25 April 2022

Honors Capstone

VPNalyzer: Researching VPN Vulnerabilities on a Large Scale

Studying the VPN ecosystem and application workflow to prepare VPNalyzer for public release and data analysis

Anna Ablove

Capstone Advisor: Roya Ensafi

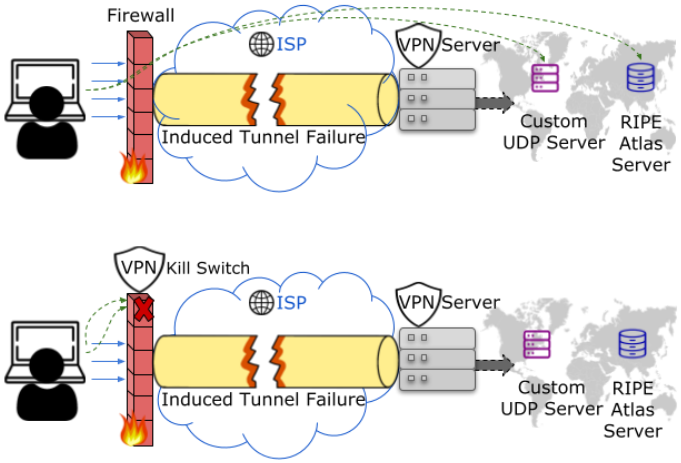
I. Introduction

The pandemic has irrevocably altered our relationship with the internet. Now, more so than ever, it is truly integral to our daily lives. People can get their news, "go" to work, have meetings, and order groceries all from the comfort of their couch — and sometimes they are forced to do so. As the digital realm has become critical to creating change in the physical one, it is of utmost importance to ensure everyone has fair and equal access to it. Additionally, specifically consideration needs to be taken for activists who live in countries with higher amounts of internet censorship and control and can become government targets if their private information is exposed.

VPN, or Virtual Private Network, is a tool which aims to allow users to browse the web with enhanced privacy and security by hiding their IP (Internet Protocol) addresses. In theory, this allows for privacy from ISPs (Internet Service Providers) and the bypassing of potential restrictions like government censorship. However, in practice many VPNs are extremely flawed and leak user data — like search queries, traffic, and other app data — to their ISPs. Not only is this a huge violation in consumer trust, but this breakdown of anonymity can be incredibly dangerous for individuals and groups using VPNs for combatting oppressive regimes.

Regarding actual examples of VPN fault points, we can examine the "kill switch" feature. A kill switch immediately disconnects a user's device from the internet if the VPN connection is lost, also known as "tunnel failure." Once connection is restored, the switch reconnects the device to the internet. Without a kill switch, the user's device will revert back to its default public IP address and effectively unmask their digital identity. Surprisingly, many VPNs either do not have kill switches, or have them turned off as a default. Furthermore, even when the kill

switch is implemented, some VPNs experience an increase in vulnerability during tunnel failure, where DNS traffic is leaked during reconnection attempts. For reference, DNS, also known as the Domain Name System, translates requests for names, like “https://vpnalyzer.org,” to IP addresses, so this can expose sites that the user visited to their ISP.



The top image depicts tunnel failure without a VPN kill switch, while the bottom image shows tunnel failure with the feature enabled.

VPNalyzer is an ongoing research project investigating commercial VPN services and seeking to give users the capacity to make informed decisions about different VPN providers. The project has three parallel tracks, quantitative and qualitative user studies, cross-platform one-click install desktop tool for users, and qualitative studies surveying VPN providers. The main focus of this capstone is to prepare the tool for public release and plan for the analysis of future data acquisitions.

The overall objective of this project is to investigate security and privacy issues in the VPN ecosystem and aim to bridge the gap between its various stakeholders; with special

emphasis on the preparation for conducting systematic, data-driven studies. Specifically, we can apply the following guiding research questions: How secure are commercial VPNs? Where do they fail? How can we enforce accountability of VPN providers? In what ways can we best empower consumers to find the most trustworthy VPNs? How can we analyze/utilize the data from the VPNalyzer tool to answer these questions?

Notably, the aforementioned questions also require a certain set of knowledge, skills, and experience. Pursuing this capstone requires a working knowledge of the fundamentals of VPN technology, as well as the work that was previously done on the VPNalyzer project, namely the alpha release. Additionally, it requires an understanding of the current VPN landscape, and the common deficiencies that are used to evaluate VPNs.

II. Developmental Stage to Public Release

As the VPNalyzer tool was in the development stage at the start of this capstone, much of the technical scope of the project relates to the finishing touches on the app, such as bundling precompiled binaries and researching and acquiring code signing certificates. The app was developed using Electron, and Node.js, and front end using React, for the three main operating systems: Windows, MacOS, and Linux. The tasks discussed relate to the Windows version, as well as the preparation of the VPNalyzer website for public release.

A. Bundling Precompiled Binaries

First, we can discuss the bundling of precompiled binaries, namely the npcap library for the Windows version. Macbooks have this functionality built in, so no further

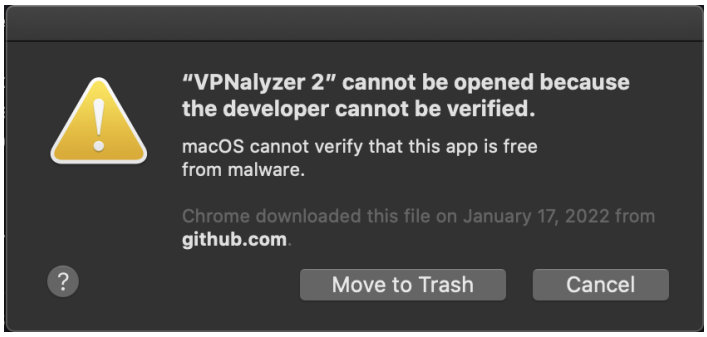
action is necessary. The npcap library is an architecture for packet capture and network analysis, which gives applications access to capture, filter, transmit, and gather statistical information from packets on the network. (Note, packets are units of data sent over networks). This allows VPNalyzer to monitor network traffic.

In order to figure out the optimal way to bundle the binary, there were several factors to take into consideration. For example, some solutions required additional packages for installation, like ‘app-root-dir’ for locating certain files in the subdirectory of the application, i.e. where to place the binaries. Additionally, as Node.js has experienced several updates over recent years, there were many versions of solutions to comb through, some deprecated or needlessly complex given new software capabilities. Ultimately, the simplest solution involved making modifications to the package.json file.

B. Code Signing Certificates

Next, we can discuss the process for acquiring a code signing certificate for VPNalyzer. In order to list the app on an official app store (Apple or Microsoft), developers must buy a code-signing certificate, which is a file housing a digital signature that verifies their identity and ensures that the code has not been modified by a malicious third party since the signing occurred. Certificates are issued by certificate authorities, and it is important to buy from a reputable provider, as certificates issued by defunct/untrustworthy certificate authorities will not inspire confidence in the application. Furthermore, encryption length, type of site seal, warranty, and a myriad of other factors vary based on the certificate authority.

In terms of the user facing feedback once the app is downloaded and opened a warning will pop up without a valid certificate.



Observe, the picture above shows the unsigned app warning message on a Mac.

Additionally, on Windows there are two types of certificates: code signing certificates and EV (Extended Validation) code signing certificates. The former, the less expensive option, only displays a warning during installation that goes away once enough users have downloaded and used the application. Conversely, the EV certificate represents a higher level of trust and thus works immediately. However, there are drawbacks to EV certificates, they are bound to physical USB dongles and cannot always be exported like regular code signing certificates can. Though, the biggest drawback is the cost, on average a couple of hundred dollars more than code signing certificates.

Ultimately, we opted for a regular code signing certificate. However, as the warning persists until a certain number of downloads, for the initial release the Windows version will remain unsigned. Still, it was a good lesson in researching software with the intention to purchase, and it was interesting to prioritize public-facing elements and cost, as opposed to purely academic considerations.

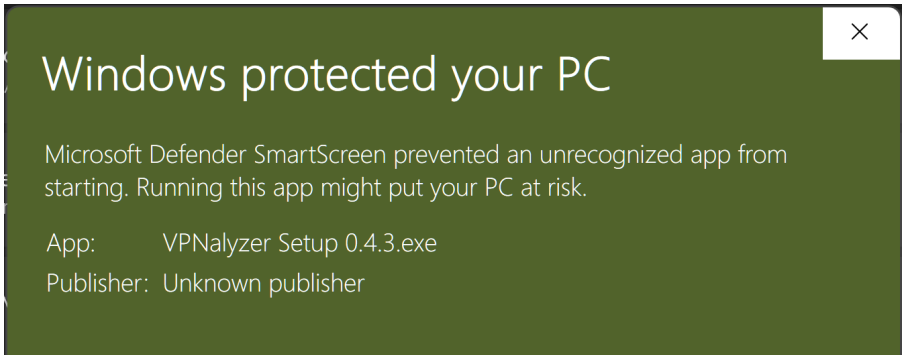
C. VPNalyzer Website Additions

As part of preparing the tool for public release, additions were made to the website to increase user understanding and allow for downloads of the tool. I implemented an explanations page that goes into more detail on each of the tests run, which is in addition to the individualized breakdown of results discussed in the following section.

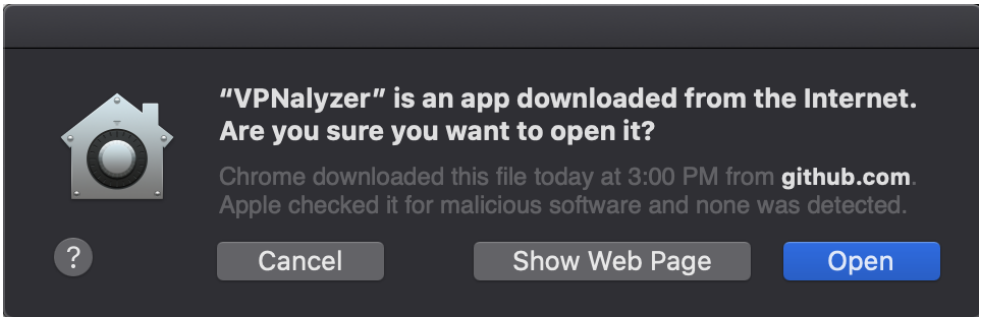
> **BGP Implementation:** If your VPN/ISP does not implement BGP safely, it means it does not drop invalid prefixes. Learn more about this from other sources: Is BGP Safe Yet? and Resource Public Key Infrastructure (RPKI).

Above is a screenshot of a test description (BGP Implementation) on the explanations page. Note the links to external resources for further information.

Additionally, I worked on a page detailing the download instructions for each of the different operating systems. This is particularly important, as there are warnings that show up during the installation process. Some examples are shown below:



This is the warning message on the unsigned Windows app, not shown are the “Run Anyway” and “Don’t run” buttons on the bottom right.



This is a warning message on the signed Mac app.

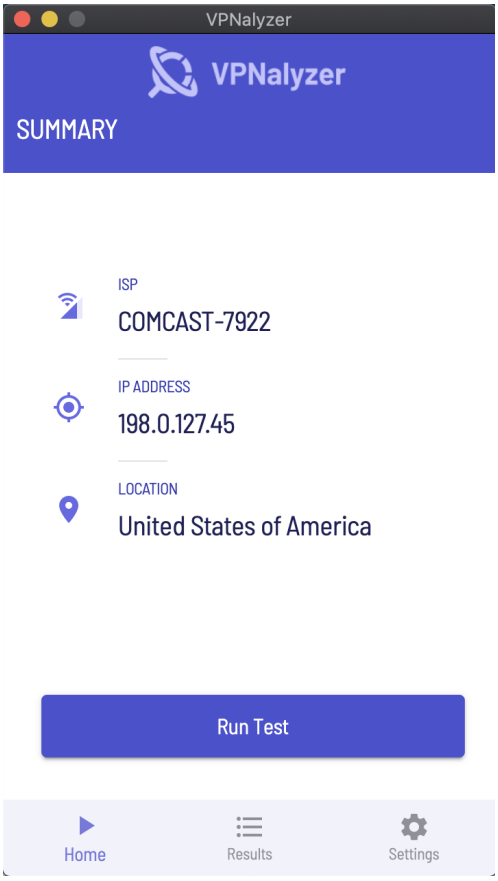
Additionally, for the Linux version, there are instructions about adjusting file permissions to allow the installation file to run as an executable.

III. Current App State and Walkthrough

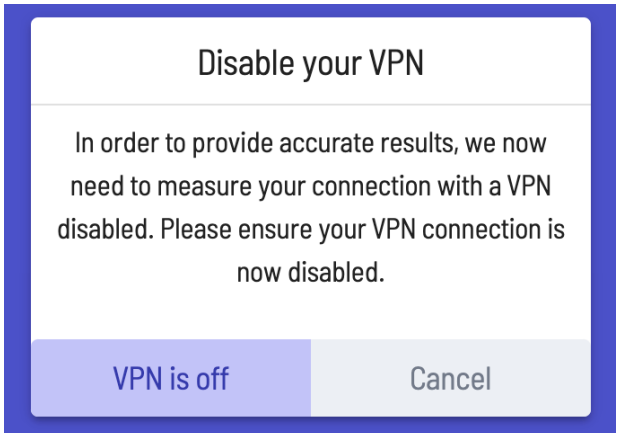
VPNalyzer has 14 tests implemented, including the aforementioned Kill Switch Test, as well as DNSSEC support, IPv6 Leakage, TLS Interception, Geolocation validation, Port Scan, Bandwidth and Latency estimation, and more. Along with the tests, the app collects packet capture. The data is uploaded to Google Cloud Storage for analysis. Preliminary results are available in the application and there is also a link to comprehensive results hosted on the website. This is discussed in greater detail in the following section.

A. Walkthrough

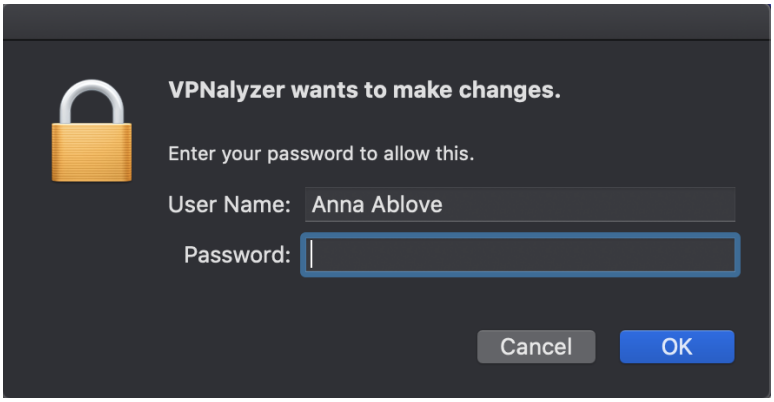
The only prerequisites for using VPNalyzer, are an internet connectivity and some VPN installed to test. Upon opening the application, users see the following screen.



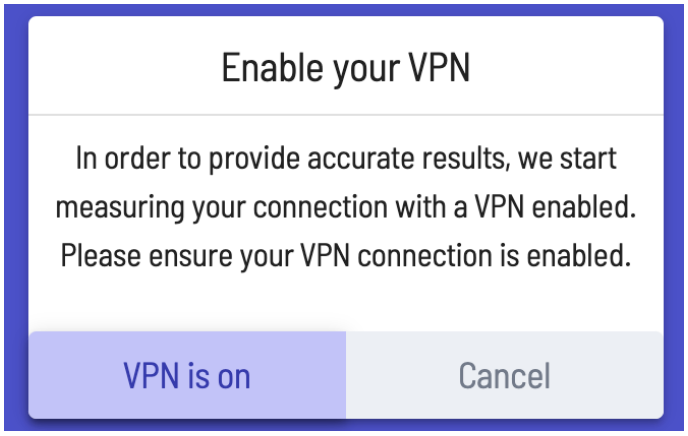
Notice, the users IP address, ISP, and location is visible. Clicking the “Run Test” button starts a test and prompts the following pop up.



Once users have confirmed their VPN is off, a system prompt will ask for the admin password.

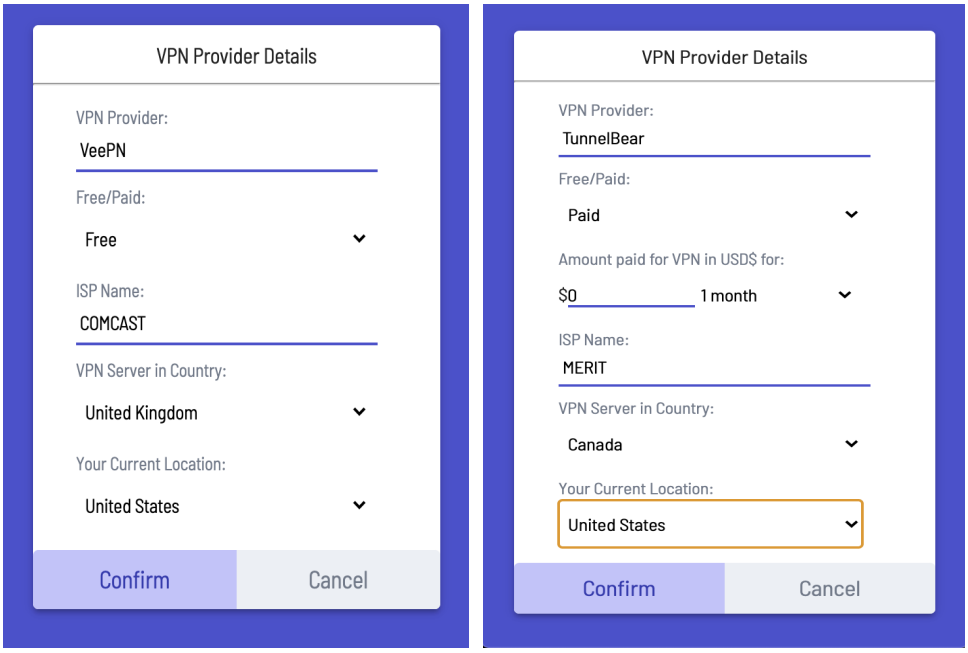


After entering their password, another prompt will request VPN activation.

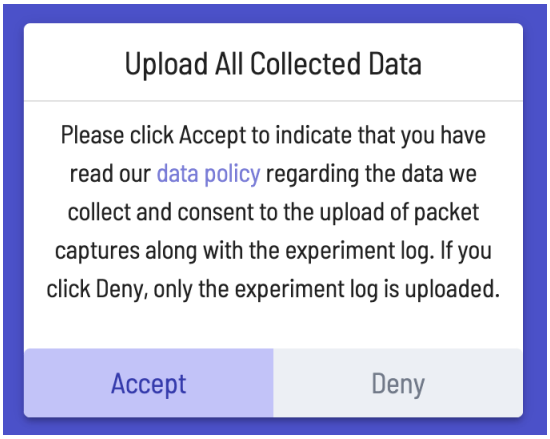


Now, the test can really begin. Halfway through the test, there will be another request to turn off the VPN, though no repeated system prompt.

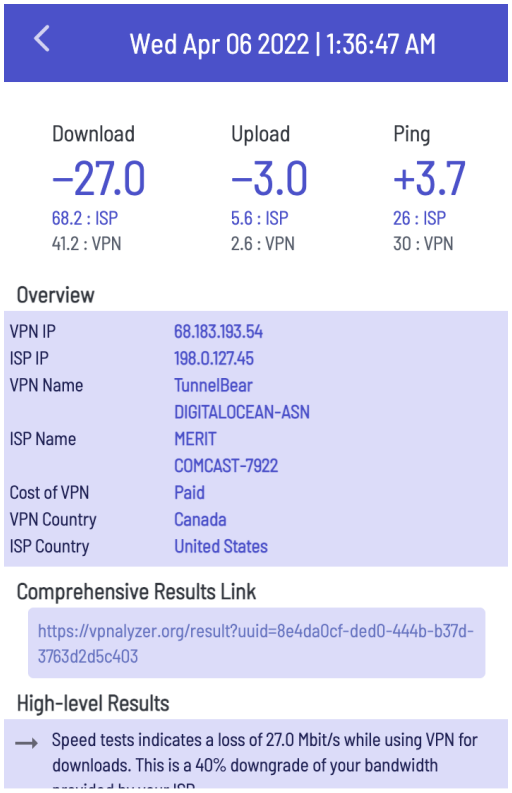
The test suite should run in around 9 minutes. Once it is done, users will reach a screen requesting information on the VPN provider they are testing. A free VPN is shown on the left, and a paid VPN is shown on the right, note paid VPNs also query cost and subscription length here.

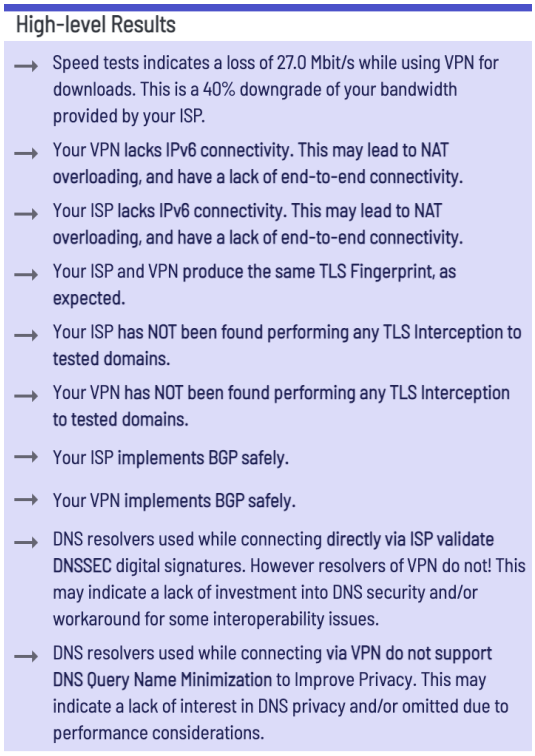


Before the results are available, users are presented with an option to upload either all collected data, including the packet captures, or just the experiment log.



Now, the preliminary results are available. At the top is a speed comparison for the ISP and the VPN. Following is an overview of the VPN/ISP's IPs, names, and associated countries. Then, the comprehensive results link, which loads the results page discussed above. Finally, the high level results consist of a series of statements. Note, these are similar to those found on the comprehensive results page, with a few minor differences.

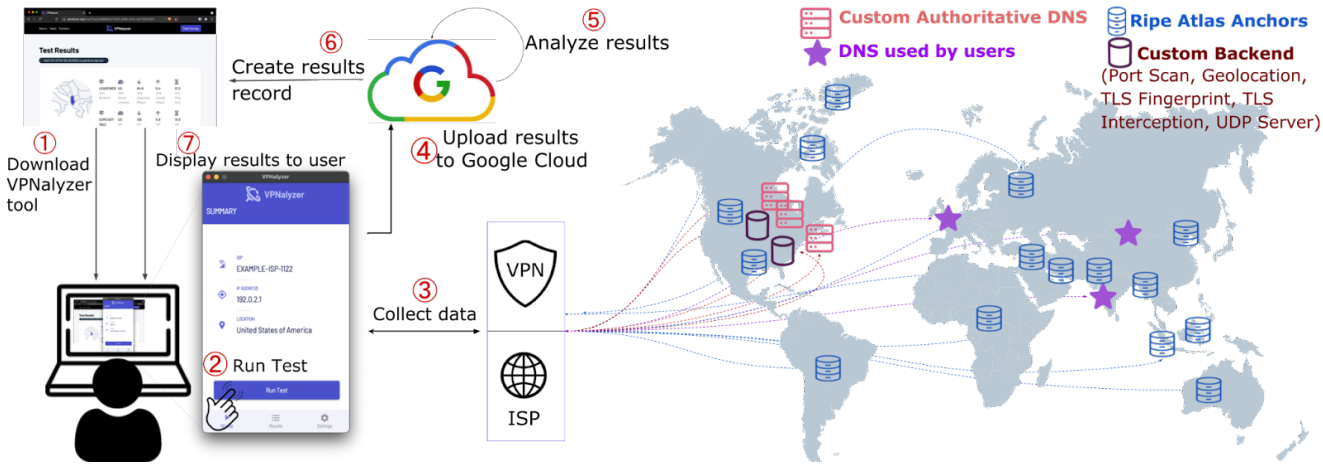




The high level results include more qualitative/helper statements, ex. “That is excellent!” or “...This may lead to…”. These serve to help those with lesser technical experience make sense of the data. Additionally, the website has an explanations page that breaks down each test in greater detail, as discussed previously.

B. Data Collection Pipeline

Once the tool goes public, a large scale data collection pipeline will be initiated to handle the analysis of the results. We will collect crowdsourced, real-world data from users; this includes reaching out to users in other countries. The subsequent study will be conducted assuming region-specific VPNs, and we will draw comparisons of local VPNs from different regions. Overall, we will analyze VPN security/privacy features as well as common leaks and misconfigurations.



The graphic depicts the data collection pipeline.

IV. Conclusion

As the project is still in progress, the questions about VPN security and provider accountability will continue to be useful tools in guiding research. VPNalyzer will be available for public release by early this summer (2022), and the data analysis will take place once the tool is up and running. This was my first time being involved in a larger-scale, long-term research project, and I learned a lot about working in a group/lab environment. Additionally, I had a great deal of reading and studying to do in the beginning while getting acquainted with the project, and it was very rewarding to translate that into actually working on the tool itself. I feel I gained a better understanding of the research process through this experience, and I look forward to continuing to work on VPNalyzer.

V. Acknowledgments

Special thanks to the VPNalyzer team: Reethika Ramesh, Diwen Xue, Leonid Evdokimov, Anjali Vyas, and Roya Ensafi.