monitors, Cloudflare. `https://malcolm.cloudflare.com/`.

[122] Massimiliano Marcon, Marcel Dischinger, Krishna P. Gummadi, and Amin Vahdat. The local and global effects of traffic shaping in the internet. *2011 Third International Conference on Communication Systems and Networks (COMSNETS 2011)*, 2011.

[123] Bill Marczak, Nicholas Weaver, Jakub Dalek, Roya Ensafi, David Fifield, Sarah McKune, Arn Rey, John Scott-Railton, Ron Deibert, and Vern Paxson. An Analysis of China's Great Cannon. In *Free and Open Communications on the Internet*. USENIX, 2015.

[124] Nathalie Marechal. From Russia With Crypto: A Political History of Telegram. In *8th USENIX Workshop on Free and Open Communications on the Internet (FOCI 18)*. USENIX Association, 2018.

[125] Mark Smirniotis. What Is a VPN and What Can (and Can't) It Do? New York Times Wirecutter, 2021. `https://www.nytimes.com/wirecutter/guides/what-is-a-vpn/`.

[126] Arunesh Mathur, Gunes Acar, Michael J Friedman, Elena Lucherini, Jonathan Mayer, Marshini Chetty, and Arvind Narayanan. Dark patterns at scale: Findings from a crawl of 11k shopping websites. *Proceedings of the ACM on Human-Computer Interaction*, (CSCW), 2019.

[127] PC Matic. VPN: A Decade's Worth of Growth, 2020. `https://www.pcmatic.com/news/vpn_report/`.

[128] Matthew Mpoke Bigg. A history of the tensions between Ukraine and Russia. the New York Times, 2022. `https://www.nytimes.com/2022/03/26/world/europe/ukraine-russia-tensions-timeline.html`.

[129] Eddy Max. The Best VPN Services for 2021. PCMag, 2021. `https://www.pcmag.com/picks/the-best-vpn-services`.

[130] MaxMind. `https://www.maxmind.com/`.

[131] Peter Mayer, Yixin Zou, Florian Schaub, and Adam J Aviv. "Now I'm a bit angry:" Individuals' Awareness, Perception, and Responses to Data Breaches that Affected Them. In *Proceedings of the USENIX Security Symposium (USENIX Security)*, 2021.

[132] Niall McCarthy. VPN Usage Surges During COVID-19 Crisis. Forbes, 2020. `https://www.forbes.com/sites/niallmccarthy/2020/03/17/vpn-usage-surges-during-covid-19-crisis-infographic/?sh=5fe8e5157d79`.

[133] Allison McDonald, Matthew Bernhard, Luke Valenta, Benjamin VanderSloot, Will Scott, Nick Sullivan, J Alex Halderman, and Roya Ensafi. 403 Forbidden: A global view of CDN geoblocking. In *Proceedings of the Internet Measurement Conference (IMC)*, 2018.

[134] Nora McDonald, Sarita Schoenebeck, and Andrea Forte. Reliability and inter-rater reliability in qualitative research: Norms and guidelines for cscw and hci practice. *Proceedings of the ACM on Human-Computer Interaction*, (CSCW), 2019.

[135] Mary L McHugh. Interrater reliability: the kappa statistic. *Biochemia medica*, 2012.

[136] William Melicher, Mahmood Sharif, Joshua Tan, Lujo Bauer, Mihai Christodorescu, and Pedro Giovanni Leon. Preferences for web tracking. *Proceedings on Privacy Enhancing Technologies (PoPETS)*, 2016.

[137] Xianghang Mi, Xuan Feng, Xiaojing Liao, Baojun Liu, XiaoFeng Wang, Feng Qian, Zhou Li, Sumayah Alrwais, Limin Sun, and Ying Liu. Resident evil: Understanding residential ip proxy as a dark service. In *Security & Privacy*. IEEE, 2019.

[138] Microsoft. Microsoft Leads Initiative for Virtual Private Networks Across the Internet, 1996. https://news.microsoft.com/1996/03/04/microsoft-leads-initiative-for-virtual-private-networks-across-the-internet/.

[139] Matthew B Miles, A Michael Huberman, Johnny Saldana, et al. Qualitative data analysis: A methods sourcebook. *Thousand Oaks, CA: Sage*, 2014.

[140] Miles Kenyon. Citizen Lab Summer Institute 2019. Citizen Lab, 2019. https://citizenlab.ca/2019/02/citizen-lab-summer-institute-2019/.

[141] Measurement Lab. https://www.measurementlab.net/tests/ndt/.

[142] Arash Molavi Kakhki, Abbas Razaghpanah, Rajesh Golani, David Choffnes, Phillipa Gill, and Alan Mislove. Identifying traffic differentiation on cellular data networks. In *Proceedings of the 2014 ACM Conference on SIGCOMM*, SIGCOMM '14, 2014.

[143] Arash Molavi Kakhki, Abbas Razaghpanah, Anke Li, Hyungjoon Koo, Rajesh Golani, David Choffnes, Phillipa Gill, and Alan Mislove. Identifying traffic differentiation in mobile networks. In *Proceedings of the 2015 Internet Measurement Conference*, 2015.

[144] Mozilla. What's next in making Encrypted DNS-over-HTTPS the Default. Mozilla Blog, 2019. https://blog.mozilla.org/futurereleases/2019/09/06/whats-next-in-making-dns-over-https-the-default/.

[145] Mozilla. Canary domain - use-application-dns.net, 2021. https://support.mozilla.org/en-US/kb/canary-domain-use-application-dnsnet.

[146] Mozilla. Can't networks just trigger the canary domain check all the time and disable DoH? Mozilla Support, 2021. https://support.mozilla.org/en-US/kb/dns-over-https-doh-faqs#w_cant-networks-just-trigger-the-canary-domain-check-all-the-time-and-disable-doh.

[147] Mozilla. Mozilla VPN, 2021. https://web.archive.org/web/20210531041656/https://www.mozilla.org/en-US/products/vpn/.

[148] Multiple authors. MITM in Russia, March 2022. https://bugzilla.mozilla.org/show_bug.cgi?id=1758773.

[149] Zubair Nabi. The Anatomy of Web Censorship in Pakistan. In *FOCI*, 2013.

[150] Moses Namara, Daricia Wilkinson, Kelly Caine, and Bart P Knijnenburg. Emotional and Practical Considerations Towards the Adoption and Abandonment of VPNs as a Privacy-Enhancing Technology. *Proceedings on Privacy Enhancing Technologies (PoPETS)*, 2020.

[151] Namecheap. Google Trends Reveals Surge in Demand for VPN. https://www.namecheap.com/blog/vpn-surge-in-demand/, 2020.

[152] National Commission for the Protection of Human Subjects of Biomedical and Behavioral Research. The Belmont Report: Ethical Principles and Guidelines for the Protection of Human Subjects of Research, 1978.

[153] Natural Language Toolkit. https://www.nltk.org/.

[154] Net Neutrality in the United States. https://en.wikipedia.org/wiki/Net_neutrality_in_the_United_States.

[155] Neustar. Neustar Announces Acquisition of Verisign's Public DNS Service, 2020. https://www.home.neustar/about-us/news-room/press-releases/2020/neustar-announces-acquisition-of-verisigns-public-dns-service.

[156] Alfred Ng. How Private Is My VPN? The Markup, 2021. https://themarkup.org/ask-the-markup/2021/08/12/how-private-is-my-vpn.

[157] Aqib Nisar, Aqsa Kashaf, Ihsan Ayyub Qazi, and Zartash Afzal Uzmi. Incentivizing censorship measurements via circumvention. In *Proceedings of the 2018 Conference of the ACM Special Interest Group on Data Communication, SIGCOMM 2018, Budapest, Hungary, August 20-25, 2018*.

[158] Nmap Project. Npcap: Packet capture library for Windows. https://nmap.org/npcap.

[159] NortonLifeLock Product and Services. https://web.archive.org/web/20210719021634/https://www.nortonlifelock.com/us/en/privacy/product-privacy-notices/.

[160] Open Observatory of Network Interference. https://ooni.torproject.org/.

[161] OONI. The test list methodology. https://ooni.torproject.org/get-involved/contribute-test-lists/.

[162] OONI. OONI Data Policy, 2020. https://ooni.org/about/data-policy.

[163] Open Technology Fund. Open Technology Fund, 2021. https://www.opentech.fund/.

[164] OpenVPN. A Modern Private Network, Built for the Cloud, 2021. https://openvpn.net/.

[165] OpenVPN. Reference manual for OpenVPN 2.4, 2021. https://openvpn.net/community-resources/reference-manual-for-openvpn-2-4/.

[166] OpenVPN Inc. Official OpenVPN Connect client software. https://openvpn.net/vpn-client/.

[167] OpenVPN Inc. OpenVPN Access Server. https://aws.amazon.com/marketplace/pp/prodview-fiozs66safl5a.

[168] Sunoo Park and Kendra Albert. A Researcher's Guide to Some Legal Risks of Security Research, 2020. https://clinic.cyber.harvard.edu/files/2020/10/Security_Researchers_Guide-2.pdf.

[169] Paul Pearce, Roya Ensafi, Frank Li, Nick Feamster, and Vern Paxson. Augur: Internet-wide detection of connectivity disruptions. In *38th IEEE Symposium on Security and Privacy*, May 2017.

[170] Paul Pearce, Ben Jones, Frank Li, Roya Ensafi, Nick Feamster, Nick Weaver, and Vern Paxson. Global measurement of DNS manipulation. In *USENIX Security Symposium*, 2017.

[171] Cristel Pelsser, Luca Cittadini, Stefano Vissicchio, and Randy Bush. From Paris to Tokyo: On the suitability of ping to measure latency. In *IMC*. ACM, 2013.

[172] Nicole Perlroth and David Sanger. North Korea Loses Its Link to the Internet. https://www.nytimes.com/2014/12/23/world/asia/attack-is-suspected-as-north-korean-internet-collapses.html?_r=0.

[173] Vasile C Perta, Marco V Barbera, Gareth Tyson, Hamed Haddadi, and Alessandro Mei. A glance through the VPN looking glass: IPv6 leakage and DNS hijacking in commercial VPN clients. *Proceedings on Privacy Enhancing Technologies (PoPETS)*, 2015.

[174] Ingmar Poese, Steve Uhlig, Mohamed Ali Kaafar, Benoit Donnet, and Bamba Gueye. IP geolocation databases: Unreliable? *ACM SIGCOMM Computer Communication Review*, 2011.

[175] Portuguese ISPs given 40 days to comply with EU net neutrality rules. https://edri.org/portuguese-isps-given-40-days-to-comply-with-eu-net-neutrality-rules/.

[176] List of websites/domains blocked by ISP's in Portugal, 2019. https://tofran.github.io/PortugalWebBlocking/.

[177] PR Newswire. New Analysis from Global Industry Analysts Reveals Steady Growth for Virtual Private Network (VPN), with the Market to Reach $77.1 Billion Worldwide by 2026, 2021. https://www.prnewswire.com/news-releases/new-analysis-from-global-industry-analysts-reveals-steady-growth-for-virtual-private-network-vpn-with-the-market-to-reach-77-1-billion-worldwide-by-2026--301368222.html.

[178] Lawful interception: the Russian approach, Privacy International. https://www.privacyinternational.org/blog/1296/lawful-interception-russian-approach.

[179] Prometheus. Prometheus - Monitoring system & time series database, 2021. https://prometheus.io/.

[180] PwC. Outlook 2022: The us digital advertising ecosystem.

[181] Emilee Rader. Awareness of behavioral tracking and information privacy concern in Facebook and Google. In *10th Symposium On Usable Privacy and Security (SOUPS)*, 2014.

[182] Rae Hodge. Why you should be skeptical about a VPN's no-logs claims. CNET, 2020. https://www.cnet.com/tech/services-and-software/why-you-should-be-skeptical-about-a-vpns-no-logs-claims/.

[183] Reethika Ramesh, Leonid Evdokimov, and Roya Ensafi. Report: Censorship in Russia. https://censoredplanet.org/russia.

[184] Reethika Ramesh, Leonid Evdokimov, Diwen Xue, and Roya Ensafi. VPNalyzer: Systematic Investigation of the VPN Ecosystem. In *Network and Distributed System Security*, 2022.

[185] Reethika Ramesh, Leonid Evdokimov, Diwen Xue, and Roya Ensafi. VPNalyzer: Systematic investigation of the VPN ecosystem. In *NDSS*. The Internet Society, 2022.

[186] Reethika Ramesh, Ram Sundara Raman, Matthew Bernhard, Victor Ongkowijaya, Leonid Evdokimov, Anne Edmundson, Steven Sprecher, Muhammad Ikram, and Roya Ensafi. Decentralized Control: A Case Study of Russia. In *Network and Distributed System Security*. The Internet Society, 2020.

[187] Reethika Ramesh, Ram Sundara Raman, Apurva Virkud, Alexandra Dirksen, Armin Huremagic, David Fifield, Dirk Rodenburg, Rod Hynes, Doug Madory, and Roya Ensafi. Network Responses to Russia's Invasion of Ukraine in 2022: A Cautionary Tale for Internet Freedom. In *32nd USENIX Security Symposium (USENIX Security 23)*, Anaheim, CA, 2023. USENIX Association.

[188] Reethika Ramesh, Anjali Vyas, and Roya Ensafi. "All of them claim to be the best": Multi-perspective study of VPN users and VPN providers. *arXiv preprint arXiv:2208.03505*, 2022. https://arxiv.org/abs/2208.03505.

[189] Reethika Ramesh, Anjali Vyas, and Roya Ensafi. "All of them claim to be the best": Multi-perspective study of VPN users and VPN providers. In *32nd USENIX Security Symposium (USENIX Security 23)*, Anaheim, CA, 2023. USENIX Association.

[190] Audrey Randall, Enze Liu, Gautam Akiwate, Ramakrishna Padmanabhan, Geoffrey M Voelker, Stefan Savage, and Aaron Schulman. Trufflehunter: Cache snooping rare domains at large public dns resolvers. In *Proceedings of the 2020 ACM Internet Measurement Conference*, 2020.

[191] Didi Rankovic. ExpressVPN critics are concerned about ties to former intelligence agents and adware. Reclaim the Net, 2021. https://reclaimthenet.org/expressvpn-sale-new-owners/.

[192] Elissa M Redmiles, Yasemin Acar, Sascha Fahl, and Michelle L Mazurek. A summary of survey methodology best practices for security and privacy researchers. Technical report, University of Maryland, 2017.

[193] Register of Internet Addresses Filtered in Russian Federation. https://github.com/zapret-info/z-i.

[194] Registry of Banned Sites. https://blocklist.rkn.gov.ru.

[195] Charles Reis, Steven D. Gribble, Tadayoshi Kohno, and Nicholas C. Weaver. Detecting in-flight page changes with web tripwires. In *NSDI*. USENIX, 2008.

[196] RIPE. RIPEstat Data API, 2021. https://stat.ripe.net/docs/data_api.

[197] S. Robertson. Understanding inverse document frequency: on theoretical arguments for IDF. *Journal of Documentation*, 2004.

[198] Roskomnadzor. https://rkn.gov.ru/.

[199] On recommendations of Roskomnadzor to telecom operators on blocking illegal information on the Internet, July 2017. https://archive.fo/LGszb;https://archive.fo/XAgJk.

[200] University of Oregon Route Views Project. http://www.routeviews.org/.

[201] Rozkomnadzor. Требования к размещаемой информации об ограничении доступа к информационным ресурса. https://eais.rkn.gov.ru/docs/requirements.pdf. Translated from Russian.

[202] David Ruiz. 21 million free VPN users' data exposed. Malwarebytes Labs, 2021. https://blog.malwarebytes.com/cybercrime/privacy/2021/03/21-million-free-vpn-users-data-exposed/.

[203] Sarah Coble. VPN Usage in US Quadruples. Infosecurity Magazine, 2020. https://www.infosecurity-magazine.com/news/vpn-usage-in-us-quadruples/.

[204] Kyle Schomp, Mark Allman, and Michael Rabinovich. DNS resolvers considered harmful. In *Proceedings of the 13th ACM Workshop on Hot Topics in Networks*, 2014.

[205] Will Scott, Thomas Anderson, Tadayoshi Kohno, and Arvind Krishnamurthy. Satellite: Joint analysis of CDNs and network-level interference. In *2016 USENIX Annual Technical Conference (USENIX ATC 16)*, 2016.

[206] Fatemeh Shirazi and Melanie Volkamer. What deters Jane from preventing identification and tracking on the Web? In *Proceedings of the 13th Workshop on Privacy in the Electronic Society*, 2014.

[207] Gurvinder Singh, Martin Eian, Svein Y. Willassen, and Stig Fr. Mjølsnes. Detecting intermediary hosts by TCP latency measurements. In *ICDF2C*. Springer, 2011.

[208] Charlie Smith. The Snowden effect: Three years after Edward Snowden's mass-surveillance leaks, does the public care how they are watched? *Index on Censorship*, 2016.

[209] Mike Snider. ISPs can now collect and sell your data: What to know about Internet privacy rules. USA TODAY, 2017. https://www.usatoday.com/story/tech/news/2017/04/04/isps-can-now-collect-and-sell-your-data-what-know-internet-privacy/100015356/.

[210] Snowball. https://snowballstem.org/.

[211] Andrei Soldatov. Russia: We know what you blocked this summer. https://www.indexoncensorship.org/2013/10/russia-censored-summer-2013/.

[212] Andrei Soldatov and Irina Borogan. In Ex-Soviet States, Russian Spy Tech Still Watches You | WIRED. https://www.wired.com/2012/12/russias-hand/.

[213] Soldatov, Andrei. Russian Surveillance State. https://media.ccc.de/v/29c3-5402-en-russias_surveillance_state_h264.

[214] Nissy Sombatruang, Tan Omiya, Daisuke Miyamoto, M Angela Sasse, Youki Kadobayashi, and Michelle Baddeley. Attributes affecting user decision to adopt a Virtual Private Network (VPN) app. In *International Conference on Information and Communications Security*. Springer, 2020.

[215] Todd Spangler. Digital advertising slowed in 2022 but was still up 10.8%.

[216] Katta Spiel, Oliver L Haimson, and Danielle Lottridge. How to do better with gender on surveys: a guide for HCI researchers. *Interactions*, 2019.

[217] Peter Story, Daniel Smullen, Yaxing Yao, Alessandro Acquisti, Lorrie Faith Cranor, Norman Sadeh, and Florian Schaub. Awareness, adoption, and misconceptions of web privacy tools. *Proceedings on Privacy Enhancing Technologies (PoPETS)*, 2021.

[218] StreisandEffect. Streisand. https://github.com/StreisandEffect/streisand.

[219] Ram Sundara Raman, Leonid Evdokimov, Eric Wurstrow, J Alex Halderman, and Roya Ensafi. Investigating Large Scale HTTPS Interception in Kazakhstan. In *Proceedings of the 2020 ACM Internet Measurement Conference*, 2020.

[220] Telegram, 2021. https://t.me/roskomsvoboda/6619.

[221] William J Tolley, Beau Kujath, Mohammad Taha Khan, Narseo Vallina-Rodriguez, and Jedidiah R Crandall. Blind In/On-Path Attacks and Applications to VPNs. In *Proceedings of the USENIX Security Symposium (USENIX Security)*, 2021.

[222] W Townsley, A Valencia, Allan Rubens, G Pall, Glen Zorn, and Bill Palter. Layer two tunneling protocol (L2TP). Technical report, RFC 2661, August, 1999.

[223] trailofbits. Algo VPN. `https://github.com/trailofbits/algo`.

[224] Nikolai Tschacher. Detecting proxies and VPN's [sic] with latency measurements, June 2021.

[225] Web blocking in the United Kingdom. `https://en.wikipedia.org/wiki/Web_blocking_in_the_United_Kingdom`.

[226] Digital Security Lab Ukraine. Crimea has a website ban list additional to russia-wide list, a research proves. `https://medium.com/@cyberlabukraine/crimea-has-a-website-ban-list-additional-to-russia-wide-list-a-research-proves-4f20fa6fc762`, September 2018.

[227] University of Chicago. Dark Patterns: UChicago/Princeton Research Reveals the Dirty Tricks of Online Shopping, 2019. `https://cs.uchicago.edu/news/dark-patterns/`.

[228] Jeroen van der Ham. Ethics and Internet measurements. In *2017 IEEE Security and Privacy Workshops (SPW)*. IEEE, 2017.

[229] Benjamin VanderSloot, Allison McDonald, Will Scott, J. Alex Halderman, and Roya Ensafi. Quack: Scalable remote measurement of application-layer censorship. In *USENIX Security Symposium*, 2018.

[230] VASExperts. DPI для СОРМ, готовимся экономить. `https://vasexperts.ru/blog/dpi-dlya-sorm-gotovimsya-ekonomit/`.

[231] Bryan Veal, Kang Li, and David Lowenthal. New methods for passive estimation of TCP round-trip times. In *PAM*. Springer, 2005.

[232] Aliza Vigderman and Gabe Turner. 2021 VPN Usage Statistics. Security.org, 2021. `https://web.archive.org/web/20211011192217/https://www.security.org/vpn/statistics/`.

[233] vpnMentor. Report: No-Log VPNs Reveal Users' Personal Data and Logs, 2021. `https://www.vpnmentor.com/blog/report-free-vpns-leak/#Timeline-of-Discovery-and-Owner-Reaction`.

[234] VPNMentor. These 7 Companies Secretly Own Dozens of VPNs, 2021. `https://www.vpnmentor.com/blog/companies-secretly-own-dozens-vpns`.

[235] Jess Weatherbed. Developer pleads guilty to hacking his own company after pretending to investigate himself. The Verge, 2023. https://www.theverge.com/2023/2/3/23584414/ubiquiti-developer-guilty-extortion-hack-security-breach-bitcoin-ransom.

[236] Nicholas Weaver, Christian Kreibich, Martin Dam, and Vern Paxson. Here be web proxies. In *PAM*. Springer, 2014.

[237] Nicholas Weaver, Christian Kreibich, and Vern Paxson. Redirecting DNS for Ads and Profit. In *USENIX Workshop on Free and Open Communicationson the Internet (FOCI)*, 2011.

[238] Allen T. Webb and A. L. Narasima Reddy. Finding proxy users at the service using anomaly detection. In *CNS*. IEEE, 2016.

[239] Valentin Weber. The worldwide web of chinese and russian information controls,. https://www.opentech.fund/documents/12/English_Weber_WWW_of_Information_Controls_Final.pdf.

[240] Zachary Weinberg, Shinyoung Cho, Nicolas Christin, Vyas Sekar, and Phillipa Gill. How to Catch when Proxies Lie: Verifying the Physical Locations of Network Proxies with Active Geolocation. In *Proceedings of the Internet Measurement Conference (IMC)*, 2018.

[241] Zachary Weinberg, Shinyoung Cho, Nicolas Christin, Vyas Sekar, and Phillipa Gill. How to catch when proxies lie: Verifying the physical locations of network proxies with active geolocation. In *IMC*. ACM, 2018.

[242] Zachary Weinberg, Mahmood Sharif, Janos Szurdi, and Nicolas Christin. Topics of controversy: An empirical analysis of web censorship lists. *Proceedings on Privacy Enhancing Technologies*, 2017.

[243] What's Happened Since Russia Banned Telegram. https://slate.com/technology/2018/04/russian-internet-in-chaos-because-of-telegram-app-ban.htm/l.

[244] Christopher Williams. How Egypt shut down the internet. https://www.telegraph.co.uk/news/worldnews/africaandindianocean/egypt/8288163/How-Egypt-shut-down-the-internet.html, 2011.

[245] Philipp Winter and Stefan Lindskog. How the Great Firewall of China is blocking Tor, 2012.

[246] Patrick Wintour. UK ISPs to introduce jihadi and terror content reporting button. https://www.theguardian.com/technology/2014/nov/14/uk-isps-to-introduce-jihadi-and-terror-content-reporting-button.

[247] Wireguard. Endpoint with changing IP. ArchLinux, 2021. https://wiki.archlinux.org/title/WireGuard#Endpoint_with_changing_IP.

[248] World IPv6 Launch, 2021. `https://www.worldipv6launch.org/measurements/`.

[249] Xu, Xueyang and Mao, Z. Morley and Halderman, J. Alex. Internet censorship in china: Where does the filtering occur? In *"Passive and Active Measurement"*, 2011.

[250] Diwen Xue, Reethika Ramesh, ValdikSS ., Leonid Evdokimov, Andrey Viktorov, Arham Jain, Eric Wustrow, Simone Basso, and Roya Ensafi. Throttling twitter: An emerging censorship technique in russia. IMC '21, 2021.

[251] Diwen Xue, Reethika Ramesh, Arham Jain, Michalis Kallitsis, J. Alex Halderman, Jedidiah R. Crandall, and Roya Ensafi. OpenVPN is open to VPN fingerprinting. In *Security*. USENIX, 2022.

[252] Tarun Kumar Yadav, Akshat Sinha, Devashish Gosain, Piyush Kumar Sharma, and Sambuddho Chakravarty. Where The Light Gets In: Analyzing Web Censorship Mechanisms in India. In *Proceedings of the Internet Measurement Conference 2018*, 2018.

[253] Yegor Sak. We're not paying for #1. Windscribe Blog, 2021. `https://blog.windscribe.com/were-not-paying-for-1-25b4e55ca10/`.

[254] Zack Whittaker. NordVPN confirms it was hacked. TechCruch, 2019. `https://techcrunch.com/2019/10/21/nordvpn-confirms-it-was-hacked/`.

[255] J. Alex Halderman Zakir Durumeric, Eric Wustrow. ZMap: Fast internet-wide scanning and its security applications. In *Security*. USENIX, 2013.

[256] Michal Zalewski. 0trace – traceroute on established connections, January 2007.

[257] ZDNS. `https://github.com/zmap/zdns`.

[258] Bendert Zevenbergen, Brent Mittelstadt, Carissa Véliz, Christian Detweiler, Corinne Cath, Julian Savulescu, and Meredith Whittaker. Philosophy meets Internet engineering: Ethics in networked systems research. In *(GTC Workshop Outcomes Paper) (September 29, 2015)*.

[259] ZGrab. `https://github.com/zmap/zgrab`.

[260] Jonathan Zittrain and Benjamin Edelman. Internet filtering in China. *IEEE Internet Computing*, 2003.

[261] Yixin Zou, Kevin Roundy, Acar Tamersoy, Saurabh Shintre, Johann Roturier, and Florian Schaub. Examining the adoption and abandonment of security, privacy, and identity theft protection practices. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, 2020.