University of Twente
Faculty of Electrical Engineering, Mathematics and Computer Science

Investigating the Geo-Unblocking Capabilities of Commercial VPN Providers

Committee	*prof. dr. ir. R.M. van Rijswijk-Deij*
	dr. A. Sperotto
	dr. M.H. Everts

Master Thesis by Etienne Khan
Enschede, January 21, 2022

Prologue

"[redacted]", Captain Raymond Holt replies, in his usual dry manner. Detective Jake Peralta, clearly aware of the significance of the captain's words, can only bring himself to repeat Holt's words, "[redacted]", before the scene changes. We now see Captain Holt standing [redacted], [redacted], because he has been [redacted]. As the camera zooms out of [redacted], we hear Frank Sinatra's "That's Life" playing. A fitting tune to end season six of the hit comedy show "Brooklyn Nine-Nine". The credits start rolling and Netflix reminds me to spend more time with them, how thoughtful.

Naturally, as a student, I have nothing better to do with my time, so I lean back and wait for season seven to continue the whacky adventures of the 99th Precinct. To my surprise, instead of continuing where I left off, Netflix started recommending me different shows. Going back to the menu, season seven was nowhere to be found.

Of course, I could have been mistaken, and season seven is not available yet, but a quick search for the show on IMDB assured me, that the season has indeed already been filmed, cut, and distributed. So why wasn't I able to watch it? My search would relatively quickly lead me towards the answer: licensing.

At the time of my search, Canada was the only country where a Netflix subscriber would be able to watch all seven available seasons. Most countries with Netflix availability would only offer six seasons. A sad exemption being the US, where Brooklyn Nine-Nine is not even available on Netflix, but instead all presentation rights lie with Hulu, a Disney owned video on demand service.

At this point my options are rather limited. I can wait for the show to eventually come to the Netherlands, but that would test my patience, something I am not known for. I could just forget about it (no!), or I could do the most sensible in this situation: Apply for a semester abroad in Canada and beat the system. This latter option lost a bit of its luster, due to a certain ongoing global event. Back to square one. I would need to find a way to let Netflix think, that my playback device is within Canada, without actually going to Canada. Strangely the Canadian embassy did not reply to my request if I could pass by on Friday evening for "Netflix 'n chill".

At this point I have thought about all ethically responsible solutions to this problem, which is why I never mentioned piracy as a backup plan. But there is just one thing left to try: to spoof my location by means of a VPN connection. A VPN connection places my device onto a different network, via the Internet, as if my device were physically there. Researchers use VPN connections to get access to their research data at the university. White collar workers can

work from home and enjoy peace and quiet instead of being in a noisy open office environment. And I can VPN into a cozy data center somewhere in Canada, to watch a comedy show about a dysfunctional police squad.

To establish a VPN connection, you need a VPN client and a VPN server. The client in this case is my laptop, so that is taken care of. The next step is having a VPN server in Canada. A remote host onto which I can install VPN server software will cost around $5 to $15 per month, and it is a price I am willing to pay. After installing the appropriate software and establishing the VPN connection, I verified that my connection indeed originated out of Canada. Logging in to Netflix I was greeted with the top 10 of watched content in Canada and navigating to Brooklyn Nine-Nine showed me what I was looking for: Season seven. I pressed play and waited for the intro to roll. Instead of the catchy theme song playing, only a small popup appeared kindly asking me to turn off my VPN connection.

I was perplexed, how did Netflix see through my scheme? As a matter of fact, there are many providers who specialize on providing VPN connections with servers in different countries for a fee. Instead of having to learn to manage a server and set up VPN server-software, these commercial VPN providers have taken care of all of this and by paying the fee I can install their software which lets me connect to any VPN server they offer with not more than a few clicks. Furthermore, they advertise on a multitude of platforms and claim that their VPN connection will not be detected by the likes of Netflix and other streaming services. But what differentiates their VPN server from my VPN server? Do their claims actually hold true, or is it all a quick money grab? And how on earth did the Dutch military get involved in all of this? These questions and more will be answered in depth in this thesis.

This, is Stranger VPNs.

Contents

1. Introduction

1.1. Motivation

Many publications shall introduce their work by mentioning that privacy or the perceived notion of privacy has become an important part of online life. And this one is no different. Journal publications[1], but more importantly news articles[2, 3] have regularly kept the importance of secure data transferal as well as secure storage and the implications of data breaches in the public eye. Similarly, news about (nation state controlled[4]) censorship[5] travels around the world. But bad news is not the only thing traveling around the world. In late 2019 a respiratory illness called COVID-19, caused by the virus SARS-CoV-2, spread first from China to Italy, where it expanded through all of Europe, and finally over the whole world. Governments worldwide urged their citizens to stay home as much as possible, either by appealing to the duties of a model citizen or by enforcing so-called lockdowns, non-pharmaceutical interventions which included stay- and work-at-home, curfews as well as other societal restrictions. Working from home, as the name strongly implies, has let employees exercise their occupational duties from the comfort of their home as long as it does not mandate a physical presence. While working from home protects the employees from potentially spreading the virus among each other or during commute in the physical world, the digital exchange of company data between employer and employee also needs a layer of protection.

One such measure is called *virtual private network*, known by its more commonly used acronym VPN. A VPN connects two clearly defined participants, a host and a client, with an encrypted connection, which will allow the secure exchange of data. As a concrete example, the office of the employer will act as the host and the home computer of the employee will act as the connecting client. Even though the employee is not present at the office to which the VPN connection has been established, he or she may access all relevant digital resources, such as network drives, databases, code repositories or even printers, as if he or she were physically present.

VPNs as a technology to facilitate remote connectivity have however found their entry onto the home computer on a large scale long before they became a necessity during the work-from-home measures. While VPNs have always been a staple for the telecommuter, a whole industry of commercial VPN providers has grown over the years to cater to privacy conscious individuals, because establishing a VPN connection can do more than just give access to work-related resources.

Before we explain this additional property of VPNs, we first very briefly summarize the workings of an Internet connection. In general, an Internet connection from an Internet service provider (ISP) supplies a single household with Internet connectivity. The members of this

household may connect any amount of Internet capable devices to their (modem-) router, a device that establishes communication between the Internet and the devices of the household's members. The ISP assigns an Internet protocol (IP) address to the (modem-) router, which uniquely identifies the household to the ISP, but not the individual users of the household. Every time a household member visits any Internet connected resource, the IP address assigned to the household by the ISP will appear in the data exchange. The household's IP will appear as a sender on outgoing requests and as a receiver when a reply is sent from the resource. The opposite is true for the requested resource. This allows the ISP to observe all destinations the members of the household decide to connect to.

We may now bring our attention back to the additional property of the VPN. If one member of the household decides to make use of a VPN with traffic forwarding capabilities, the situation explained above will not hold true anymore. Instead, the sender/receiver fields of the data packets, will only contain two distinct IP addresses, one being the household's address, the other one being the IP address of the VPN host. From this moment on, the VPN connection acts as a bridge between the VPN client within the household and the VPN host somewhere on the Internet. This effectively thwarts the ISP's ability to observe any data exchange between the household and the Internet, as IP addresses other than the VPN host's are not visible to the ISP anymore. In other words, if the user of the VPN client requests an online resource, like a website, it will look as if the request originated from the VPN host.

These two features ((1) obscuring Internet activity from the ISP and (2) masking the source IP) are the essence of the commercial VPN provider industry. The first feature lets a user exchange ideas and share data freely in cases when this is otherwise not possible, such as when the ISP is run by an authoritarian government, which spies on its users. The second feature allows the user to circumvent restrictions based on the user's origin. For example, a British public broadcaster may only offer its content to users within the United Kingdom. In order to still access the resources from outside the UK, a VPN host in e.g. London would suffice. This can be generalized to any public broadcaster globally, as well as to the commercial entertainment industry.

Even though many commercial video on-demand (VOD) providers operate globally, they may not offer the same programming globally. The programming depends on the licensing of the material by the appropriate rights holders and therefore it may occur that a local VOD provider is the exclusive distributor of certain content within a given jurisdiction. Hopping between these different content jurisdictions becomes trivial with the help of commercial VPN providers. In this thesis the practice of connecting to a different jurisdiction with the goal of receiving access to a different content library, compared to the user's domicile, will be referred to as **geo-unblocking**.

A survey conducted in Q1 of 2018 has asked VPN users between the age of 16 and 44 (n = 24 462), if they have ever used their VPN provider, to access different entertainment content, to which 49% of the respondents have answered with yes[6]. (The same group of users has been asked if they use their VPNs to keep anonymity while browsing, to which only 31% responded positively.)

Placing this before the backdrop, that the commercial VPN provider business is a billion dollar industry, one could extrapolate that a substantial amount of users make use of the capability to geo-unblock streaming content. As per a report from Allied Market Research commercial VPNs accounted for $25.41 billion in 2019, and is projected to reach $75.59 billion by 2027[7]. This figure includes institutional users (i.e. companies outsourcing VPN deployment to third parties), as well as retail users. Considering that popular VPN provider NordVPN, who exclusively advertises to retail users finds itself among some of the top spenders in advertising (1.6 million USD as of the first quarter of 2018, compared to T-Mobile's 1.91 million USD and Office Depot's 1.38 million USD)[8], one may make the assumption that this advertising budget is a direct consequence of a large retail user base.

1.2. Goal, Research Questions & Approach

1.2.1. Goal

A substantial amount of commercial VPN users not only use their VPN connection for privacy related means, but also to access geo-blocked content. So much so, that commercial VPN providers advertise and heavily promote this feature. This advertisement does not go unnoticed by the VOD providers, who wish to keep the geo-block in place. Based on the amount of total revenue generated though, offering this service seems to be lucrative, in spite of all the roadblocks put in place. We therefore formulate the following research goal for this thesis:

Goal: *How do commercial VPN providers facilitate geo-unblocking?*

To answer the research goal stated above, we will look at which commercial VPN providers offer geo-unblocking and select a subset of these providers for further investigation. Academic research has only recently taken an interest in commercial VPN providers[9] and therefore no rigorous selection criteria are available. Similarly to an end-user who wishes to purchase a VPN subscription, we will survey the available providers and from there narrow down the selection by looking at independent reviews and the promises of the commercial VPN providers themselves. Furthermore, we will divide the research goal into three distinct research questions formulated below.

1.2.2. Research Questions & Approach

VOD providers by contract have to protect the interests of their licensing partners. Without knowing how to differentiate between a geo-unblocking and a normal viewer, the VOD providers cannot abide by their contract. Therefore, we can formulate our first research question:

> **RQ 1:** *How do commercial video on-demand providers detect geo-unblocking attempts?*

To answer this question we will first look at how a user's physical location can be geographically accurately deduced, as soon as he or she establishes the connection, and therefore nothing but the IP address is known. The process is called geolocation, and we will discuss its early beginnings as well as state-of-the-art geolocation solutions. These state-of-the-art solutions play a major role in detecting geo-unblocking.

The previous research question already hints at IP addresses playing an important role. Knowing how VOD providers defend against geo-unblocking, namely by blocking IP addresses, beckons the question, how do commercial VPN providers circumvent these blockades? Thus, we can state our second research question:

> **RQ 2:** *Which methods are used by commercial VPN providers to perform geo-unblocking?*

Our approach here will be to try to follow the paths our traffic takes, when connected to a commercial VPN provider and using its geo-unblocking feature.

There are no concrete numbers on the total amount of commercial VPN users, so we can only estimate the size of the commercial VPN industry, by looking at the projected sales figures by market researchers. In one research among roughly 24 000 VPN users, about 50% have responded positively to having used the geo-unblock feature. Assuming that there are more than 24 000 commercial VPN users worldwide, we formulate the following research question:

> **RQ 3:** *How do the geo-unblocking methods employed by commercial VPN providers scale to size?*

To answer this last question we will deploy a large scale measurement infrastructure that will measure the selected VPN providers simultaneously and from several vantage points over the span of several months.

1.3. Structure & Contributions

The remainder of this thesis has the following structure: Section 2 will explain how we arrived at our selection of commercial VPN providers. Section 3 gives a brief introduction to geolocation and enhanced geolocation databases which are able to classify IP addresses on much more than just their physical location. The next section, Section 4 investigates how the VPN providers are circumventing the barriers the streaming providers have put in to place, to defend against geo-unblocking. We deployed two novel methods to be able to gather this data. In Section 5 we explain how we embedded our novel methods into our methodology and also present our results of the longitudinal measurement. With these results we are able to divide the geo-unblocking black box intro 3 distinct approaches the commercial VPN providers take.

Before concluding the work (Section 7), the related work section (Section 6) will help place this thesis in the currently available body of research.

Of particular importance are several of the appendices. Most of them are referenced throughout the text, but we still give a small overview nonetheless. Appendix A (Technical Appendix) may be read accompanying to Section 5 (Methodology & Results), as it will give insight into the technical limitations of the measurement setup. Appendix B (Ethical Committee Correspondence) may be read at any time, as it mostly functions as an alternative introduction. Lastly Appendix C (Dutch Ministry of Defense) and Appendix D (Letter to identified geo-unblocking hosts at the University of Twente) should be read after Section 5, or when they are referenced in that section.

Lastly, Appendix E is left over: Our poster we presented at ICT.OPEN 2021 and won the second prize in the poster competition.

2. Commercial VPN selection

The commercial VPN ecosystem is ever-changing, with some providers going out of business, or being shutdown by law enforcement due to almost exclusively offering their services for criminal purposes. To fill this vacuum, new providers take their places. Some providers stand the test of time and are a stable presence in the industry.

In 2018, Khan et al.[9] have identified 200 unique commercial VPN providers through three selection methods:

- Popular review sites
- Reddit crawl
- Personal recommendations

The fact that neither an exhaustive nor a more systematic approach for selection exists, highlights the size of this industry and also the lack of academic overview. This thesis only considers commercial VPN providers that offer geo-unblocking and thereby reducing the set of possible providers. This enables a more targeted search, for example by utilizing specific websites such as the "VPN TIER LIST"[10].

2.1. VPN TIER LIST

The "VPN TIER LIST" is a website maintained by an individual who goes by the name "Tom Spark" and its main purpose is to maintain a hierarchical ranking of the "best" commercial VPN providers available for purchase. "Best" is set in quotes, because all VPNs listed on the site are solely reviewed by Mr. Spark. The review scores are measured on a 5-point scale (including zero) and are comprised of the following aspects:

- Price
- App
- Speed
- Reputation
- Support
- Streaming

A final column then averages the scores to create a final rating. All of his reviews are conducted in video format and can be watched on the video platform YouTube.

Mr. Spark's drive to review all of these providers does not seem to stem from an intrinsic academic rigor, as all of his reviews also contain so called affiliate links. Every sign-up to a commercial VPN provider by means of an affiliate link, will result in a non-disclosed monetary compensation for Mr. Spark.

We would like to refer to Khan et al.[9], who wrote:

> "Unsurprisingly, almost all of the top sites participate in some form of affiliate marketing. So the rankings here must be taken with a grain of salt, but they offer the best public metric for popularity available."

In the case of the VPN TIER LIST, Mr. Spark claims that that his reviews are unbiased and not affected by any third party:

> "Reviews are not affected by commissions or any other monetary basis. Reviews are made from my opinion and subsequent tests, and then ranked on the tier list. We do not accept sponsorships or paid reviews to influence opinions, and standard affiliation agreements are made and whatever a VPN offers the channel, is what we accept. We do not negotiate better deals to influence rankings."

2.2. Additional Selection Methods

Similarly to Khan et al.[9], we also traversed Reddit (a popular online forum) and other review sites for recommendations. A notable exception is the inclusion of YouTube advertisement. YouTube content creators often accept a sponsorship with a third party in return for product placement in their videos. To highlight just one instance of this, consider the video "Raid 2020 (NES) - Angry Video Game Nerd (AVGN)" by Cinemassacre[11]. The video opens with a sponsored message for a commercial VPN provider (77 seconds) and between 00:25 and 00:47 it especially emphasizes the geo-unblocking capabilities. The video has amassed over 2 848 000 views and the channel itself boasts over 3.53 million subscribers. Because nothing is known about this particular sponsorship, we cannot reason about the costs for the provider, but considering that this channel has advertised the provider several times, one can assume that it is a mutually beneficial partnership.

Ultimately these methods did not add any new commercial VPN providers to the selection criteria, as all of them were already included in the VPN TIER LIST, which has proven itself as a reliable resource.

2.3. Commercial Provider Selection

Purchasing a subscription for every geo-unblocking commercial VPN provider is not feasible. Each provider incurs a monthly fee, which can be brought down, by committing to longer subscriptions (i.e. 12 or 24 months). Not only does that reduce the monthly cost of some providers by roughly 75%, it also aids the longitudinal character of this study. By combining the previous methods, we chose to select 3 of the largest and most prominent commercial VPN providers, 2 medium-sized providers and 1 new provider.

Provider	Cost	Subscription in months	Provider Maturity
Surfshark	€ 42.00	24	2018
CyberGhost	€ 11.99	1	2004
CyberGhost	€ 33.00	12	
ExpressVPN	€ 12.00	1	2009
ExpressVPN	€ 88.25	15	
WeVPN	€ 26.27	12	2020
NordVPN	€ 75.28	24	2012
PrivateVPN	€ 39.30	24	2009
Sum	€ 328.09		

Table 1: Listing of purchased commercial VPN providers.

An overview of the accumulated expenses can be seen in table 1. The two single month purchases represent test subscriptions, which were purchased for a pilot study. The Internet Measurement & Security subgroup of DACS[12] kindly reimbursed us for the costs for which we are very grateful.

3. From Geolocation to Geo-blocking

For decades, the Internet Protocol (IP) has been one of many responsible parts in delivering and receiving the data we send across the Internet. It not only facilitates the connectivity between hosts on the Internet, but it indirectly also allows for geolocation. (We say *indirectly*, because geolocation is not a feature of the protocol, but a derived property of the uniqueness of IP addresses.) Geolocation is the practice of attributing an IP address to a physical location on earth, mostly countries, but more granular assignments like cities are also possible [13]. This technique has found its use in advertisement (Discover the best deals in your area: ANONYMOUS PROXY), fraud detection, customized content (i.e. local weather), geo-blocking and more [14]. Academically, geolocation has been conceived in the early 2000s [15] and was limited to physical location mappings. Currently, geolocation databases have expanded beyond those basic mappings, by adding additional meta information to IPs such as: connection speed, connection type (wired or mobile), connection origin (data center or residential) as well as proxy detection (Tor, VPN, open proxies, etc.)[16]. The latest additions even include demographics [17], but so far there is no academic research available on the accuracy or methodology of these enhanced databases.

This section will highlight the difficulties of acquiring correct geolocation data and then discuss the accuracy of several (commercial) databases. At the end, we discuss how content providers rely on these enhanced geolocation databases to detect geo-unblocking attempts, with the goal of answering **RQ1**.

3.1. The Difficulties of Accurate Geolocation

Every IP address belongs to an autonomous system (AS). An AS is defined as "a set of routers under a single technical administration [...]" [18] and consequently this administration needs to be a registered entity with a regional Internet registry (RIR). Currently, there are five RIRs: AfriNIC serving Africa, APNIC serving parts of Asia and the Pacific region, ARIN serving North America and parts of the Caribbean, LACNIC serving Latin America and parts of the Caribbean, and RIPE NCC serving Europe, parts of Asia and the Middle East, as stated in *RFC7020* [19].

As a practical example we can look up information on the university's IP address in Europe's RIR database. A simple DNS query for the *A record* of *www.utwente.nl* reveals the IP address: *130.89.3.249*. We may now enter this IP into RIPE's search (`https://apps.db.ripe.net/db-web-ui/query`) to receive all officially registered information. Among the output of the query we find data fields like country (Netherlands), address (Drienerlolaan 5 in 7500 AE Enschede) and AS number (AS1133). Even though this data is correct, the picture is less clear on internationally operating companies and institutes. Take for example Telia Company AB, a Swedish

multinational telecommunications company and mobile network operator. It serves Sweden, Finland, Norway, Denmark, Lithuania, Latvia and Estonia, but it also runs an international IP backbone network which is ranked number two in the world through Telia Carrier [20].

We have procured five distinct Telia (AS1299) IP addresses as a result from a traceroute between two hosts in the Netherlands and Switzerland (1).

```
2  zch-b2-link.ip.twelve99.net (62.115.180.122)    1.908 ms
3  zch-b1-link.ip.twelve99.net (62.115.138.12)     2.050 ms
4  ffm-bb1-link.ip.twelve99.net (62.115.138.16)   18.548 ms
5  adm-bb4-link.ip.twelve99.net (62.115.122.200)  16.754 ms
6  adm-b10-link.ip.twelve99.net (62.115.120.229)  13.722 ms
```

Listing 1: Traceroute excerpt from a Swiss to a Dutch host

Running the same query as before will show that the country of origin is Europe. This is only somewhat correct, because Europe is not a country and the hostnames shown on the left side of listing 1 contain city codes which clearly reference cities in three different European countries. These city codes are very often International Air Transport Association (IATA) airport codes, but not always, as there is no universal naming consensus [21, 22]. In this particular case *zch* stands for *Zürich* (Switzerland), *ffm* is *Frankfurt am Main* (Germany) and *adm* stands for *Amsterdam* (Netherlands). [1]

Naming convention is not definitive proof of the indicated location, but Telia themselves offer two methods to verify these locations. Firstly they supply a map of their network (`https://www.teliacarrier.com/our-network.html`), where we can indeed see points of presence (PoP) in the aforementioned cities and secondly they offer a looking glass (`https://lg.twelve99.net`). A network looking glass (lg) lets you run simple commands like *ping* or *traceroute* on the infrastructure of the looking glass provider.

Listing 2 shows that the round-trip time (RTT) is lower than $1ms$ when we select Telia's Swiss host, to ping one of the IPs we suspect to be in Switzerland. The results are similar for the other locations. On the other hand, if we select the same Swiss IP and ping it from Telia's Amsterdam location, we expect to see a substantial increase in RTT, as shown in listing 3.

[1] **N.B.:** The proper IATA codes would be *ZRH*, *FRA* and *AMS* respectively.

```
Router: zch-b2 / Zurich
Command: ping count 5 62.115.180.122

PING 62.115.180.122 (62.115.180.122): 56 data bytes
64 bytes from 62.115.180.122: icmp_seq=0 ttl=64 time=0.093 ms
64 bytes from 62.115.180.122: icmp_seq=1 ttl=64 time=0.076 ms
64 bytes from 62.115.180.122: icmp_seq=2 ttl=64 time=0.074 ms
64 bytes from 62.115.180.122: icmp_seq=3 ttl=64 time=0.086 ms
64 bytes from 62.115.180.122: icmp_seq=4 ttl=64 time=0.100 ms

--- 62.115.180.122 ping statistics ---
5 packets transmitted, 5 packets received, 0% packet loss
round-trip min/avg/max/stddev = 0.074/0.086/0.100/0.010 ms
```

Listing 2: RTT from Telia's Zürich PoP to a Telia IP address in Switzerland

```
Router: adm-b11 / Amsterdam (InterXion AMS7, Schiphol)
Command: ping 62.115.180.122

PING 62.115.180.122 (62.115.180.122) from 62.115.129.202 : 72(100) bytes of data.
80 bytes from 62.115.180.122: icmp_seq=1 ttl=61 time=12.7 ms
80 bytes from 62.115.180.122: icmp_seq=2 ttl=61 time=12.6 ms
80 bytes from 62.115.180.122: icmp_seq=3 ttl=61 time=12.8 ms
80 bytes from 62.115.180.122: icmp_seq=4 ttl=61 time=12.6 ms
80 bytes from 62.115.180.122: icmp_seq=5 ttl=61 time=12.5 ms

--- 62.115.180.122 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 49ms
rtt min/avg/max/mdev = 12.503/12.665/12.852/0.153 ms, pipe 2, ipg/ewma
    12.286/12.678 ms
```

Listing 3: RTT from Telia's Amsterdam PoP to a Telia IP address in Switzerland

So far we have seen, that there is no ground truth when it comes to IP geolocation if we rely on the provided information by the RIRs. Furthermore, the shown examples so far are nice, in the sense that Telia provides proof of their location through their websites (network map and looking glass) and city codes in the hostname. The picture can also be less clear though, as

shown in listing 4. This traceroute shows the last three hops to an IP address of a residential connection close to Arnhem. From the three hops, only one contains city information encoded into the hostname: *nm* (very likely Nijmegen due to its proximity to Arnhem). The other two hops are an IP address without hostname and a hostname without city information. The provider in question does not offer the same resources as Telia (i.e. network map and lg), so we can't easily verify the location.

```
10  nm-rc0110-cr102-be2-2.core.as33915.net (213.51.7.85)  19.652 ms
11  212.142.53.230 (212.142.53.230)  19.855 ms
13  ip-217-103-110-213.ip.prioritytelecom.net (217.103.110.213)  24.112 ms
```

Listing 4: Traceroute to an IP address located in Arnhem (Netherlands)

In summary: Relying only on RIR provided information will sometimes lead to inaccurate data, so more data sources are needed. Some providers offer network maps and looking glasses, but not all providers do this and network maps cannot be easily queried for information. Lastly, very often a traceroute to an IP address will offer helpful clues in the form of city codes, but not all hostnames contain these codes. To achieve more accurate geolocation results, additional methods on top of relying on provider provided information are required. We will explore these methods and their accuracy in the following subsection.

3.2. Towards more Complete and Accurate Geolocation

A lot has changed from the humble beginnings in the early 2000s to now, 2021. Geolocation is not only an academic research field anymore, as commercial entities have joined the fray, e.g. [23, 24, 25, 26].

The techniques used by these companies are proprietary[27], but we can assume that they include some state-of-the-art methodologies such as private data feeds [28], triangulation through geographical constraints [29] as well as delay and topology measurements [30]. A noteworthy case to highlight is the geolocation service of Akamai, a content delivery network (CDN) with a strong worldwide presence. According to their facts & figures website Akamai "has deployed the most pervasive, highly-distributed CDN with approximately 325,000 servers in more than 135 countries and nearly 1,435 networks around the world" [31]. Due to this high coverage they claim to be 100% accurate for US cities and 97.22% accurate for other cities worldwide [32], this has also been verified by an independent third party [33].

Before we continue, let us first quickly take a look at how these more advanced methods work.

3.2.1. Private Data Feeds

RFC8805 [28] describes a comma separated file format for ISPs to publish IP ranges as well as geolocation data (i.e. [34, 35]). These files can be continuously polled to always receive up-to-date information. See listing 5 for an example.

```
$ cat egress-ip-ranges.csv | rg Enschede
2a02:26f7:f980:4a61::/64,NL,,Enschede,
2a02:26f7:f984:4a61::/64,NL,,Enschede,
2a02:26f7:f988:4a61::/64,NL,,Enschede,
```

Listing 5: Excerpt of Apple's geolocation feed (filtered on Enschede).

3.2.2. Constraint-Based Geolocation of Internet Hosts

Constraint-Based Geolocation (CBG) infers the geographic location of Internet hosts using multilateration. Multilateration in this case means that several distance measurements (ie. through ping) from known geographical positions are sent to one target. Because digital information flows through fiber at 2/3 of the speed of light, this will put an upper bound on the distance between endpoints. Stacking multiple of these upper bound distance circles on top of each other, will result in a constrained space within which the target has to find itself. This system works similarly to GPS.

3.2.3. Topology-Based Geolocation

Topology-Based Geolocation (TBG) works similarly to CBG, with the difference that network topology is taken into account too. A data packet on its way from one host to another over the Internet will pass several routers on its way. These routers are very likely from large ISPs and their real world physical location is known. Therefore, the delay measurements of CBG can be applied from these intermediate known routers, for a more fine-grained constraint mesh.

3.3. Accuracy

These techniques alone do not create a geolocation database of all IPs though. Several commercial enterprises have begun collecting geolocation data on IPs and started to sell access to these databases. We referenced some of these providers on the previous page. In 2011, the Center for Applied Internet Data Analysis (CAIDA) investigated the accuracy of commercially available geolocation databases [36]. In total they compared 7 different geolocation providers against each other. Interesting to note is, that CAIDA used RIR information as ground truth. This

does not need necessarily have to be accurate as we had previously shown. Furthermore, to determine the accuracy CAIDA has checked how many databases agree on a particular result, considering that this data does not have to be accurate either, there are a few shortcomings with this kind of approach. Nonetheless, most geolocation providers were within 95% of accuracy on a country level, and some even for 93% accurate on a city level, which does highlight the accuracy of these databases.

In summary: Several new techniques as well as provider-supplied data allow us to create more fine-grained and accurate geolocation. Provider feeds directly provide IP and location pairings. CBG and TBG are able to accurately locate a host due to creating a mesh of physical constraints, since the speed of light and therefore digital information has an upper speed limit with which it can propagate. Lastly we looked at a comparison study of geolocation databases. Most databases are able to accurately locate the physical location, on a country level as well as city level.

3.4. Detection of Geo-unblocking Attempts

So far we have explained in some detail how geolocation databases accumulate their data and achieve their accuracy. This is not enough though to answer **RQ1** (How do commercial video on-demand providers detect geo-unblocking attempts? 1.2.2).

Netflix for example is able to actively detect VPN connections [37, 38], but they do not do so on their own. As we mentioned in the introduction to this section, there are also geolocation databases which have additional metadata (such as proxy-usage and connection type) for each IP. GeoComply and Digital Element are two examples of companies who offer geolocation databases with proxy detection to streaming providers like Prime Video and Netflix [39, 40].

This is the simple, yet unsatisfying answer to the research question. Streaming providers rely on commercial black-boxes in a two-step process to distinguish between a customer connection with and without VPN/proxy. In the first step, the origin of the connection is determined with the help of the earlier explained methods and cross-referenced to a potential allow-list. For example, a US based online publication does not want to admit European users to its site due to GDPR compliance, thus they refuse any IP which is from European origin [41]. In the second step, after the region has been validated, the metadata on the IP should show if a geo-unblocking method like VPN/proxy has been used, to definitely permit or deny access.

It is only natural to ask the follow-up question: How do these commercial providers retrieve their additional metadata? Answering this question is outside the scope of this thesis, due to several factors we will briefly discuss before ending this section.

Firstly, how are these commercial companies able to make the distinction between an IP from a residential connection and from a hosting provider?

We can only guess, as there is no academic research on this topic. A major assumption we can make, is that most or even all IPs from major infrastructure providers (i.e. Google Cloud, Amazon Web Services, Microsoft Azure, etc.) are automatically flagged as hosting connection and thus by default are not able to unblock content. There are many other cloud/hosting providers though, so we think that the commercial geolocation providers manually aggregate their names from news reports, technical reports and other sources, to create a complete picture. At the same time this means, that small or local providers might not appear in these databases. In the case of VPN/proxy providers the answer might be as simple as purchasing subscriptions and then enumerating all IP addresses used by these services. On the other hand, the Tor project publishes all its exit-relay IPs [42].

Secondly, is it economically sensible achieve a high accuracy on the additional metadata fields?

Again, we may only guess, because the companies do not publish their earnings, but several factors point in this direction. There are the streaming providers who wish to inhibit the use of proxy technology, as we are discussing in this work. Additionally, knowing if a connection is from a proxy or not may help payment providers to detect fraud online. The combination of these stakeholders seem to also influence the pricing model of the commercial geolocation providers. They only operate on a business to business level, where the pricing structure is only available on request. (We have contacted one provider and have received a quote of $10 000 per month for 40 000 000 requests.)

Conclusion: The goal of this section was to answer **RQ1** (How do commercial video on-demand providers detect geo-unblocking attempts?) We first showed that geolocation data is difficult to gather, because there does not exist any authority with accurate ground truth. Then. we went on to show how advanced techniques may achieve a high geolocation accuracy. Finally, we arrived at the current state of geolocation, which relies on commercial companies using their proprietary methodologies to detect VPN/proxy connections.

4. Unblocking Methods

In the previous section we have learned that content providers rely on enhanced geolocation databases, to distinguish between different types of Internet connections, such as hosting and VPN providers or residential connections. Based on this, we can assume that the databases cover enough commercial VPN[2] and hosting providers, to make geo-unblocking a non-trivial matter. At the same time, some of the biggest commercial VPN providers actively advertise their unblocking capabilities. This dissonance will guide us while we explore the unblocking mechanisms. For example, we can assume that for the actual unblocking process the geo-unblocking provider will resort to using IP addresses which are not listed as hosting, VPN or proxy.

In order to actively investigate their methods, we now have to focus on the actual software supplied by the commercial VPN providers. Most of the providers support all kinds of desktop and mobile operating systems (OS). We have chosen to conduct our investigation on a Linux based OS (Debian), because this type of open source OS lends itself very well for technical analysis, should there be a need for it.

The structure of this section will resemble the exploratory approach taken to investigate the unblocking methods and will answer **RQ2** (Which methods are used by commercial VPN providers to perform geo-unblocking?).

The main contribution of this section is that we will present two distinct methods to retrieve IPs used by the commercial VPN providers to circumvent geo-blocking.

4.1. Initial Setup

Before we can begin measuring or investigating anything, we have to conduct the simplest (and possibly most joyful) experiment, so that we may exclude any provider who does not deliver on its promise to facilitate geo-unblocking, namely watching geo-unblocked content. We have followed the user guide for every provider to establish a VPN connection. In most cases this just consisted of running an installer, or downloading OpenVPN/Wireguard configuration files. Irrespective of the used method, authentication is done by a username and password combination or activation code. An exception to this are Wireguard configuration files, which use cryptographic keys for authentication only.

To find geo-blocked content, we have consulted the *unofficial Netflix online Global Search* (`https://unogs.com`). The maintainers of the site ask some of their users who are situated in different countries around the globe, to run a small Python script which will scrape all available Netflix titles for their respective region. The scraped data then is sorted by title and country,

[2]We would like to quickly remind the reader, that not every commercial VPN provider offers geo-unblocking.

and finally presented on the site as a searchable database. We consider the site to be trustworthy, firstly due to the methodology of collecting the data they present and secondly because a random sampling of movies/shows which are not available in the Netherlands, overlaps with our result of not seeing those titles on our own device. Conversely, when we connected to one of our geo-unblocking providers and select a supported unblocking region (i.e. Japan), we also saw the titles which were exclusive to Japan. Lastly, and most importantly, when trying to actually play back this content we were able to do so, without receiving an error message, such as [37] or [38]. We are exclusively using Netflix as a VOD provider, due to its leading position in the market, as well as being supported by all commercial VPN providers for geo-unblocking.

In summary: All chosen commercial VPN providers were able to geo-unblock content from different regions.

4.2. Plain Text Configuration Files

Running the VPN software lets us access geo-unblocked content. We might be tempted to suspect that the applications engage in "foul play", by modifying our system in undocumented ways, because these applications are precompiled binaries and closed source. We therefore propose to only resort to VPN configuration files if possible, instead of using the precompiled binaries. The provider *PrivateVPN* for example, does not even offer a compiled binary for Linux systems [43], making the use of configuration files mandatory.

Listing 7 shows the OpenVPN configuration file which we retrieved from the provider's setup guide. The first line defines the remote host, its port and protocol to use. In this case we would connect to one of their New York City (NYC) servers. At the time of writing, the hostname contains five *A record* entries, as seen in listing 6.

```
1 ;; QUESTION SECTION:
2 ;us-nyc.pvdata.host.            IN  A
3 ;; ANSWER SECTION:
4 us-nyc.pvdata.host.     300     IN  A   45.130.86.10
5 us-nyc.pvdata.host.     300     IN  A   45.130.86.3
6 us-nyc.pvdata.host.     300     IN  A   45.130.86.8
7 us-nyc.pvdata.host.     300     IN  A   45.130.86.5
8 us-nyc.pvdata.host.     300     IN  A   45.130.86.12
```

Listing 6: A records for the domain us-nyc.pvdata.host

This serves as a simple load-balancing mechanism, so that not all end-users connect to the same server. Similarly, if one of the servers is not available, for example due to DDoS or main-

tenance, then the program will just try the following server and continue this cycle until a connection has been established successfully.

Let us now look at the rest of the configuration (listing 7). **nobind** will let the IP stack dynamically allocate a port for returning packets. **dev tun** will encapsulate IP packets, whereas *dev tap* would send complete Ethernet 802.3 frames.

tun-ipv6 adds IPv6 compatibility. **remote-cert-tls server** verifies that the client connects to the actual VPN server, and not to another client, as both certificates are derived from the same certificate authority. Line 8, **client**, tells the client to accept configuration data which is sent by the server, such as the IP address, gateway and DNS server. Line 9 enables compression, if possible. Lines 10 and 11 are caching options. **verb 3** defines how verbose the logging output should be. The values 1 to 4 are defined as normal usage range.

Lines 15 to 16 specify the cryptographic primitives that are to be used by the client. Line 17 enables username and password authentication with the service.

The CA certificate option supplies the CA certificate, with which all certificates in use are signed. The clients receive the public part of the certificate, to verify the server's private key and vice versa.

And lastly **tls-auth**: It provides an additional layer of HMAC authentication on top of the TLS control channel to mitigate DoS attacks and attacks on the TLS stack, according to OpenVPN's documentation [44].

Reading this configuration file shows us, that many of the options are certificate management and security settings. Most notably, only one *tun* device gets created which will tunnel all traffic through the remote VPN server, leaving no doubt about the destination of our traffic.

We may even use a more condensed and easy to interpret configuration file. Listing 8 shows the Wireguard configuration file needed, to connect to the same NYC server from the same VPN provider.

Lines 1 to 4 setup the adapter, similarly to **dev tun** in OpenVPN. Instead of getting an address pushed from the server, it is explicitly defined in Line 3. This is due to Wireguard's implementation, which specifically mentions that IP management is not part of the Wireguard core, to keep the code base as small and easy to audit as possible. Also, the hard coded DNS settings are needed, because contrary to the **client** option in OpenVPN, we can not receive it through a push from the server.

We are left with two important options in line 8 and 9. Firstly the **AllowedIPs** option specifies which IP addresses are allowed to traverse through the Wireguard interface to the remote location.

```
remote us-nyc.pvdata.host 1194 udp
nobind
dev tun

# Options
tun-ipv6
remote-cert-tls server
client
comp-lzo
persist-key
persist-tun
verb 3

# Crypto
cipher AES-128-GCM
auth SHA256
auth-user-pass

# Cert
<ca>
-----BEGIN CERTIFICATE-----
MIIErTCCA5WgAwIBAgIJAPp3HmtYGCIOMA0GCSqGSIb3DQEBCwUAMIGVMQswCQYD
...
5g==
-----END CERTIFICATE-----
</ca>
<tls-auth>
-----BEGIN OpenVPN Static key V1-----
f035a3acaeffb5aedb5bc920bca26ca7
...
1477b537261cb56a958a4f490d961ecb
-----END OpenVPN Static key V1-----
</tls-auth>
key-direction 1
```

Listing 7: OpenVPN configuration file from PrivateVPN. The certificates have been shortened for readability.

The two entries `0.0.0.0/0` and `::/0` are catch-all rules, meaning all traffic will be tunneled. Lastly, similar to the **remote** option in OpenVPN, we still have to specify to which remote host we should connect. In this case it is again *us-nyc.pvdata.host*, but on port 3389 (usually used for Microsoft's RDP).

We would like to mention, that a public key infrastructure (PKI) as it is used in OpenVPN is not needed. There is no CA, which signs all keys and has to be present on every client, because in the Wireguard system, only a single public and private key are exchanged (line 2 and 7).

```
1 [Interface]
2 PrivateKey = sdv2e...krSo=
3 Address = 10.34.26.184/16
4 DNS = 10.35.53.1
5
6 [Peer]
7 PublicKey = W31EZ...34wQ=
8 AllowedIPs = 0.0.0.0/0, ::/0
9 Endpoint = us-nyc.pvdata.host:3389
```

Listing 8: Wireguard configuration file from PrivateVPN. The keys have been shortened to prohibit missuse.

We have repeated our previous random sampling of geo-blocked content using these plain text configuration files and were able to reproduce our findings, namely that geo-unblocking is still performed, even when not using the proprietary application.

In summary: The proprietary applications of the commercial VPN providers could have influenced our research by potentially introducing undefined behavior, which would not have been easily detectable due to the software being closed source. Therefore, we relied on using thoroughly tested open source VPN software like OpenVPN and Wireguard, for which plain text configuration files are often provided by commercial VPN providers. In all three cases (applications, OpenVPN, Wireguard), trying to geo-unblock content was successful.

4.3. DNS: Dat's Not Supposed to Happen

When connecting to a remote host, especially in an Internet measurement context, it may be prudent to not only look at the remote host, but also at the hosts along the way. From our earlier investigation into the plain text configuration files, we have concluded that the Internet traffic may only exit through one interface. We can therefore trace the route from our VPN interface, to our destination with the help of a network traceroute. In listing 9 we have performed such

a traceroute, from the university's premises to the host of netflix.com. It shows the traffic being handed over from the university, to Surf, the national research and education network of the Netherlands. Finally, the traffic gets handed over to Amazon. It may seem counter-intuitive, that traffic destined for Netflix may be handed off to Amazon, who runs a competing VOD service, Prime Video, but Amazon also offers a customer agnostic cloud infrastructure of which Netflix is a customer[45].

```
traceroute to netflix.com (54.170.196.176), 64 hops max, 72 byte packets
1  cr-ct-a.routing.utwente.nl (130.89.136.4)
2  130.89.254.201 (130.89.254.201)
3  e0-0-3-0.es001b-jnx-01.surf.net (145.145.4.45)
4  ae20.zl001a-jnx-01.surf.net (145.145.176.3)
5  * * *
6  ae41-0.asd002a-jnx-01.surf.net (145.145.0.196)
7  amazon-router.peer.surf.net (145.145.166.129)
8  * * *
```

Listing 9: ICMP traceroute from the University of Twente to Netflix. RTTs have been omitted for readability.

We may now repeat this experiment while being connected to one of the commercial VPN providers, for example NordVPN. The resulting traceroute can be seen in listing 9. It consists of a single hop, from the VPN interface on the VPN server, to the host *192.0.0.69*. Only having a single hop would technically mean, that the destination address is routed on the same interface as the VPN provider's server. Additionally, the IP subnet *192.0.0.0/24* is an IANA IPv4 Special Purpose Address Block, according to RFC 5736 [46]. Essentially this means, that this IP address range is not routed globally, thus it is not reachable from the Internet [47].

```
traceroute to netflix.com (192.0.0.69), 64 hops max, 72 byte packets
1  192.0.0.69 (192.0.0.69)
```

Listing 10: ICMP traceroute while connected to NordVPN to Netflix. RTTs have been omitted for readability.

Before we continue to reason about this result, we need to discuss an integral point when running a traceroute to a host, by supplying its domain name. The domain name needs to be resolved to an IP address, by the configured DNS server of the system. In the first traceroute 9, we used the university's DNS server for this purpose (130.89.2.5), whereas in the second traceroute 10, we used one of the two DNS servers supplied by NordVPN (103.86.96.100, 103.86.99.100).

(These two DNS servers were automatically configured on the system through OpenVPN.)

Only having a single hop to netflix.com and having the domain resolve to an IANA reserved block may already be seen as rather suspicious. We shall cast an even greater light on this finding, by showing that an AWS IP address should only ever be the only destination when resolving netflix.com.

To do so, we have used 100 RIPE Atlas probes, to resolve netflix.com and report their results back. The probes are spread all over the world and were instructed to use Cloudflare's DNS server (1.1.1.1). In total we received 199 IPs, 22 of those were unique, all IPs belonged to two distinct autonomous systems, except for a single IP. The two ASes are AS14618 and AS16509 and belong to Amazon. The single IP which does not belong to any AS is a non-routable IP address (10.10.34.35) and has been recorded from a probe in Iran. Due to Iran's censorship of western media, sites like Netflix are not available and DNS servers in Iran return a bogus IP, which redirects the user to a government controlled site.

From these two examples, NordVPN and Iran's censorship, we can see that returning manipulated DNS replies to our requests can be used to redirect Internet traffic to unexpected hosts. We have repeated the experiment with the remaining commercial VPN providers. Our findings were largely similar, in that most providers returned IPs not belonging to AWS. An exception is PrivateVPN, who does not return falsified DNS answers, but maintains a specific list of servers which have to be used should one want to make use of geo-unblocking.

In summary: Most commercial VPN providers we have investigated make use of their own DNS infrastructure, to return IP addresses under their control, instead of expected AWS IPs when querying for netflix.com.

4.4. Through the Looking-Glass, and What Alice Found There

Based on our previous findings, one might assume that the VPN providers are hosting a copy of Netflix on their own infrastructure. This claim however can be immediately refuted when looking at the SSL certificate of netflix.com. We have requested a SHA256 fingerprint of their certificate while not being connected via VPN. This can easily be done with the help of the openssl command line tools, because we can specify the host, either through hostname or IP and the actual site's name. Listing 11 shows this process while being connected at the university's network. We specify that we would like to connect to the host behind netflix.com on port 443 (SSL) and that the specific servername to retrieve is netflix.com. The reason why we specify netflix.com twice is as follows: Assume that a webhost is hosting two distinct sites on the same IP address. As a fictitious example we take example.com and example.tld, both hosted on 93.184.216.34. If we would only specify the IP address as our connect parameter (*-connect 93.184.216.34:443*), then the answer is potentially undefined, because the webserver

might choose to send back the SSL certificate of one of the two sites at random, or it might have a default site configured and return this one, or lastly, might not return a certificate at all. We therefore have to specify the servername independently of the hostname. Very often these two are the same, but as the later listings show, being verbose will increase the reproducibility of the experiments.

```
$ openssl s_client -connect netflix.com:443 -servername netflix.com < /dev/null
    2>/dev/null | openssl x509 -fingerprint -sha256 -noout -in /dev/stdin
SHA256 Fingerprint=FD:5A:01:92:AF:39:BA:BB:...:40:2D:91:11:43:71:7E:46
```

Listing 11: SHA256 fingerprint of Netflix's SSL certificate. The fingerprint has been shortened for readability.

Listing 12 repeats the earlier experiment, but with three randomly chosen AWS IPs, which were retrieved from the RIPE Atlas measurement.

```
$ openssl s_client -connect 54.155.178.5:443 -servername netflix.com < /dev/null
    2>/dev/null | openssl x509 -fingerprint -sha256 -noout -in /dev/stdin
SHA256 Fingerprint=FD:5A:01:92:AF:39:BA:BB:...:40:2D:91:11:43:71:7E:46

$ openssl s_client -connect 3.251.50.149:443 -servername netflix.com < /dev/null
    2>/dev/null | openssl x509 -fingerprint -sha256 -noout -in /dev/stdin
SHA256 Fingerprint=FD:5A:01:92:AF:39:BA:BB:...:40:2D:91:11:43:71:7E:46

$ openssl s_client -connect 18.200.8.190:443 -servername netflix.com < /dev/null
    2>/dev/null | openssl x509 -fingerprint -sha256 -noout -in /dev/stdin
SHA256 Fingerprint=FD:5A:01:92:AF:39:BA:BB:...:40:2D:91:11:43:71:7E:46
```

Listing 12: SHA256 fingerprint of Netflix's SSL certificate, while directly querying AWS IPs. The fingerprint has been shortened for readability.

It shows that all fingerprints match, because all certificates are copies of the public key, which belong to the private key in possession of Netflix. And, to show that the servername is indeed required, we have rerun the lookup without specifying the servername in listing 13. The fingerprint has changed, because the server is not aware that we would like to retrieve the fingerprint for netflix.com, so instead it sends back the certificate for its default configuration.

```
$ openssl s_client -connect 18.200.8.190:443 < /dev/null 2>/dev/null | openssl
    x509 -fingerprint -sha256 -noout -in /dev/stdin
SHA256 Fingerprint=68:42:DD:F0:7E:09:C9:57:...:B2:B2:AD:56:4B:1C:EB:43
```

Listing 13: SHA256 fingerprint of 18.200.8.190's SSL certificate, without specifying a servername. The fingerprint has been shortened for readability.

Finally, when running this experiment while connected to NordVPN's service (see listing 14), we retrieve the same fingerprint as shown in our earlier listings (see listing 11 and 12). If NordVPN, or any other commercial VPN provider for this matter, were to host a copy of Netflix on their own infrastructure, the implication would be that either the VPN providers are in possession of the SSL certificate's private key, or that they are cooperating with Netflix.

```
$ openssl s_client -connect 192.0.0.69:443 -servername netflix.com < /dev/null
    2>/dev/null | openssl x509 -fingerprint -sha256 -noout -in /dev/stdin
SHA256 Fingerprint=FD:5A:01:92:AF:39:BA:BB:...:40:2D:91:11:43:71:7E:46
```

Listing 14: SHA256 fingerprint of NordVPN's 192.0.0.69 host, when requesting Netflix's SSL certificate. The fingerprint has been shortened for readability.

We do not think that there is any merit to this theory beyond speculation, thus we will not entertain this train of thought any longer. This also means, that we can reject the idea of potentially making use of TLS termination proxies, because they too would require the use of the SSL certificate's private key.

If the commercial VPN providers cannot terminate the connection at their supplied host, because they lack the private key, the only remaining answer is that the traffic gets forwarded opaquely. A common name for this procedure is "SNI proxying" [48]. SNI stands for Server Name Indication, a field in the TLS "*Client Hello*" packet. The SNI field saves the same purpose as the *-servername* flag we discussed when using the openssl tool. The host knows which certificate to present to the client, based on the supplied server name, but in a SNI proxy the proxy knows to which host to forward the TLS traffic. Due to the opaque nature of this type of proxying, the client does not necessarily know through which networks his or her traffic will flow, before arriving at the true destination.

Looking back at the answer of our first research question, namely that VOD providers rely on commercial blocklists to detect geo-unblocking attempts, we can now start piecing together, how these blocklists are being circumvented. The solution can be found, if the connections through the TLS forwarding proxy could be traced. In the next subsection we will discuss several approaches to do so.

In summary: Commercial VPN providers with geo-unblocking abilities very often resolve DNS queries through a DNS server they control. These DNS servers then reply with a bogus host for Netflix.com, the bogus host again is under control of the commercial VPN provider and runs a TLS forwarding proxy, also known as SNI proxy. These proxies can forward TLS traffic through egress networks, which are not traceable for end-users.

4.5. Methods

All connections to the bogus netflix.com hosts terminate, as far as we can tell, at the IP address supplied by the DNS servers from the commercial VPN providers. From there the connection opaquely gets forwarded to the true netflix.com destination. This makes it impossible to use traceroute, to see the true route the packets take. Sometimes though, helpful metadata can be found in the headers of HTTP(S) connections, especially if these connections are made to content delivery networks (CDNs). Since all connections to and from Netflix are encrypted, we can use the web-debugging capabilities of web browsers, such as Firefox or Safari, to look at the unencrypted header data.

4.5.1. Netflix Headers

To look at the header information Netflix sends us, we navigate to netflix.com, log in, and request a video, while having the web-development console opened. More specifically, we will look at information from the "Network" tab, because other sources like HTML and CSS are not relevant for networking.

In the network tab, we can see connections to multiple Netflix affiliated domains, such as *nflxvideo.net* or *nflxso.net*. The actual video CDN seems to be located at *nflxvideo.net*, because while streaming only a few seconds of Netflix content, we have seen tens of megabytes transferred from this domain.

We would like to note at this point, that querying the commercial VPN provider's DNS for these domains, also results in the return of bogus hosts.

Selecting one of the requests, we may look at its content, or at the supplied HTTP headers. Listing 15 shows our outgoing request to one of the CDN endpoints. From our knowledge of the geolocation chapter, we can assume that this endpoint is hosted in Amsterdam, The Netherlands. The request furthermore shows details such as the resource we are trying to access, which kind of encoding our browser accepts, and our user-agent, a string of text identifying the vendor of our browser as well as operating system.

```
Request
GET /range/54199355-58364897 HTTP/1.1
Accept: */*
Origin: https://www.netflix.com
Accept-Encoding: gzip, deflate, br
Host: ipv4-c060-ams001-ix.1.oca.nflxvideo.net
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/605.1.15
    (KHTML, like Gecko) Version/15.1 Safari/605.1.15
Accept-Language: en-GB,en;q=0.9
Referer: https://www.netflix.com/
Connection: keep-alive
```

Listing 15: Request headers to *ipv4-c060-ams001-ix.1.oca.nflxvideo.net*, one of the sources of Netflix's video material.

Every request also results in a response, the exact contents of said response to our previous request can be seen in listing 16. It contains for example the HTTP status code 200, meaning that my request was successfully received, understood and accepted, as well as the time and date of my request. More importantly, a field we have not manually requested, called "X-TCP-Info" has been returned, containing an IP address. Specifically, the IP address is equivalent to the address of the workstation, connected to Netflix. We have repeated this procedure while connecting to different vantage points in the eduVPN system. EduVPN allows researchers to connect to participating national research and education networks (NREN) around the world.

Our first eduVPN connection is to Cyprus. Listing 17 shows that our request originates from 82.116.200.196, belonging to AS3268, the Cyprus Research and Academic Network (CYNET). And indeed, the reverse DNS entry for said IP also reads as *eduvpn.cynet.ac.cy*. The curious reader might have noticed, that even though our connection was originating from Cyprus, our Netflix requests were sent to a host presumably residing in Vienna, Austria. The explanation for this is not trivial, and we shall only offer a short explanation. Internet connectivity on islands can be a costly undertaking, especially if you plan to use high speed fiber connections. Since we are connected via GÉANT, the pan-European data network for NRENs, our Internet traffic is able to pass through uncongested fiber links, which appear to have a direct link to Vienna, via Athens.

Our second eduVPN connection is to South Africa. Listing 18 again shows the response to our request and the originating IP is displayed as 154.114.48.6 with reverse DNS *vpn-cpt-048-006.guest.eduvpn.ac.za*, belonging to AS2018, the Tertiary Education & Research Network of South Africa (TENET). Both hosts, the reverse DNS and Netflix's CDN contain the city code

cpt, which stands for Cape Town.

```
Response
HTTP/1.1 200 OK
Content-Type: application/octet-stream
Access-Control-Allow-Origin: *
Pragma: no-cache
Access-Control-Expose-Headers: X-TCP-Info
Timing-Allow-Origin: *
Cache-Control: no-store
Date: Tue, 09 Nov 2021 12:43:38 GMT
Content-Length: 3284126
Connection: keep-alive
Last-Modified: Sun, 24 Oct 2021 05:15:07 GMT
Server: nginx
X-TCP-Info: addr=130.89.190.225;port=58219;
    sc=TQwrFRYKckBwVHJASEAJcVpJZXFeX
    1QXR1Y7IxxAcFRtUHRbVwgaYEM8W2lJFBpzRUFaLQ==
```

Listing 16: Response headers from *ipv4-c060-ams001-ix.1.oca.nflxvideo.net*, one of the sources of Netflix's video material.

Our last eduVPN connection brings us to Pakistan. This time, an IPv6 has been returned: 2400:fc00:22::1015. The AS is identified as AS45773, belonging to the Pakistan Education & Research Network (PERN). Listing 19 provides the shortened response.

We can now reasonably assume, that the IP returned through the X-TCP-Info field, does indeed correspond to the IP used to send the request. Therefore, instead of being connected to eduVPN servers, we can rerun this experiment, while being connected to geo-unblocking commercial VPN providers. So far, we have also retrieved the headers through the web-developer console in the browser, a more automatable solution would be welcome. Instead of requesting actual video data from Netflix, we decided to try to retrieve the headers for the base domain such as *ipv4-c060-ams001-ix.1.oca.nflxvideo.net*. cURL is a small open source command line tool which can send and receive data over a multitude of protocols[49]. We make use of it due to its ability to receive HTTP headers in a simple fashion. Listing 20 shows the exact command pipeline.

```
Response
HTTP/1.1 200 OK
Content-Type: application/octet-stream
...
Server: nginx
X-TCP-Info: addr=82.116.200.196;port=49525;
...
```

Listing 17: Response headers from *ipv4-c045-vie001-ix.1.oca.nflxvideo.net*, shortened to preserve space.

```
Response
HTTP/1.1 200 OK
Content-Type: application/octet-stream
...
Server: nginx
X-TCP-Info: addr=154.114.48.6;port=49982;
...
```

Listing 18: Response headers from *ipv4-c001-cpt001-kznog-isp.1.oca.nflxvideo.net*, shortened to preserve space.

```
Response
HTTP/1.1 200 OK
Content-Type: application/octet-stream
...
Server: nginx
X-TCP-Info: addr=2400:fc00:22::1015;port=50437;
...
```

Listing 19: Response headers from *ipv6-c004-khi001-transworld-isp.1.oca.nflxvideo.net*, shortened to preserve space.

First, the base URL is requested with curl, any headers we received are filtered for the line containing the "X-TCP-Info" header. Filtering is done with *rg*, *ripgrep*, a Rust re-implementation of *grep*. Finally, all characters up until and including the first equal sign (=) and after as well as

including the first semicolon (;) are deleted, resulting in only an IP address as output. For this, *sed* is used. We also make use of icanhazip.com to verify our used IP address.

```
$ curl icanhazip.com -4
130.89.190.225

$ curl -sLIX GET https://ipv4-c116-atl001-ix.1.oca.nflxvideo.net | rg
    X-TCP-Info: | sed 's/;.*//; s/.*=//'
130.89.190.225

$ curl icanhazip.com
2001:67c:2564:318:4f8:bf94:fd7:5a7e

$ curl -sLIX GET https://ipv6-c116-atl001-ix.1.oca.nflxvideo.net | rg
    X-TCP-Info: | sed 's/;.*//; s/.*=//'
2001:67c:2564:318:4f8:bf94:fd7:5a7e
```

Listing 20: Filtered response from *ipv4-c060-ams001-ix.1.oca.nflxvideo.net*, one of the sources of Netflix's video material.

We shall quickly also explain the used cURL options, with the help of the cURL manual:

- **-s:** Silent or quiet mode. Don't show progress meter or error messages. Makes Curl mute. It will still output the data you ask for, potentially even to the terminal/stdout unless you redirect it.
- **-L:** If the server reports that the requested page has moved to a different location (indicated with a Location: header and a 3XX response code), this option will make curl redo the request on the new place.
- **-I:** Fetch the headers only!
- **-X GET:** Specifies a custom request method to use when communicating with the HTTP server. The specified request method will be used instead of the method otherwise used (which defaults to GET).

Let us now run one of these queries while being connected to a geo-unblocking commercial VPN provider, similar to the eduVPN experiments. In this first experiment (listing 21), we connect to a random Dutch NordVPN server. First we retrieve our external IP through the site icanhazip.com, and receive 159.48.55.108 which belongs to AS49981. AS49981 corresponds to WorldStream B.V. a known Dutch hosting provider. The second query, to Netflix's

CDN though, returns a different IP address. We are purposely not disclosing the exact IP address, because its AS as well as reverse DNS hint at it belonging to a residential user. The AS is AS50266, T-Mobile Netherlands, and the reverse DNS is [redacted]-187-31.ftth.glasoperator.nl. FTTH stands for "fiber to the home". We repeat the experiment on a different Dutch NordVPN server (listing 22). As expected we receive another IP address for the VPN server, this time 176.119.195.68, belonging to AS49453 also known as Global Layer B.V., yet again a known hosting provider. The actual request to Netflix though, seems to be sent through 217.101.0.0/16 (217-101-[redacted].cable.dynamic.v4.ziggo.nl) an IP address belonging to Dutch consumer ISP Ziggo B.V. (AS33915).

```
$ curl icanhazip.com
159.48.55.108

$ curl -sLIX GET https://ipv4-c116-atl001-ix.1.oca.nflxvideo.net | rg
    X-TCP-Info: | sed 's/;.*//; s/.*=//'
31.187.128.0/17
```

Listing 21: Responses from icanhazip.com and *ipv4-c116-atl001-ix.1.oca.nflxvideo.net*. The second response has been edited to preserve the IP user's privacy.

```
$ curl icanhazip.com
176.119.195.68

$ curl -sLIX GET https://ipv4-c116-atl001-ix.1.oca.nflxvideo.net | rg
    X-TCP-Info: | sed 's/;.*//; s/.*=//'
217.101.0.0/16
```

Listing 22: Responses from icanhazip.com and *ipv4-c116-atl001-ix.1.oca.nflxvideo.net*. The second response has been edited to preserve the IP user's privacy.

We repeat this procedure for another commercial VPN provider, Surfshark (listing 23). Again, we connect to a Dutch server and observe the following: icanhazip.com refuses our connection, but ifconfig.me can be used as an alternative. The returned IP for our VPN server is shown as 89.46.223.242 belonging to Hydra Communications Ltd. (AS25369) a hosting provider from the United Kingdom. Looking at our Netflix request, we receive 82.197.192.0/16 (82-197-[redacted].dsl.cambrium.nl). The owner of the address is Cambrium IT Services B.V. (AS25596), who in turn owns the brand "Tweak", again a Dutch consumer ISP. The sub-string *dsl* in the host name, also hints at a consumer connection. *DSL* stands for digital subscriber line, a family

of technologies to transmit digital data over telephone lines.

In all our measurements so far the IP address of the VPN server differed from the IP returned through Netflix's header. On top of the difference, the IPs returned through Netflix always seem to belong to residential Internet connections. The autonomous systems and reverse DNS entries very strongly support this notion.

```
$ curl icanhazip.com
curl: (7) Failed to connect to icanhazip.com port 80: Connection refused

$ curl ifconfig.me
89.46.223.242

$ curl -sLIX GET https://ipv4-c116-atl001-ix.1.oca.nflxvideo.net | rg
    X-TCP-Info: | sed 's/;.*//; s/.*=//'
82.197.192.0/19
```

Listing 23: Responses from icanhazip.com and *ipv4-c116-atl001-ix.1.oca.nflxvideo.net*. The second response has been edited to preserve the IP user's privacy.

In summary: Some commercial VPN providers who offer geo-unblocking use opaque TLS forwarding proxies, which halt any tracerouting attempts. Netflix's CDN though saves the IP address of every request in a response header which can be extracted easily. The recorded IP therefore allows us to see from which IP the request originated. This experiment has been manually run on two different commercial VPN providers (NordVPN and Surfshark). Preliminary results show, that the requesting IPs seem to be originating from Dutch consumer ISPs, whereas the IP addresses of the connected VPN servers belong to well-known hosting providers.

4.5.2. Akamai Headers

A similar feature is available on the CDN of Akamai, but it differs in that our request has to include a special header, which will return the requesting IP on response. The header is called "Pragma: akamai-x-get-client-ip". Listing 24 shows a regular header request of a site hosted on Akamai, in this case the VOD provider Disney+. Contrary to Netflix, the requesting IP is not automatically included, but as can be seen from listing 25, sending the header name in our initial request, will result in a response with our client's IP address.

We can easily rerun our measurement which has been used in the previous section, but this time targeting Disney+ through the commercial VPN providers. We use NordVPN again, as a first test and check if www.disneyplus.com resolves to a bogus host, which it does (192.0.0.56).

```
$ curl -sLIX GET https://www.disneyplus.com

HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Content-Length: 151792
Server: nginx/1.21.3
Content-Security-Policy: frame-ancestors 'self'
X-DNS-Prefetch-Control: off
Expect-CT: max-age=0
X-Frame-Options: DENY
Strict-Transport-Security: max-age=15552000; includeSubDomains
X-Download-Options: noopen
X-Content-Type-Options: nosniff
X-Permitted-Cross-Domain-Policies: none
Referrer-Policy: strict-origin-when-cross-origin
X-XSS-Protection: 0
Cache-Control: public, max-age=813
Date: Fri, 12 Nov 2021 09:40:07 GMT
Connection: keep-alive
Set-Cookie: optimizelyEndUserId=37c61cb8ba78000077368e61af030000c9030500;
    expires=Wed, 11-May-2022 09:40:07 GMT; path=/; domain=..disneyplus.com
```

Listing 24: Header response from www.disneyplus.com

```
$ curl -H "Pragma: akamai-x-get-client-ip" -sLIX GET https://www.disneyplus.com

HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Server: nginx/1.21.3
...
X-Akamai-Pragma-Client-IP: 130.89.190.225, 130.89.190.225
```

Listing 25: Header response from www.disneyplus.com including "Pragma: akamai-x-get-client-ip. Superflous headers have been cut for legibility.

Just as before, a random Dutch server has been chosen. Listing 26 shows that our VPN connection is using 178.239.167.175, an IP belonging to a hosting provider we have already mentioned earlier, Hydra Communications Ltd. (AS25369). The unblocking process seems to be facilitated through IPv6 this time, 2a02:a400::/25 belongs to KPN B.V. (AS1136). KPN is a Dutch landline and mobile telecommunications company.

```
$ curl icanhazip.com
178.239.167.175

$ curl -H "Pragma: akamai-x-get-client-ip" -sLIX GET https://www.disneyplus.com
    | rg X-Akamai-Pragma-Client-IP | sed 's/.* //'
2a02:a400::/25
```

Listing 26: Responses from icanhazip.com and https://www.disneyplus.com. The second response has been edited to preserve the IP user's privacy.

A repeated test of Surfshark results in a similar outcome. We noticed though that the TLS proxy is hosted on a globally routed IP address, 92.249.37.48 (AS209854) belonging to Surfshark Ltd., instead of a non-globally routed address, as is the case with NordVPN.

Requesting the headers results in receiving 94.212.0.0/14 (AS33915) as listing 27 shows. This IP belongs to Ziggo B.V., a residential ISP we previously already encountered.

```
$ curl ifconfig.me
89.46.223.21

$ curl -H "Pragma: akamai-x-get-client-ip" -sLIX GET https://www.disneyplus.com
    | rg X-Akamai-Pragma-Client-IP | sed 's/.* //'
94.212.0.0/14
```

Listing 27: Responses from ifconfig.me and https://www.disneyplus.com. The second response has been edited to preserve the IP user's privacy.

The geo-unblocking mechanism used for Disney+ resembles the one used for Netflix: residential Internet connections.

In summary: Disney+, a VOD provider similar to Netflix is also a target of geo-unblocking by commercial VPN providers. Contrary to Netflix though, its CDN does not automatically provide us with the IP of a requesting client. We therefore exploit the fact that Disney+ is hosted on Akamai, a CDN provider which lets us insert special purpose debug headers, such as "X-Akamai-Pragma-Client-IP", which will then return the requesting IP in the response headers. The egress host seems to be a residential connection.

4.5.3. SNI Manipulation

The last method we will cover, combines our earlier findings to create an approach that can work for VOD/CDN providers who do not supply IP information in their headers. Similarly to the commercial VPN providers, who send us bogus hosts for domains such as netflix.com, nflxvideo.net, disneyplus.com, etc. we now force a bogus host onto a domain under our control.

To conduct this experiment, we again make use of cURL, but this time we add the *–resolve* flag to our statement. This flag lets us replace the real IP of a domain, with a different one, such as a bogus host. Additionally, we have created our own HTTP header, that will return the IP of the requesting host if we query the domain under our control. We took inspiration from Akamai's "akamai-x-get-client-ip" and called our header "dacs-x-get-client-ip". Implementing this header was rather trivial. We make use of NGINX as a httpd daemon and only had to add a few lines to the configuration to support this custom header. Listing 28 shows the complete NGINX configuration. Our header additions can be seen on line 15 and 16. As you can also see from line 4, the domain we control is aptly called poisoned.dns.ph.

Now that the setup is complete, we will walk you through a complete cycle of this method. We begin by connecting to a commercial VPN provider and check if the provider sends bogus hosts for either Netflix or Disney+. In this example, we connect to a random US server from Surfshark. We resolve for netflix.com and receive a bogus host as expected, namely 138.199.42.165 (AS212238, Datacamp Limited). With this bogus host known, we can now construct our own query. All cURL parameters and other processing commands have been discussed earlier already, so we will not mention their role again. Listing 29 shows the complete query.

The returned IP 50.104.0.0/13 (50-105-[redacted].snpr.wi.frontiernet.net) belongs to AS5650, Frontier Communications of America. Frontier seems to be an ISP serving residential, small business and enterprise customers across the United States [50].

To compare our results to a previously established method, we run a query using our "Netflix header method". Listing 30 shows that our VPN server's IP differs from the IP address returned in the header. On the other hand, the returned IPs from both the Netflix header as well as the custom header match. The IP of the VPN server belongs to AS60068, another autonomous system belonging to Datacamp Limited. Datacamp Limited is owner of the brands CDN77 and Datapacket, which explains the reverse DNS of the VPN server's IP as well as the bogus host's IP: unn-84-17-35-102.cdn77.com and unn-138-199-42-165.datapacket.com.

```
server {
    listen 443 ssl;

    server_name poisoned.dns.ph;

    root /var/www/html;
    index index.nginx-debian.html;

    ssl_certificate /etc/letsencrypt/live/dns.ph/fullchain.pem;
    ssl_certificate_key /etc/letsencrypt/live/dns.ph/privkey.pem;
    ssl_ciphers         TLS_AES_256_GCM_SHA384:TLS_CHACHA20_POLY1305_SHA256:
        ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:
        ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305;
    ssl_protocols       TLSv1.3 TLSv1.2;

    location / {
        if ($http_pragma = "dacs-x-get-client-ip") {
            add_header "dacs-x-get-client-ip" $remote_addr;
        }
        try_files $uri $uri/ =404;
    }
}
```

Listing 28: NGINX configuration for https://poisoned.dns.ph

```
$ curl --resolve poisoned.dns.ph:443:138.199.42.165 -H "Pragma:
    dacs-x-get-client-ip" -sLIX GET https://poisoned.dns.ph | rg
    dacs-x-get-client-ip: | sed 's/.* //;'

50.104.0.0/13
```

Listing 29: Constructed query to force traffic through Surfshark's bogus host and the corresponding reply. The returned IP has been generalized.

```
$ curl ifconfig.me
84.17.35.102

$ curl -sLIX GET https://ipv4-c116-atl001-ix.1.oca.nflxvideo.net | rg
    X-TCP-Info: | sed 's/;.*//; s/.*=//'

50.104.0.0/13
```

Listing 30: Responses from ifconfig.me and *ipv4-c116-atl001-ix.1.oca.nflxvideo.net*.

This third and final method can also successfully retrieve the true IP that is used when geo-unblocking content, but there are two scenarios where this method can fail. In the first scenario the SNI proxy in use may filter incoming connections based on their SNI field. For example, the SNI proxy is configured with an allowlist of domains which may be forwarded. Connections to domains which are not on this list, will just be dropped. This also leads to the second scenario, namely that unknown domains will be forwarded through a generic egress host. This generic egress host may not be a part of active geo-unblocking, but rather act as a catch-all.

In summary: We can employ the same methods as the commercial VPN providers, by using their bogus hosts to forward traffic to a domain under our control. A major drawback of this method is that it may not work at all, if the SNI proxy does not allow packets that contain our domain to be forwarded. The geo-unblocking IP that seems to be in use can be assigned to an American ISP, because we used a US server to test for unblocking.

4.6. Concluding Remarks

In this section we have analyzed the geo-unblocking process from start to finish. We started by looking at the commercial VPN providers' VPN configuration files in order to be certain that they do not introduce any unexpected behavior. From there we went on to find out, that these providers replace the true IPs for domains such as netflix.com and disneyplus.com with bogus hosts under their control. These bogus hosts act as TLS forwarding proxies, also known as SNI proxies. Our connection traverses these proxies opaquely, thus it is not immediately clear through which hosts the traffic is egressing. We have contributed two distinct methods (headers and SNI manipulation), to identify how traffic to the VOD sites is forwarded. Based on these preliminary results, we can answer **RQ2** - Which methods are used by commercial VPN providers to perform geo-unblocking?

It seems that the commercial VPN providers are using residential proxies to circumvent detection by the VOD providers. More specifically, residential connections are of course not classified as hosting or similar in the enhanced geolocation databases, and thus they evade detection.

We now have the appropriate tools to reliably identify the unblocking methods and to answer **RQ3** (How do the geo-unblocking methods employed by commercial VPN providers scale to size?) in the following section.

5. Methodology & Results

In section 1 we discussed the size of the commercial VPN business and that a substantial amount of users (49% according to one survey) make use geo-unblocking. Two sections later, in section 3.4 we showed that enhanced geolocation databases provide additional meta information on IPs such as if they are being used for commercial VPNs or not. Throughout section 4 we saw that the commercial VPN providers route traffic meant for video on demand sites first through a TLS forwarding proxies and then through the actual geo-unblocking host. We were also able to retrieve the IP addresses of the geo-unblocking hosts we were routed through and these hosts seem to have been of residential ISP origin. Combining these findings, we hypothesize that the commercial VPN providers have a hidden infrastructure in the form of residential proxies to facilitate geo-unblocking. It is not known how vast these capacities are, so that the commercial VPN providers can provide their customers an uninterrupted viewing experience.

For this section, we have systematically deployed our measurement setup to all six providers we had acquired (CyberGhost, ExpressVPN, NordVPN, PrivateVPN, Surfshark and WeVPN) so that we may gain insight into this hidden infrastructure. Our measurements ran over the span of seven months and collected geo-unblocking hosts for 7 different regions. We also acquired a trial of Digital Element's NetAcuity, the enhanced geo-location database which is also used by Netflix, to verify that the unblocking IPs in use are not flagged as VPN or hosting endpoints.

With the results of our measurements we will be able to show the scale of the commercial VPN provider's hidden infrastructure and answer **RQ3** (How do the geo-unblocking methods employed by commercial VPN providers scale to size?) In this section we first discuss our measurement methodology (sec. 5.1) and then present our results (sec. 5.2).

5.1. Methodology

This subsection describes the criteria we used to arrive at our vantage point selection, as well as the details of our measurement infrastructure.

5.1.1. Vantage Point Subset Selection

For every commercial VPN provider we have to first determine a set of vantage points, from which we want to measure the geo-unblocking capabilities. To be considered as a vantage point, geo-unblocking capabilities have to be enabled for the location in question. The commercial VPN providers either list their geo-unblocking locations on their website, or within the application. One example of this can be seen in figure 1, which shows a screenshot of PrivateVPN's application. Only servers in the "Dedicated IP" category facilitate geo-unblocking (16

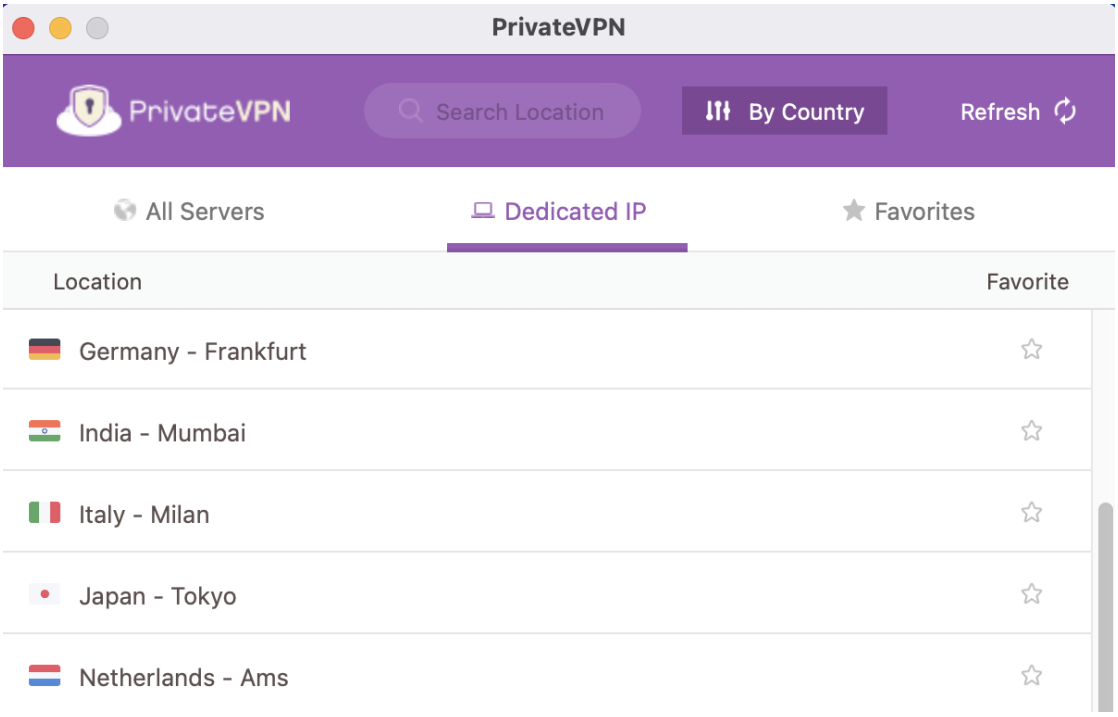


Figure 1: Partial overview of supported geo-unblocking regions from commercial VPN provider PrivateVPN.

distinct countries), even though PrivateVPN offers many more VPN locations (63 distinct countries). Therefore, our first criterion is that the selected VPN endpoint country must support geo-unblocking.

Not every commercial VPN provider facilitates geo-unblocking and those that do might not always service the same regions. As a concrete example we quickly look at Netflix unblocking at PrivateVPN and WeVPN. While both providers offer to unblock at their Japanese and US locations, PrivateVPN does not unblock in Austria, whereas WeVPN does. Popular unblocking regions (ie. Japan and US) are generally supported. Thus, if we want to compare providers with each other, there should be some overlap of geo-unblocking locations, making this our second criterion.

When we ran our preliminary measurements in section 4.5.1, we encountered Dutch ISPs as unblocking hosts. We suspect to encounter more of these and therefore also lean on a convenience sample. In other words, we are more familiar with Dutch providers than for example French and can possibly also get into contact with them, should it be necessary. Convenience sampling is our third criterion.

Random selection may also be used, to diversify the subset.

Lastly, many providers limit the amount of simultaneous connections per subscription. As a result, the sum of vantage points per provider should be lower or equal to the provider's limit.

5.1.2. Measurement Methodology

After having selected the measurement locations, we start a dedicated container environment for each vantage point. A containerized environment has multiple advantages, such as maintaining complete isolation between each VPN connection, portability and needing less resources than virtual machine setups. More technical details may be found in appendix A.

Within each container, we establish one VPN connection per vantage point. Once the connection has successfully been established, data collection can begin by using one or multiple of the methods presented in section 4, such as the Netflix Header from section 4.5.1 or the Akamai header from section 4.5.2. Once an IP has successfully been recorded, it is used as input in NetAcuity, to retrieve the meta information like AS number and name, as well as hosting/proxy flags. In some cases no IP address can be retrieved, this might be due to a timeout of the connection, or any other kind of error. That is why every measurement receives one of three status flags: Ok, Timeout and Connection Error. Because it is not clear how many hosts are used for unblocking, it is prudent to continuously retrieve hosts at a fixed interval, in order to traverse the set of possible unblocking hosts. Finally, a UNIX timestamp marks the exact measurement time and the results are written to a csv file to allow for easy processing later on.

5.2. Results

5.2.1. Overview

The first measurement was taken on Friday the 13th of November 2020 and the last measurement on Monday the 7th of June 2021. In total there were 6 351 216 measurements of which 5 606 902 (88.3%) were successful. Table 2 gives an overview of the other two statuses.

By following our vantage point selection procedure from section 5.1.1 we arrive at the following: We chose three major geo-unblocking regions, which are present on all six providers. These regions are New York, Frankfurt and Tokyo. We had one convenience sample, which was Amsterdam, also present on all providers with the exception of CyberGhost. As many commercial VPN providers limit the amount to around 5-7 simultaneous connections, we consider this our core selection. To still be able to debug or investigate, we kept one connection unused at all times, and thus also reduced the possible vantage points by one.

We applied random sampling to three additional regions: London, measured through Surfshark and PrivateVPN. Toronto measured through Surfshark, PrivateVPN and WeVPN. And lastly Zürich, only measured through PrivateVPN. Table 3 summarizes our selection graphically.

Only looking at successful measurements (Ok), we can produce a timeline for each provider, such as figure 2. In the course of the measurement, there were setback periods during which we were not able to collect data. For example, in November and December we worked on some intricacies of different containerization tools (Docker and Podman, see also appendix A.2). In early February, we presented preliminary results at ICT.OPEN 2021 (see appendix E), during which we were not able to maintain the measurement environment. Further measurement complications arose, when commercial VPN providers were running maintenance on their servers we were connected to, received DDoS, or general Internet outages occurred between our measurement setup and a vantage point. Manual intervention was needed to restart these hanging sessions.

Apart from just measuring connection status, we have also observed the connection and

Status	*n*	%
Ok	5 607k	88.3%
Connection Error	440k	6.9%
Timeout	304k	4.8%
Sum	6 351k	100%

Table 2: Measurements by status.

Provider \ Vantage Point							
CyberGhost	✓	✓	✓	✗	✗	✗	✗
ExpressVPN	✓	✓	✓	✓	✗	✗	✗
NordVPN	✓	✓	✓	✓	✗	✗	✗
PrivateVPN	✓	✓	✓	✓	✓	✓	✓
Surfshark	✓	✓	✓	✓	✓	✓	✗
WeVPN	✓	✓	✓	✓	✓	✗	✗

Table 3: Matrix showing the availability of measurements of a region per provider.

proxy type. Digital Element has decided to classify these types in the following way [51]: Connection types specify the originating nature of a host, whereas proxy type describes how the host is used. For example, the University of Twente is classified as an educational connection, and hosting providers unsurprisingly belong to the hosting category.

Table 4 shows the different types of connections we encountered during our measurements and table 5 shows their numerical appearance in our data set. Table 5 also shows that an overwhelming majority (98.30%) of measurements are of unknown or residential origin. This is an expected result, because anonymous proxies or commercial hosting providers are not able to geo-unblock, as we have discussed in section 3.4. Corporate and educational connections do allow for geo-unblocking, as contrary to anonymous proxies or hosting, these types of connections can not be purchased, with the intention of masking the user's location. Educational connections require the user to be enrolled in an academic program which in some cases can be associated with considerable costs. Alternatively the user may also be employed at the educational institute or corporation (for corporate connections). Taking the University of Twente as an example again which has been classified as educational host: streaming Netflix works with-

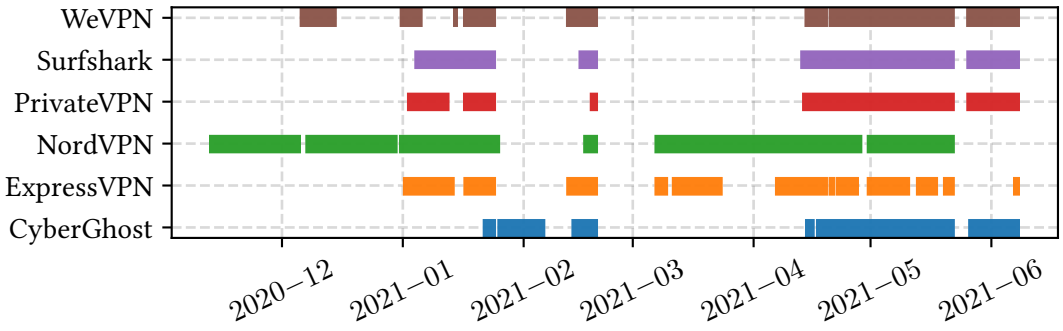


Figure 2: Timeline of successful measurements.

Connection Type	Definition
Unknown / Residential	Unknown or residential origin
Anonymous	Includes services that change location to beat DRM, Tor nodes, temporary proxies, and other masking services.
Hosting	Address belongs to a hosting facility and is likely to be a proxy as end users are not typically located in a hosting facility.
Education	Proxied users from an educational institution.
Corporate	Groups of users that are proxied through a central location or locations, and thus share a single network-apparent IP address.

Table 4: Connection type definitions.

out geo-unblocking warnings. We will discuss the implications of "unknown or residential" connections in more detail in section 5.2.2.

In the proxy category we encountered the types seen in table 6. Together with the connection type, this allows for a more fine-grained classification of a connection. For example, Microsoft's Azure is clearly a hosting provider, but at the same time it also gets called a cloud hosting provider, as they operate from several interconnected data centers, contrary to more classical hosting providers, with a single point of operations. A similar case can be made for Amazon's AWS and others.

Table 7 shows our measurement data for this category. Unsurprisingly almost all of our measurements belong to the unknown or no proxy group. Yet, we do still have other proxy categories present. Their appearances are rather limited except for the "VPN" classification. We will discuss this more in-depth in section 5.2.4, together with the overlapping IP addresses. We would like to mention that NetAcuity provides a few more connection origin types as well

Connection Type	*n*	%
Unknown / Residential	5 511.8k	98.30%
Hosting	91.6k	1.64%
Anonymous	1.7k	0.03%
Education	1.2k	0.02%
Corporate	0.6k	0.01%
Sum	5 606.9k	100%

Table 5: Measurements by connection type for all measured commercial VPN providers. Rounded to the nearest hundreds.

Proxy Type	Definition
Unknown / No proxy	Not known to be a proxy.
VPN	Virtual private network that encrypts and routes all traffic through the VPN server, including programs and applications.
Cloud-Security	A host accessing the internet via a web security and data protection cloud provider.
Cloud	Enables ubiquitous network access to a shared pool of configurable computing resources.
Tor-Relay	Receives traffic on the Tor network and passes it along. Also referred to as “routers”.

Table 6: Proxy type definitions.

as proxy types, but we only included those in our tables, which appeared in our data set.

Lastly, we counted 68 425 unique IPs *across all providers*, a majority of which were IPv4 addresses (66 073, 96.6%) and the rest being IPv6 (2 352, 3.4%). The IP addresses belong to 693 unique autonomous systems. 692 of those, cover all IPv4 addresses we observed and 4 ASes were responsible for the IPv6es. Three ASes were shared among IPv4 and IPv6: AS33915 - Vodafone Libertel B.V., AS40676 - Psychz Network and AS61317 - Digital Energy Technologies Ltd. A single AS (AS33659 - Comcast Cable Communications, LLC) was found to be IPv6 only.

Let us also approach these IPs from a different angle. If we sum all unique IPs *per provider*, we receive 68 782. In other words, there are 357 IPs which appear more than once in our provider set. More specifically: 11 IPs appear at three different providers (3.1%) and 346 IPs appear at two different providers (96.9%). We will discuss this more in section 5.2.4.

In summary: In the span of 7 months we acquired about 6.35 million measurement events spread across 7 regions. Of those regions, 4 were classified as core measurement regions (US, DE, JP, NL) and 3 were of an auxiliary nature (CA, UK, CH). In total there were 29 different vantage points (as shown in table 3). 5.6 million (88.3%) of all measurements were successful

Proxy Type	*n*	%
Unknown / No proxy	5 557 531	99.120%
VPN	48 810	0.870%
Cloud-Security	390	0.007%
Cloud	117	0.002%
Tor-Relay	54	0.001%
Sum	5 606 902	100%

Table 7: Measurements by proxy type for all measured commercial VPN providers.

and 98.3% of these connections are of residential or unknown origin. A similarly high percentage (99.12%) of successful connections are not classified as proxies either. 68 425 unique IPs were spread over 693 unique ASes. A majority of those IPs were IPv4 (96.6%) We will discuss the implications of unknown or residential connection types next.

5.2.2. Modes of Operation

We are assuming at this point that you, the reader, are wondering what kind of IP addresses are being used to facilitate geo-unblocking? Table 5 already showed that more than 98% of the connections are residential or can not be classified with Digital Element's NetAcuity. Based on a percentage this high, we hypothesize that the providers we measured are all using residential proxies to evade Netflix's VPN detection.

Figure 3 graphically represents the different types of connections used, per commercial VPN provider, to see if the connection types are evenly distributed per provider. From this overview, we spot three outliers, CyberGhost (92.9%) and ExpressVPN (94%) who both seem to make a bit less use of unclassified or residential ISPs and on the other side WeVPN (99.9%) who almost exclusively makes use of those connections.

Only looking at the connection types does not paint the whole picture though. In Section 5.2.1 we mentioned that we found 693 unique autonomous (AS) systems. The distribution of those ASes reveals operational differences between the commercial VPN providers. From Figure 4 we can see that CyberGhost, ExpressVPN and WeVPN employ between 40 and 80 different ASes, PrivateVPN makes use of only 6, Surfshark just above 200 and NordVPN just above 400.

Does the distribution of used IP addresses correlate to the number of used ASes? Figure 5

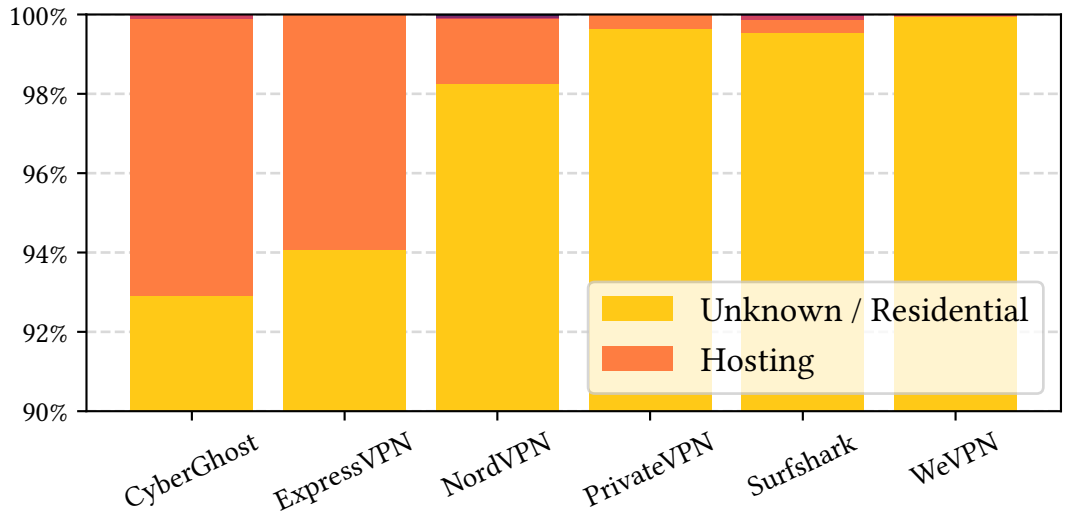


Figure 3: Connection types normalized per commercial VPN provider. Insignificant connection types (Anonymous, Educational, Corporate) are included, but not labeled. The y-axis begins at 90%.

shows that it does correlate, with NordVPN using just above 53 000 IPs and PrivateVPN using only 38 IPs.

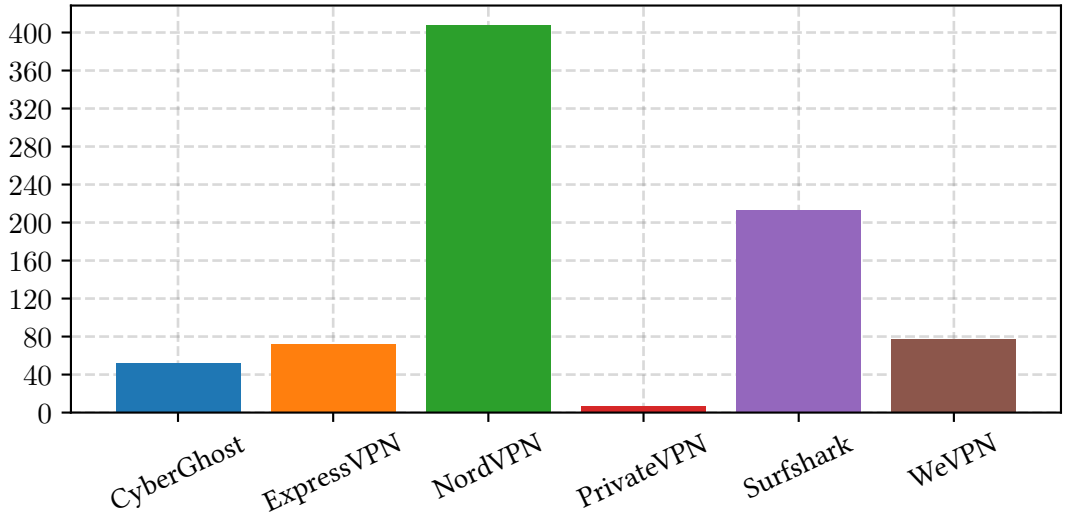


Figure 4: Number of unique autonomous systems used for geo-unblocking per commercial VPN provider.

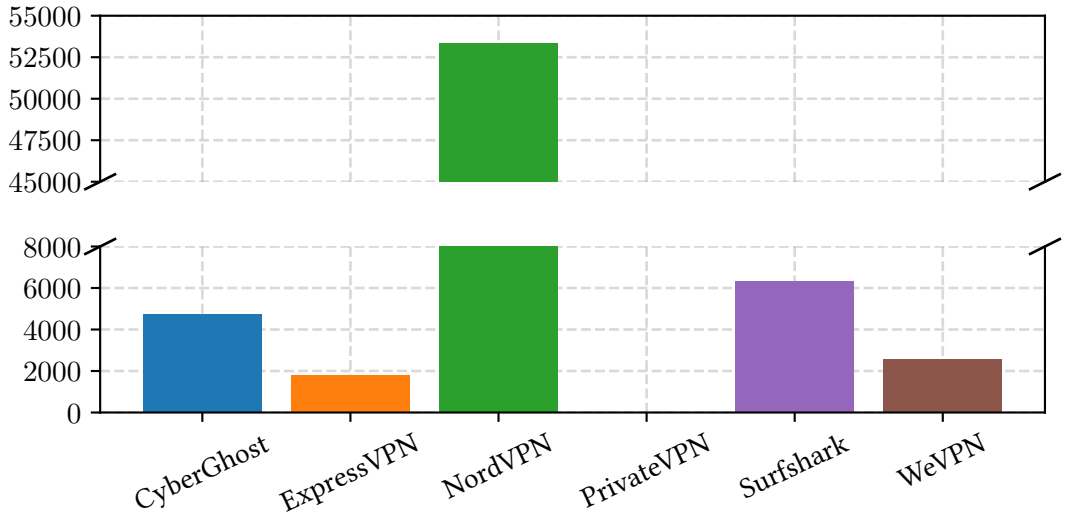


Figure 5: Number of unique IPs used for geo-unblocking per commercial VPN provider. The graph has been cut between 8000 and 45000 to highlight the difference between NordVPN and the other providers. PrivateVPN only used 38 IPs.

PrivateVPN is quite the outlier, which is why this provider will be discussed on its own.

This discussion can be found in Section 5.2.3. Let us now look at the other commercial VPN providers. Figure 6 shows the distribution of unique ASes per provider and vantage point region. From the distributions across each region, we can derive that some regions seem to be more contested than others. Take for example Japan, all commercial VPN providers except for PrivateVPN and WeVPN have been using at least several dozen different ASes, whereas in the US that number is between 1 and 9. But differences like this do not only appear between geo-unblocking regions, even between the providers themselves we can see different behavior on a per-region basis. Looking at WeVPN for example, we see that they use around 60 different ASes for geo-unblock in the Netherlands, but only a single one in Japan. ExpressVPN takes the opposite approach: around 70 ASes in Japan, but only 2 in the Netherlands. There does not seem to be a clear modus operandi or "standardized" way to provide geo-unblocking to the VPN provider's clients, instead every provider tailors their methods to their individual needs.

We do like to mention though that the two regions with the most amount of different ASes used (Japan and Netherlands) are also the regions in which we found the most ASes that have been flagged as hosting. Figure 7 shows the amount of unique ASes per commercial VPN provider and region, which have been classified as hosting in NetAcuity. Section 5.2.4 will look at hosting IPs a bit more in depth, because we require our conclusions from this as well as the following subsection, to understand the mechanisms at play.

With hosting ASes out of the way, we take a quick step back to the beginning of this subsec-

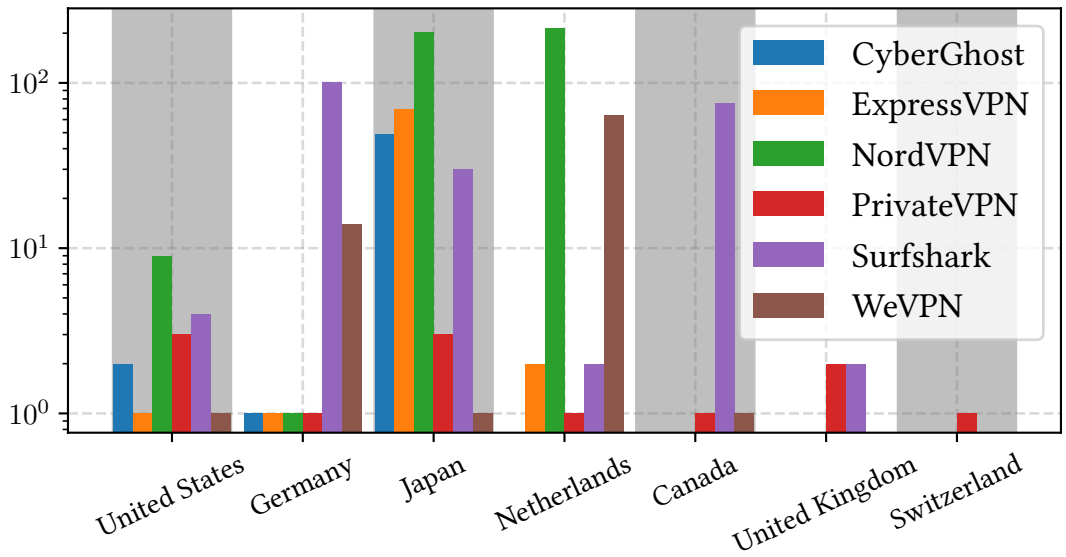


Figure 6: Number of unique AS per commercial VPN provider and vantage point region on a log scale. The shading differentiates between vantage point groups.

Country	AS	*n*
Germany	Deutsche Telekom (AS3320)	1338
Japan	NTT (AS4713)	5293
Netherlands	KPN B.V. (AS1136)	8636
United States	Metalink (AS13638)	1844

Table 8: Residential ISPs with the largest amount of unique IPs per core unblocking region.

tion, where we asked which kind of IPs are the commercial VPN providers using to geo-unblock content? From our initial findings in Section 4 it was already quite clear, that residential IPs play a part in this and our data proves this as well. Table 8 shows for each of our core regions, the residential ISP with the largest amount of unique IPs we found. In Section 5.2.4 we will return to the topic of residential IPs.

We end this section with a look at what is left over, once we remove the residential ISPs on top of the already removed hosting providers. We call this group "obscure hosting providers". In essence, these companies offer traditional data center services, like dedicated servers or VPSes, but they are not listed in Digital Element's NetAcuity as "hosting" company. Therefore, they can facilitate geo-unblocking at data center speeds, contrary to regular consumer ISP speeds. In Surfshark's US region, we encountered InterConnecx (AS13737) what seems to be a fairly

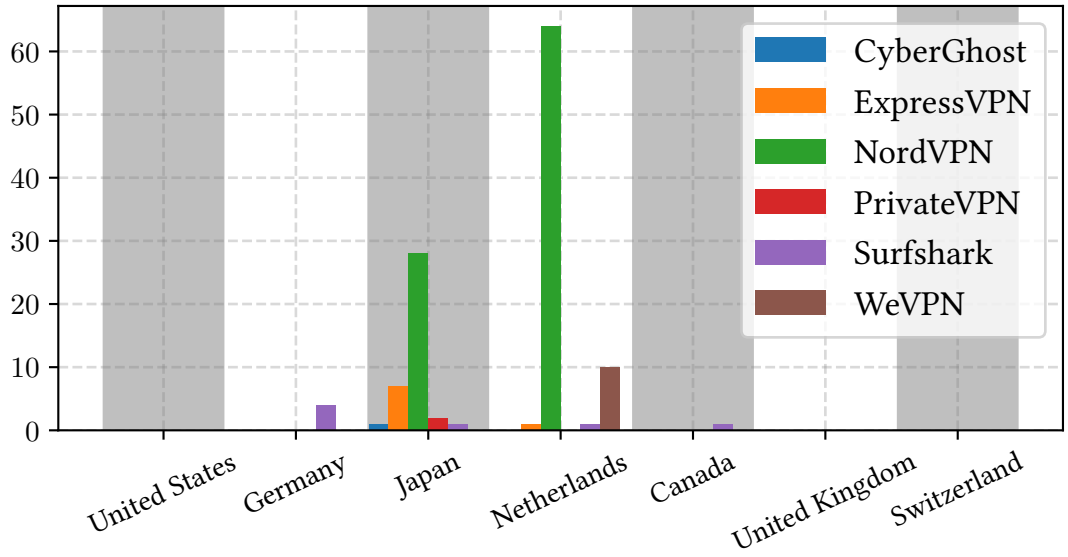


Figure 7: Number of unique ASes per commercial VPN provider and vantage point region. The ASes have been filtered by connection type "hosting". The shading differentiates between vantage point groups.

normal hosting provider, as their web presence is not out of the ordinary. This is in stark contrast to the next example: web2objects, LLC from NordVPN's US region.

web2objects' website (`http://www.web2objects.de`) is a simple HTML site, without a customer panel or any other kind of self-service checkout. The only information available is their services offering ("ISP, Carrier, Network provider, IT infrastructure in Europe and North America"), their German as well as American business address, a German fax number, as well as 4 email addresses (noc@ support@ abuse@ and legal@web2objects.com).

These two examples show, that enhanced geolocation databases are not infallible when it comes to detecting hosting providers, even though InterConnecx' website and offerings were similar to countless other hosting providers. web2objects' site on the other hand does not look like it has regular visitors or even customers, as it barely offers any information at all.

In summary: Even though all of our measured commercial VPN providers were using geo-unblocking mechanisms, there was no clear pattern visible. A provider like NordVPN used up to 400 different ASes, whereas CyberGhost for example only used around 40. In general this trend was also observable when looking at the IP addresses in use. The more ASes in use, the more unique IP addresses we found. We also checked if the usage pattern of ASes is the same for each provider for each region they unblock. This was not the case, as some providers use dozens ASes in a region where another provider only needs one. When these providers are then compared to yet another region, their usage pattern is completely flipped. The provider who previously only used one AS, is now using a dozen or more and vice versa for the other provider.

From our sampling of unblocking IPs in Section 4 we already saw that residential ISP connections are used for geo-unblocking. For every of our core countries (US, Germany, Japan, Netherlands) we found many IPs of the most common residential ISPs. But a closer look also revealed, that residential ISPs are not the only types of geo-unblockers in use. We discovered regular hosting providers that were not flagged as such in NetAcuity, meaning that can facilitate geo-unblocking with regular data center capabilities. Our findings ranged from providers with regular websites, to providers who only used an almost blank HTML page. We call this method of geo-unblocking "obscure hosting provider", because as the name implies, the hosting providers are so obscure, that they are not classified as hosting in NetAcuity.

5.2.3. Provider Deep Dive: PrivateVPN

PrivateVPN takes a special role, as the amount of unique ASes (6) and IP addresses (38) they utilize is far below the other providers. But not only that, they are also the only provider who did not use the TLS forwarding proxies. Instead they had a list of VPN servers to connect to, which do the geo-unblocking. Their two most used ASes are Telia Company AB (AS1299) and PVDataNet AB (AS42201). We first start with Telia Company AB (in short: Telia). Telia is a Swedish multinational telecommunications company operating mainly in the Nordic and Baltic countries. (We already shortly discussed Telia in section 3.1.) In our PrivateVPN data set though, we encountered their IPs at our vantage points in Switzerland, Germany, Netherlands, United Kingdom and United States. Since Telia does not offer any service in those regions under their own name or one of their consumer ISP subsidiaries [52], they could be providing transit connectivity through their global transit subsidiary Telia Carrier [53] in order for local companies to exploit their network for economic gain.

Consequently, we can see if any third party is registered in RIPE's IP database as responsible contact for the IPs we found. Table 9 lists an overview of all Telia and PVDataNet AB IPs from our data set.

The data shows that Telia is allowing two companies, Privat Kommunikation Sverige AB and Nordic Internet Services AB to use their IP addresses. Privat Kommunikation Sverige AB (company reg. 556895-1486), was the registered company name for PrivateVPN [54], before they changed it to PrivateVPN Global AB (company reg. 559282-2182) on January 1st 2021 [55]. The company registration of Privat Kommunikation Sverige AB reveals something else too, namely that the company was renamed PVDataNet AB [56].

Coincidentally this was the other provider with their own AS, also being used for geo-unblocking. PVDataNet has a dysfunctional web presence, where they advertise several different services, such as dedicated servers, IP transit or colocation [57]. On the bottom right of the website, they included links to their Facebook, Twitter, LinkedIn and GitHub presence. In reality though these links are empty and do not link to any presence at all. Similarly, they link to a map of their network which is "Coming Soon" since at least October 2020. This is the earliest entry we find for that page on `archive.org`. Contrary to regular providers, it is also not possible to purchase any of their services through a checkout system. Any kind of business inquiries are to be strictly handled via email. We have not attempted to contact PVDataNet, in order to keep our investigation non-intrusive.

Nordic Internet Service AB (Norisab) claims to be a "Swedish Internet, telephone and TV service provider for commercial and residential communities alike.", according to their website [58]. On their availability map they also list the UK, Germany, Netherlands, France, US and Denmark as territories where they offer Internet connectivity. We strongly doubt this claim,

as we have not encountered this ISP in the Netherlands or Germany to date. Furthermore, ordering their service is only possible through email or by phone, yet the telephone number they supply is located in the United Kingdom (+44) and not locally in Sweden (+46) or the Netherlands (+31). In RIPE's database their network operations center (NOC) entry is associated with a Swedish phone number and a P.O. box in the United Arab Emirates [59]. Similarly, to PVDataNet, business is to be conducted via email, as no checkout or shopping system exists. Also in this case we did not attempt any direct contact.

Neither PVDataNet AB nor Nordic Internet Service AB list their company registration number on their website which would allow us to gain some more insight through the Swedish companies registration office (Bolagsverket), but it is possible to search through the companies registry based on names. We found all three companies and with those entries also a list of board members. The VD (verkställande direktör or CEO) for PrivateVPN Global AB and for Nordic Internet Service AB is the same person, [redacted]. This person also acts as ordinary board member for PVDataNet AB.

PVDataNet AB and Nordic Internet Service AB are also both registered local Internet reg-

Country	AS	IP	RIPE Maintainer
Canada	PVDataNet AB (AS42201)	45.148.7.0/24	PVDataNet AB
Switzerland	Telia Company AB (AS1299)	217.212.245.0/24	Privat Kommunikation Sverige AB
Germany	Telia Company AB (AS1299)	193.104.198.0/24	Nordic Internet Service AB
		80.239.128.0/19	Privat Kommunikation Sverige AB
Netherlands	Telia Company AB (AS1299)	80.239.128.0/19	Privat Kommunikation Sverige AB
United Kingdom	Telia Company AB (AS1299)	193.104.198.0/24	Nordic Internet Service AB
		213.248.64.0/18	Privat Kommunikation Sverige AB
United States	Telia Company AB (AS1299)	193.104.198.0/24	Nordic Internet Service AB
	PVDataNet AB (AS42201)	45.130.86.0/24	PVDataNet AB

Table 9: Matrix of PVDataNet AB and Telia Company AB's ASes and IPs per vantage point and the corresponding maintainer according to RIPE. IPs are represented by their network prefix.

istries (LIR) with RIPE [60], meaning that they can get IP address ranges assigned to them by a RIR, or have their own AS. From this available information we conclude, that PrivateVPN is using shell companies under their control for two specific goals. Firstly, by being a registered LIR they can easily enter the commercial Internet transit market to buy transit and IP addresses in a region they would like to geo-unblock. And secondly, by creating the impression of being a consumer ISP, they evade classification by Digital Element, which in turn allows for geo-unblocking. We call this unblocking method "Fake ISP".

There are 4 other ASes PrivateVPN makes use of though, which we think are unrelated to the shell companies. Two of them (M247 Ltd. and Datacamp Ltd.) are flagged as hosting providers and are probably included on accident by PrivateVPN, as their appearance in our data set is only 0.18% and 0.17%. The remaining two ASes (Internet Initiative Japan Inc. and CenturyLink Communications, LLC) each facilitate geo-unblocking for a distinct region. The first for Japan and the second for the US. Because the unique IPs used for both regions are low (3 and 4 respectively), we doubt that their origin is residential. On the other hand we can not find proof that they are using a hosting construction like we previously uncovered. This leads us to believe that those might be enterprise class connections.

In summary: PrivateVPN only uses a handful of ASes (6) and IPs (38) for their geo-unblocking. Two of the ASes, Telia Company AB and PVDataNet AB are responsible for six geo-unblock regions. Telia is renting IP space to Privat Kommunikation Sverige AB and Nordic Internet Service AB. The Swedish business register reveals, that Privat Kommunikation Sverige AB is an outdated name, as the company renamed itself to PVDataNet AB. The relationship between these companies gets even clearer if one considers the director of Nordic Internet Service AB. This person also serves on the board of PVDataNet AB and is the director of PrivateVPN Global AB, PrivateVPN's legal name. PVDataNet AB and Nordic Internet Service AB claim to be regular (enterprise/residential) ISPs, in order to evade detection by Digital Element. We coined the term "Fake ISP" for this geo-unblocking method. From the remaining four ASes, two were probably included on accident, as their appearance in our data set is extremely limited (0.18% and 0.17%). The other two provide geo-unblocking for Japan and the US. We believe that those are enterprise connections and not residential, because only three IPs were used for unblocking in Japan and four for the US.

5.2.4. IP Overlap & IP sources

In this subsection we will take a look at IP overlap of geo-unblocking hosts between different commercial VPN providers. We will also hypothesize about where the providers might source their geo-unblocking IP addresses from. In Section 5.2.1 we already mentioned that we had multiple IPs appear at different providers: 11 IPs appear at three different providers and 346 IPs appear at two different providers. Figure 8 shows how all overlaps are distributed per region and per provider. Each bar represents the amount of duplicate IPs we measured, and the colors represent the regions those IPs are from. For example, the bar representing CyberGhost shows that there are about 100 IPs from Japan, that overlap with at least 1 other provider, but at most 2 other providers.

We immediately notice the strong presence of Japan and the Netherlands, whereas the other regions except for the United States are not present and even the US is only barely represented. We also see that the Netherlands appear at 2 providers substantially (NordVPN and WeVPN), but less at the others. From this, we strongly suspect that NordVPN and WeVPN at least partially share a source for their geo-unblocking IPs.

We prepared a matrix of the 11 triple provider appearances in Table 10. The table shows the country of the IP, its AS, as well as the commercial VPN provider overlap. Closer inspection reveals that all Dutch occurrences are from the same network prefix. Similarly to our provider deep dive, we gain a lot of valuable insight from analyzing outliers like this.

First we have to answer the question, who is Starry Network Limited? It is certainly not a

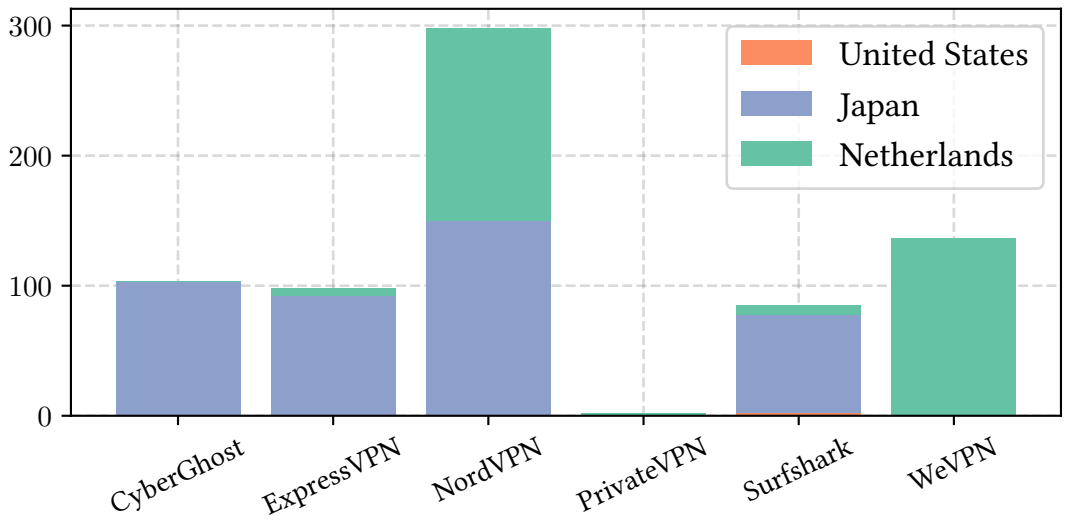


Figure 8: Unique IPs per region and per commercial VPN provider.

Country	AS	IP Address	Provider
Japan	Sony Network Communications Inc. (AS2527)	90.149.0.0/16	CG / Ex Nord
Japan	Softbank BB Corp. (AS17676)	60.141.0.0/16	CG / SS Nord
Japan	NTT Communications Corporation (AS4713)	223.216.0.0/14	CG / SS Nord
Netherlands	Starry Network Limited (AS134835)	185.170.209.0/24 185.170.209.0/24 185.170.209.0/24 185.170.209.0/24	SS / We Nord
Japan	Sony Network Communications Inc. (AS2527)	160.86.176.0/20	CG / Ex Nord
Japan	Sony Network Communications Inc. (AS2527)	131.147.0.0/16	CG / SS Nord
Japan	Softbank BB Corp. (AS17676)	126.234.0.0/16	CG / Ex Nord
Japan	KDDI CORPORATION (AS2516)	106.72.0.0/15	CG / SS Nord

Table 10: Overview of the 11 IPs which appear at 3 different commercial VPN providers. IPs are represented by their network prefix. Provider abbreviations: **C**yber**G**host, **Ex**pressVPN, **Nord**VPN, **S**urf**S**hark.

familiar Dutch consumer ISP, nor a well-known hosting provider. According to their website and APNIC registration, they seem to be a Hong Kong based hosting provider, offering servers in several countries around the world, but also in the Netherlands. Contrary to PrivateVPN's fake ISPs, we have an actual customer environment we can log in to and place orders. It is also noteworthy, that this provider has not been detected as a hosting provider by Digital Element, putting it into the obscure hosting provider category of geo-unblocking hosts. As a small reminder: obscure hosting providers offer traditional data center services, but for whatever reason they are classified as unknown / residential in Digital Element's NetAcuity. This allows them to facilitate geo-unblocking.

Figure 9 shows a timeline of all Starry Network IPs (four triple occurrences and two double occurrences, in total six) we encountered in our data set, and at which VPN provider. During several measurements we detected the IPs to be active at the same time at NordVPN and at Surfshark, but at WeVPN the IPs only appear in a very contained time frame, just before the beginning of May. We do not know what prompted this behavior, so we can only speculate. One explanation might be, that the VPN providers are colluding and exchanging geo-unblocking IPs based on some economic or other factor. If there is no collusion, then whoever is in control

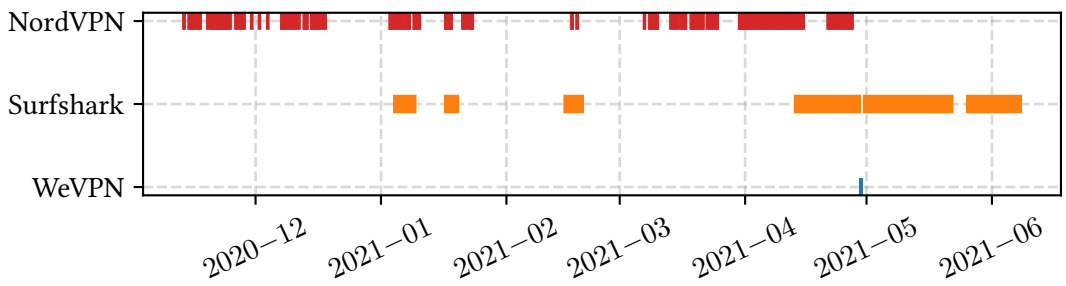


Figure 9: Timeline of Starry Network's overlaps at the different commercial VPN providers.

of the geo-unblocking hosts at Starry Network might have approached the commercial VPN providers directly to be included in the geo-unblocking pool. We would assume that there is some sort of compensation involved, and being in as many geo-unblocking pools (and therefore at different VPN providers) would surely maximize the compensation. If we take in the size of some of these commercial VPN providers and the amounts of IPs we discovered, we do not think that it is realistic that the providers individually manage all their geo-unblocking hosts, meaning there might be a third party responsible for geo-unblocking IPs.

We would now like to move on from Starry Network Limited and concretize the concept of this geo-unblocking third party a bit more. Towards the end of the research we were able to identify several actors, who advertised to sell consumer ISP (residential) bandwidth with the clear intention of using it for geo-unblocking purposes. These bandwidth providers work under the following premise: Residential ISP users install custom proxy software on their systems, because in return they will receive a compensation. As soon as traffic is proxied through their connection, they will be credited a certain amount of money for every gigabyte of data they proxy. One provider we found has a pay-out rate of USD 0.10 per GB of data proxied. If one wishes to make use of these proxies, the provider asks a price of USD 1.00 per GB. This leaves USD 0.90 of profit for each GB proxied, for the bandwidth reseller. We have only taken a cursory glance at these providers, but these amounts seem to be the going rate for providers in this business.

With this new knowledge we can go back to one of our earlier tables (Table 5), which showed the different types of connections that were used for geo-unblocking. Specifically we look at a type we have not investigated so far, educational connections. Table 11 shows which institutes we have encountered in our data set. To our surprise, even our own institute and NREN (SURFnet) are on the third and first position. We have contacted LISA, the network administrators of the University of Twente, with our results and asked if they can establish an anonymized communication channel between the users of the IPs we discovered and our

Institute	AS	n
SURFnet	1103	535
Universität Stuttgart	553	268
University of Twente	1133	138
Technische Universität Kaiserslautern	199578	91
Technische Universiteit Eindhoven	1161	78
Science Information Network (SINET) - Japan	2907	43
Keio University	38635	32
University of Tsukuba	37917	25
Kyoto University	2504	1

Table 11: Educational institutes by appearance.

research group. We only received some meta information from LISA, namely that all detected IPs do belong to students and not to employees, but nothing else because none of the students have accepted our communication request. Our contact letter can be found in Appendix D.

Our current working hypothesis is, that these students have installed the software to act as a geo-unblocking host on their laptops, for extra passive income. When they then carry these laptops from their homes to the university, they inadvertently use an IP of an educational instance which is what shows up in our measurements. But that might not be our most surprising measurement. In the prologue of this thesis we asked how the Dutch military got involved in this. Among all networks we observed, we also saw AS15466 which belongs to the Dutch ministry of defense. We immediately alerted their computer emergency response team (Defensie Cyber Security Centrum - DCSC) of our findings ahead of our publication, due to the sensitivity of their network. The letter can be found in Appendix C. We think that our message has been received in good faith, but we have not heard back. Our assumption regarding their IPs appearance is similar to the student's case: namely that some military personnel has installed the geo-unblocking software on their machine at home, but when it was brought inside the military complex it kept on unblocking using the ministry of defense's IPs. These two examples show that geo-unblocking bandwidth providers may be able to provide access to highly specialized networks for more than just geo-unblocking. For instance, to download almost any current research paper we only need to visit the publisher's website with an IP address of our university, no other form of authentication is required. Consequently, someone who buys "educational bandwidth" for USD 1.00 could download 1 GB worth of scientific papers, but would only pay a tiny fraction of the nominal cost. A rough calculation of the actual costs could look as follows: A randomly selected paper of about 9 pages with a few graphics and tables produced a 1.4 MB PDF file and the cost to download would be USD 14.95 for an IEEE member and USD 33.00 for non-members. 1.4 MB fits 714 times into 1 GB if we ignore the remainder,

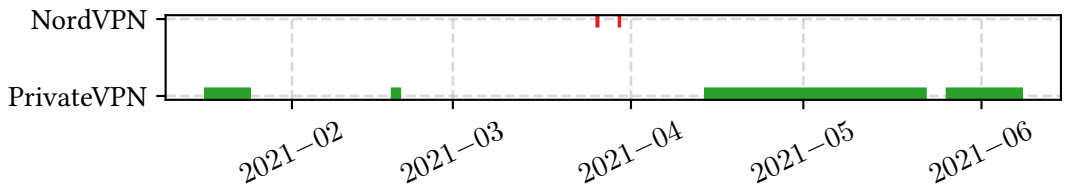


Figure 10: PrivateVPN IPs appearing at NordVPN.

thus 1 GB could result in about 714 downloaded papers. At IEEE member price this would be about USD 10 674 and at non-member price USD 23 562. Naturally there are discounts for bulk subscriptions, but it still shows the enormous potential savings when downloading them through the proxy system.

Before we end this subsection we would like to highlight one last case. We saw that the geo-unblocking proxy software keeps on working, even if it is running in non-typical networks, like the two we previously discussed. We therefore hypothesize, that the software also keeps on working, even if a VPN connection is active on the system. To explain why, we have to take a look at Table 7 where we saw that roughly 49 000 hits were detected as VPNs. We assume that some of those hits are geo-unblocking hosts that have turned on a VPN connection, meaning that instead of measuring the original geo-unblocking host's IP, we measure the VPN server's IP with which the host has connected. For example, we detected 185.34.136.0/23 in our measurement, which belongs to one of OctaneVPN's Dutch servers or 45.14.71.0/24 one of ProtonVPN's Japanese IPs.

Not all of those accidental tunnelings get flagged as "VPNs" by Digital Element though, as the following example will show. Imagine that the geo-unblocking host connects to a server from a geo-unblocking VPN provider such as PrivateVPN, who use dedicated geo-unblocking servers instead of TLS forwarding proxies. Because that server's connection type is "unknown / residential" in Digital Element's database, it will continue to facilitate geo-unblocking, but essentially through an unrelated party's systems. Let us look at an actual example involving NordVPN and PrivateVPN.

Figure 10 shows a timeline of 2 PrivateVPN IPs we were recording. These IPs belong to PrivateVPN's Dutch geo-unblocking servers, and we reliably encountered them while measuring PrivateVPN. Just before April though, these 2 IPs suddenly appear for a very short time at NordVPN. Because PrivateVPN is the owner of these IPs (as proven in Section 5.2.3) we can not explain their appearance at NordVPN in any other way than that the geo-unblocking host who should have geo-unblocked our connection, was actually VPNed to PrivateVPN's server.

At this point in time we do not have enough samples nor insight into these residential band-

width providers to draw further conclusions, for instance if a lack of Japanese bandwidth providers are responsible for the stronger overlap in Japanese IPs at the VPN providers. In any case, we see monitoring and measuring of this behavior as an invitation for future work, together with an investigation of these bandwidth providers.

In summary: During all of our 5 600 000 measurements, we found a small amount of IP overlap between the individual commercial VPN providers. Virtually all of these overlaps were concentrated on two distinct regions, the Netherlands and Japan. We took a look at one provider, Starry Network Limited (SNL), who appeared at three different commercial VPN providers in the same time frame. Because SNL is a hosting provider, we hypothesized that the IPs are under control of a single entity. If that single entity were a commercial VPN provider, the appearance of SNL's IPs at other VPN providers could hint at collusion between the VPNs. As this scenario is not realistic, we think that a third party might manage geo-unblocking IPs on a commercial basis. We found several actors online who claim to indeed offer these kinds of services at a certain price. With this new knowledge we took another look over our measured data and various indicators made the hypothesis of geo-unblocking as a service more likely. It might offer an explanation for the IPs we found from educational institutions, as this sort of bandwidth could not only geo-unblock television programs, but also allows for access to scientific publications, through the university's contracts. We also found IP addresses of other VPN services among our measured data, possibly because those geo-unblocking hosts themselves were VPNing to a different location. All in all we see this as an invitation for future work in the sphere of network access (residential, or educational) as a service.

5.2.5. Error Analysis

So far we have only looked at data which we gathered from successful measurements. But before we finish this section, we want to take a look at the unsuccessful measurements, like timeouts and connection errors. Table 2 has shown that we had 6.9% of connection errors and 4.8% of timeouts after all. Just as in our previous analyses we check if this behavior is consistent across commercial VPN providers. Table 12 shows all of our measured providers and their respective statuses. Cyberghost, PrivateVPN and WeVPN are close to, or above, 95% regarding their successful measurements, while Surfshark slightly lags behind with 87.4% followed by NordVPN with 80.8%. ExpressVPN comes in last, with 69.9%.

Creating a time series representation of the data, paints a bit of a different picture though. Figure 11 shows all measurement events for each provider, similarly to Figure 2, but the different colors represent the measurement status. Green stands for "Ok", yellow for "Timeout" and red represents "Connection Error".

From this graph we can see that the large amount of connection errors at both providers seem to have appeared in clusters. We may yet again get further insight by listing the connection statuses per measured region. Figure 12 shows this list for the ExpressVPN regions.

At ExpressVPN the connection errors appear in three different regions, Japan being unaffected. At NordVPN (Figure 13) the connection errors also appear in three different regions, but this time Germany is unaffected. Since these connection error clusters appear close in time at each provider we assume that the providers were changing their unblocking configuration. Our measurement client did not receive this updated configuration, as it has not reconnected

	CyberGhost		ExpressVPN		NordVPN	
Status	*n*	%	*n*	%	*n*	%
Ok	440k	94.4%	523k	69.9%	1374k	80.8%
Connection Error	11k	2.4%	188k	25.2%	201k	11.8%
Timeout	15k	3.2%	37k	4.9%	126k	7.4%
Sum	466k	100%	748k	100%	1701k	100%
	PrivateVPN		Surfshark		WeVPN	
Status	*n*	%	*n*	%	*n*	%
Ok	1170k	99.4%	843k	87.4%	1255k	97.1%
Connection Error	3k	0.3%	26k	2.7%	9k	0.7%
Timeout	3k	0.3%	95k	9.9%	28k	2.2%
Sum	1176k	100%	965k	100%	1292k	100%

Table 12: Measurements by status for each of the commercial VPN providers. All values are rounded to the nearest thousand.

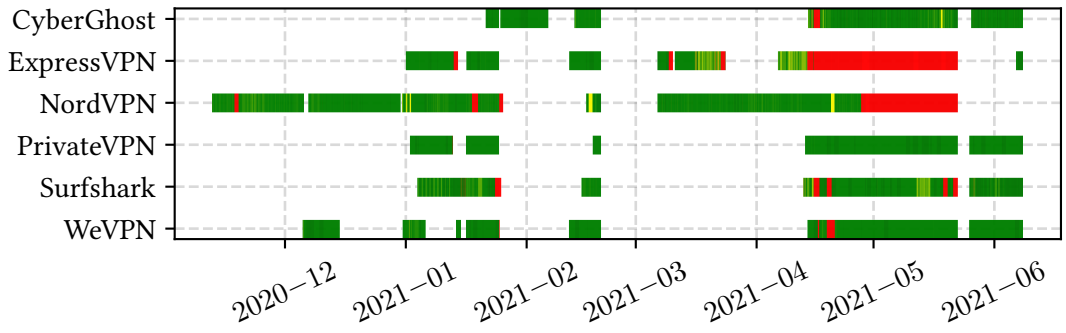


Figure 11: Timeline of all measurements statuses.

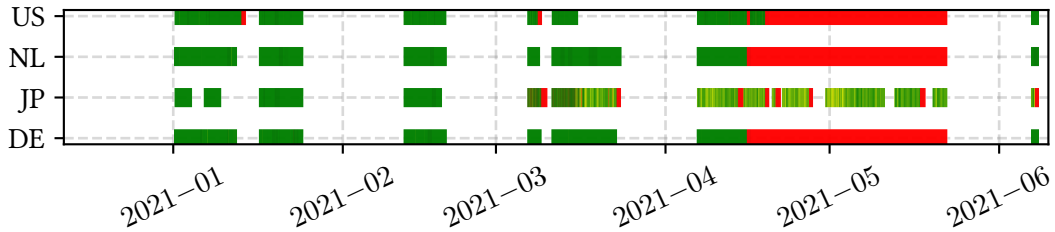


Figure 12: Timeline of ExpressVPN measurements.

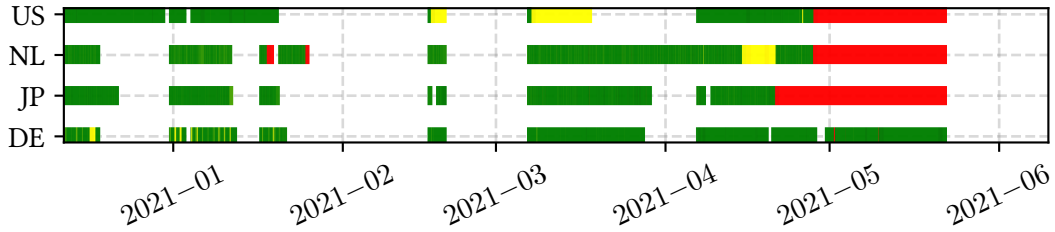


Figure 13: Timeline of NordVPN measurements.

since. We take this as a lesson learned, to not only make measurements more reliable in the future, but to also actively reconnect sessions, to receive the most up-to-date information from the commercial VPN provider in question.

In summary: Not only unstable connections may hinder measurements, stable connections may too. Specifically, because some of our measurement connections did not receive push updates when the providers changed their settings. Restarting connections regularly should help in always receiving the most up-to-date configurations.

6. Related Work

Researching commercial VPN providers is a relatively new direction in the Internet measurement community and most of it is privacy & security focused.

In 2015, Perta et al. have manually analyzed 14 providers on their privacy and security claims, and concluded that almost all providers are vulnerable to IPv6 leakage [61]. A followup study by Ikram et al., one year later, has extended the previous work, by analyzing 283 Android VPN apps [62]. Even though the Android apps benefit from a standardized networking interface, many of the apps came with embedded malware, ad-trackers, JavaScript injection, ad-redirections and even TLS interception. The work by Khan et al. from 2018 presented a more encompassing view of the commercial VPN ecosystem as a whole [9]. They do not only include the previously mentioned privacy & security issues, but also investigate the VPN providers' claims regarding the physical location of the VPN servers. In their results they show that 5-30% (depending on the geolocation database used) of all servers are located in a different country than what is advertised and in one extreme case a provider claimed to have 190 distinct locations, but ultimately only 10 different data centers were responsible for the hosting of the servers. A similar research has been conducted by Weinberg et al., who tried to verify advertised proxy locations with the help of geolocation [63]. They conclude that one third of all proxies are definitely not in the advertised location and another third might not be. A different study by Winter et al. tried to geolocate BGP prefixes, in order to better understand routing anomalies, outages and more [64]. One of their data points showed that a /23 network (512 IP addresses, 510 of them usable) geolocated to 127 different countries (including Vatican City and North Korea). This is of course highly unlikely and only after consulting WHOIS data, did it become clear that this was a commercial VPN provider owned IP range. [3]

All the previously mentioned studies highlight the usage of VPNs (or proxies) to circumvent geo-blocking, either in the abstract of the paper or in the body, yet to the best of our knowledge, there has not been any study so far, which investigated the unblocking methodologies of these providers. There has been a naive assumption, that through the sheer amount of servers operated, geo-unblocking can be facilitated [4]. Our contribution to this field presents new insights, by showing that this is not necessarily the case. We extend existing methodologies to research (commercial) VPN providers, by providing 2 distinct techniques to retrieve the IP address which connects to the content provider. In most cases these IP addresses are not the same as the IP address of the VPN server the user connects to.

[3] The provider in question is the same provider who claimed to operate 190 distinct locations from the Khan et al. study.

[4] Some commercial VPN providers claim to run between 2000 and 4000 servers [9]

7. Conclusion

At the start of this thesis we asked the broad question:

How do commercial VPN providers facilitate geo-unblocking?

Due to the opaqueness of the commercial VPN ecosystem, we did not have a a full body of research to base our work on. Instead, we used an exploratory approach in which we first identified a subset of commercial VPN providers, with methods similarly to other VPN research nonetheless. Furthermore, we derived three sub-questions to aid us in answers the research goal.

Our first research question centered around *How do commercial video on-demand providers detect geo-unblocking attempts?* In Section 3 we surveyed the available methods to differentiate IP addresses of customers who wish to stream content for consumption. Netflix, but also other VOD providers makes use of IP intelligence databases, that try to classify IP addresses in more categories than simple geolocation databases do (ie. Country/City). More advanced intelligence databases are also able to label an IP address by its use. For example, the label **edu** indicates that the IP belongs to an educational institution. **Hosting** indicates, that the IP very likely belongs to a data center and could therefore be used as a proxy. As a matter of fact, these enhanced gelocation databases even keep track of the proxy status of an IP address. In other words, not only does an IP receive the label **hosting**, it can also receive the label **vpn**, to make the IP's use even clearer. We found that Netflix and other VOD providers make use of these databases, in order to detect geo-unblocking attempts from regular VPN providers.

Now knowing how geo-unblocking is detected, namely based on classifications of the IP address in use, we worked on identifying the methods used by the commercial VPN providers to circumvent this IP address detection. In Section 4 we analyzed the configuration files we received from the VPN providers, in order to not be reliant on their closed source software. Also, the plain text configuration files let us see exactly what parameters are given, to start a VPN connection to one of the VPN servers. With our connections in place we resorted to basic tools like tracerouting, to get an initial idea of our connection to netflix.com. Instead of a regular traceroute though, we saw that our packets did not leave the VPN server as expected, but were redirected to a non-globally routable IP. We used the openssl library to confirm that our destination was still the legitimate netflix.com domain, by comparing the fingerprints of the SSL certificates we received with Netflix's certificates when we were not connected to any VPN. Since the certificates matched, we knew that we were connected to the legitimate Netflix website, but through some sort of TLS forwarding proxy. The proxy would forward our connection opaquely to netflix.com without using the VPN server's original IP address, therefore circumventing Netflix's IP detection. Due to the opaqueness of the forwarding, we cannot

trace any connection between the proxy and netflix.com, but we were able to deduce **2 distinct methods** to shed some light on the inner workings. **Method 1** consists of 2 variants: The first is that Netflix has included some debug information in the HTTP header of their video CDN, more specifically it contains the IP of the device making the request to Netflix's CDN. The second variant works similarly but for a different CDN, namely Akamai. Instead of receiving this debug information automatically, we need to send a custom request to Akamai, requesting to include the IP address in their response. **Method 2** makes use of the fundamental principle of the TLS forwarding proxy. The proxy is mostly agnostic when it forwards packets to unblock, and due to the TLS connection, it does not know the exact content of the packets. But, whenever a connection was made through the proxy for the first time, the proxy would be able to see the "Server Name Indicator" while it establishes the TLS connection. The SNI contains the requested domain in plain text, letting the proxy know where to forward the packets to. By crafting custom packets, we were able to establish a connection with the proxy, but with an SNI that pointed to a domain under our control. Naturally we were then also able measure which IPs connected to our domain. After having sampled these methods on a few commercial VPN providers, we detected that the IPs we measured were of **residential origin**, ie. **consumer ISPs**. This also lets us answer our second research question: *Which methods are used by commercial VPN providers to perform geo-unblocking?* The answer is that the providers are offering their customers a bogus host through their DNS resolvers, that forwards the TLS connection to Netflix or any other VOD provider through an opaque proxy system. Our findings thus far suggest that the proxy system makes use of residential IP addresses.

Residential Internet connections are mostly asynchronous, meaning they do not upload as fast as they can download and are also limited to a fraction of the speed of data center connections. Using residential IPs for unblocking therefore means, that many residential hosts are needed to be able to forward the same traffic a data center host could forward on its own. To get some impression of the commercial VPN provider's geo-unblocking network, we used our methods we developed in Section 4 to create a longitudinal measurement on all 6 commercial VPN providers we selected. The measurements ran over a time span of 7 months and resulted in about **6 351 000** measurement events of which **5 607 000** were successful (88.3%). In total, we recorded **68 425** unique IPs stemming from **693** unique autonomous systems and because we acquired a trial version of Digital Element's NetAcuity, we were able to categorize the measured IP addresses in the same way Netflix does. Since Netflix does not allow streaming via VPN or hosting providers, it was unsurprising that 98.3% of all measures IPs are categorized as residential or unknown ISP. Through fine-grained analysis of the measurement data we were able to identify more unblocking mechanisms than just residential IPs though. Hiding within the large group of residential or unknown ISPs we detected groups of unblocking IPs

that belong to hosting providers. These hosting providers either operate in a way to evade classification by Digital Element or are for other unknown reasons not listed in NetAcuity as a hosting provider. We therefore called this group of unblockers "**obscure hosting providers**". A third and last category of unblockers was found, when we analyzed PrivateVPN as a case study. PrivateVPN was the only provider on our list not using TLS forwarding proxies, but dedicated geo-unblocking servers. When analyzing these IPs, we noticed that they belonged to similar looking ISPs. Further research then revealed, that the director of PrivateVPN is also involved as a director or board member at the previously mentioned ISPs. These ISPs are made to look like residential ISPs and therefore evade Digital Element's classification. We call this last type of unblockers "**fake ISP**". So to answer our third and final research question: *How do the geo-unblocking methods employed by commercial VPN providers scale to size?* Commercial VPN providers make use of three distinct methods to evade detection by enhanced geolocation databases. Not only do they use **residential proxies**, they also make use of **obscure hosting providers** and if necessary, create **fake ISPs.**

7.1. Future Work

The research done in this thesis has only categorized geo-unblocking methods without delving deeper into the mechanisms of acquiring unblocking IP addresses. The two methods that concern themselves with using obscure or fake hosting providers are of little academic interest, as their relevance expires as soon as video on-demand providers or their IP intelligence services classify those IP ranges as hosting providers. On the other hand, the use of residential IPs poses a more difficult conundrum for video on demand providers. If they were to be blocked like IPs of hosting providers, then one would also exclude those honest customers whose Internet connection has been (ab)used.

During our analysis we discovered actors online who claimed that they were able to sell geo-unblocking bandwidth for a fee, and they did not hide their source either. Interested users could sign up to their service and install proxying software which would then be used for geo-unblocking or other purposes. For every gigabyte of data tunneled, the users would get paid by the bandwidth broker. We will take it upon us to continue the research into geo-unblocking and specialized bandwidth brokering in the future, after having discovered this new and unexplored avenue.

7.2. Closing Thoughts

The future of geo-unblocking as an industry might also just fade into irrelevancy. Several providers, for example Disney+ and Spotify, have adopted a proof-of-residency scheme to

counter geo-unblocking. This means that when payments for these video or music services are handled via credit card the payment processor of the VOD service will look at the BIN (a serial number which uniquely identifies the issuing bank as well as region), to confirm and lock the region to those identified. In very many cases acquiring a new method of payment requires the user to authenticate themselves to the issuing bank with legal documents like proof of residency of the specific country, which is a hurdle that the majority of geo-unblocking users will very likely not take.

A. Technical Appendix

This technical appendix will discuss the measurement setup in greater detail than the main body of the thesis does. On the one hand this will provide instructions regarding the reproducibility of the experiment, on the other hand open technical challenges will be highlighted. At the very core of a commercial VPN provider lies the promise to route all network traffic through one or multiple servers of the provider. This can either be achieved by downloading and running a software bundle the provider offers (ie. VPN apps on mobile operating systems, or all-in-one software solutions for desktop operating systems) or by downloading a configuration file for well-known VPN software, like OpenVPN or Wireguard.

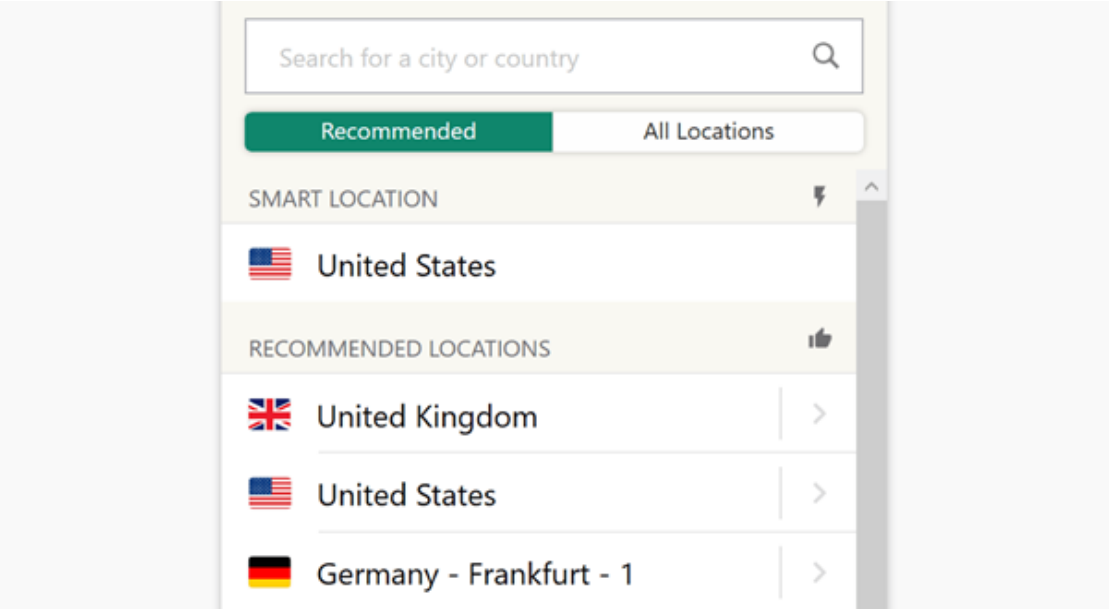


Figure 14: VPN-Provider's GUI to select a server from a pool of countries.

The first case very often offers an intuitive to use GUI: It lets the user choose a country or city to connect to, by displaying a list of all available locations. A single click then immediately connects to the selected location.

In the second case, users are expected to know how to setup as well as manually supply configuration files to the VPN software they want to use. While functionally there is no difference between the two methods, the latter one will very likely not be as aesthetically pleasing, nor will it offer the easy to access list of all available locations.

To make true on their promise of rerouting the network traffic via servers of the VPN provider, the network traffic of the device on which the software is running must be rerouted. This tautology may seem self-evident, but it highlights the first technical challenge: **Only a single VPN provider may change the network traffic route of a single device.**

A.1. Parallel Measurements

Every Internet connected device needs a network adapter to be able to exchange network packets. The operating system manages this adapter with the help of a routing table. In the presence of multiple network adapters, the operating system can be configured to egress traffic to certain hosts via the first adapter, and traffic destined for other hosts via the second. VPN software will add a virtual adapter to the device and configure the route to egress all available traffic towards the selected VPN server. Thus: programs competing for control over the routing will not produce well defined behavior, therefore every VPN connection (very often multiple connections per VPN provider) shall be contained to its own device. Luckily, these devices need not be physical (ie. a laptop or phone for every connection) but may also be virtual (virtual machine, virtual private server, containerized), because the VPN software does not have any physical dependencies. For this project we chose to containerize every single VPN connection. This choice is a result from several different factors.

- Firstly, economic: Buying a VPS for every single connection would incur unnecessary costs, especially due to the availability of compute resources at the university.
- Secondly, minimizing resource waste: Every VPN connection would require at least one vCPU as well as 256 – 512 MB of RAM.
- Thirdly, management overhead: Virtual Machines would have to be updated or reconfigured manually or with some management software like Ansible.

In contrast to the above, a single measurement container only uses about 75 MB of RAM and barely any CPU resources, as the container environment does not need to support a complete operating system. Containerization does not come without its issues. Even though every container has its own networking namespace, some essential networking capabilities must be explicitly “unlocked”, whereas others cannot be changed, seemingly by design. As an example, the capability to add VPN network adapters must be specifically enabled.

A.2. Arbitrary Limitation

Up to this point, we only mentioned containerization as a general term, but in this work, Red Hat’s “Podman” is the chosen implementation. Initial prototyping used “Docker”, but very

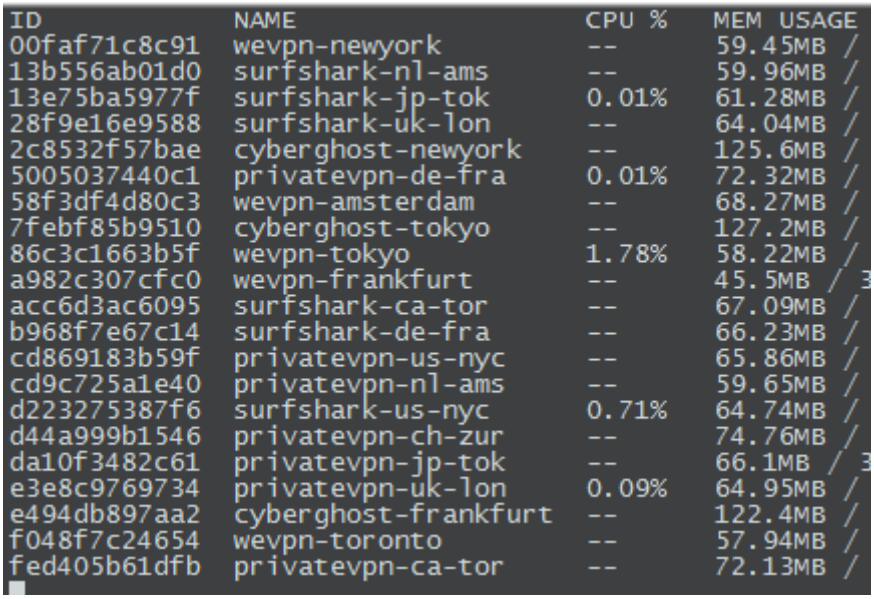


Figure 15: A list of running Podman instances, showing their CPU and memory usage

quickly limitations of the system led to the search of an alternative. We shall quickly recount the principal reason for choosing the alternative software here, as it fits the overall theme of "technical appendix" and would have been too distracting in the main thesis. All commercial VPN providers change the locally configured DNS server to a DNS server maintained by the VPN provider, for example from 1.1.1.1 [Cloudflare] to 10.35.35.53, a DNS server within the virtual private network. The rationale is two-fold: Changing the DNS server will not "leak" queries to third parties like Google or Cloudflare, a privacy guarantee by the VPN provider. More importantly though it enables the geo-unblocking feature. DNS queries which would ordinarily return the legitimate IP for a streaming site, will now answer with the IP for the TLS-forwarding proxy. To be able to change the DNS server on Linux, the operating system needs to have write permissions to "/etc/resolv.conf", more specifically only the root user or a set of networking services may do so. In the case of Docker, "resolv.conf" as well as a few additional files, are mounted into the containerized environment as immutable, essentially forbidding any changes. As a result, the VPN providers cannot update the local DNS resolver and therefore we cannot measure their geo-unblocking network. It is rather noteworthy, that Docker does not allow[5] any adjustments[6] to these mounts: Thus, using an alternative software is inevitable. This alternative comes in the form of "Podman". Free from artificial hindrances, we can launch a Podman container with the flag "–dns none" and no immutable file will be mounted. In this

[5] https://github.com/moby/moby/issues/1297
[6] https://github.com/moby/moby/issues/41229

state VPN tools or DNS management software such as “resolvconf” may freely edit the file.

A.3. Backups

Often overlooked but equally as often cherished, are the backups of critical files, configurations, and results. Our measurement containers run on a central European node, connected to a backup node at the university via SSH. Measurement results were duplicated in real time through the SSH connection, by means of a SSHFS. At first glance this looks beneficial, because at every point in time, there will be a mirrored copy of the measurement data. In hindsight, but also during the measurements some flaws have emerged which we will highlight here as well as ways to improve these flaws.

The SSHFS relied on a stable connection between the data measurement node and the data backup node. While this was the case most of the time, any kind of timeout between the two nodes would make the SSHFS become unresponsive. Because we mounted parts of the SSHFS into our podman containers, they too started hanging and had to be restarted manually.

A simpler solution would be to have a mandatory log rotate of the measurement data and then push complete logs to the backup server via rsync.

B. Ethical Committee Correspondence

During preliminary research of the geo-unblocking topic, we realized that some of the results could lead to personally identifiable information, such as reverse DNS records, residential IPs, personal websites, as well as sleep/wake patterns of potentially involved users. The research therefore did not purely qualify as a technical analysis, but also touched upon human subjects. In order to comply with data protection regulations we requested an evaluation from the EEMCS Ethics Committee[7]. The relevant mail exchange has been duplicated below and personally identifiable information of specific committee members has been blacked out. Approval was granted under the reference number **RP 2020-54**.

[7] https://www.utwente.nl/en/eemcs/research/ethics/

B.1. Letter to the Committee

```
To:    ethics-comm-eemcs@utwente.nl
Date:  14 May 2020, 11:48
Subject: Request for research proposal evaluation [DACS / geo-unblocking]
```

Dear ethics commission,
Dear legal department,

My name is Etienne Khan, and I am enrolled in the Cybersecurity Master. In this letter I would like to explain, as well as ask for permission to carry out my master thesis research project, which I am currently working on at the DACS group. A first meeting with Jeroen van der Ham has shown that this research project needs an evaluation by this board, which is why I am writing this letter. I will first talk about the project's background in general and then highlight some of my surprising findings. At the end I will present a table regarding the ethical and legal concerns.

Background

Not too long ago, Netflix (NF) was the legal go-to streaming service that took the world by storm. Fast-forward to today, there are many different services, some with worldwide coverage, some specific to local markets. Two examples are Amazon's Prime Video (worldwide) and BBC's iPlayer (UK only). Worldwide availability of a service though, does not mean that the available content is the same in every jurisdiction the streaming service operates in. An example for this is David Attenborough's famous documentary "Planet Earth II". While it is "on" Netflix, it is only available to NF viewers in the UK. Again, not too long ago, this "problem" was easily circumventable by renting a cheap virtual server in a country with an appealing content library. One would then VPN into that server and Netflix would treat you as a viewer from the same country as the one the virtual server is situated in. Currently it is not that straightforward anymore: At first Netflix started banning the individual IP addresses of well-known proxy/VPN services, from there on the bans started covering the whole Autonomous System (AS) of known server providers[8]. As more and more providers were banned from accessing the service, users from online communities[9,10] started to look for more obscure VPN/server

[8] https://blog.f-secure.com/why-is-netflix-blocking-vpn/
[9] https://old.reddit.com/r/NetflixByProxy/
[10] https://old.reddit.com/r/NetflixViaVPN/

providers, in the hope that those IP ranges were not banned yet. Anecdotal evidence seems to suggest that finding obscure and unbanned providers is still possible. So far, I have only superficially mentioned the commercial VPN providers, which are the focus of my research. During the time that the server providers were banned from accessing the streaming services, the commercial VPN providers split into two groups regarding this matter. A first group only offering their VPN services without any region unblock guarantees. The second group started adding unblocking as one of their core features[11,12,13]. My research details how the VPN providers are circumventing the bans by the streaming providers. Under regular circumstances this is not trivially possible, because we as researchers cannot inspect the complete network topology of a VPN provider, nor of a streaming service. Recently though, it has come to my attention that one streaming provider makes use of the content delivery network (CDN) Akamai. This CDN features a not well documented and not widely known development feature, namely returning the IP address of the client who requested the content within a HTTP header. Due to this feature, we can see which IP is detected at the streaming provider's server. After having used this development feature a few times, I have discovered that the VPN provider I have been using for this experiment, seems to be tunneling the streaming traffic through so called residential proxies, because the returned IP with the development feature, does not originate from the network the VPN server belongs to. As the name suggests, residential proxies are hosted not at a datacenter, but at a private person's home. If a streaming provider now looks up the connecting IP in a database, to determine if the connection comes from a legitimate user, or an unblocking service, the provider will be unable to make a distinction. I have presented and discussed my findings with several members of DACS, all of whom encouraged me to pursue this project. It follows in the footsteps of very young research, because commercial VPN providers have not yet been the target of large-scale academic research[14]. Additionally, a recent paper has shown[15], that an entire shadow economy is built upon the use of residential proxies, many of which were not voluntarily provided by the owners, but rather are compromised hosts. My research would be the first to combine these two avenues of research.

Results

[11] https://nordvpn.com/unblock/
[12] https://www.cyberghostvpn.com/en_US/unblock-streaming
[13] https://www.expressvpn.com/vpn-service#unblock-websites
[14] https://dl.acm.org/doi/10.1145/3278532.3278570
[15] https://ieeexplore.ieee.org/document/8835239

Thus far the background of my research. At the time of writing this letter it is not entirely clear if the found residential IPs really belong to private individuals or not, several signs do point to this conclusion though. I have looked up some of the results I found in IP databases such as Shodan. While not every IP gave me a result, I did find some that indicated that the IP in question is in use by a private person. The results were a personal website, as well as the RIPE entries claiming that the IPs belong to Dutch ISPs Vodafone, KPN, Tele2 and Caiway. While this alone is not compelling evidence, I have so far only checked roughly 25 distinct IP addresses. Another measurement I have done is, gathering IP addresses in a 24-hour timeframe, and then plotting their occurrence. As can be seen a few of them cease to reply between roughly midnight and 10 am CET. This somewhat correlates with a sleeping pattern of a human being. It should be also noted though, that some IPs are available continually, while yet again others reply erratically. At this point, it is not entirely clear if these really are internet connections of private individuals and if so, if they participate in this scheme willingly. Working under the assumption that they are not participating willingly, any kind of high-bandwidth measurement could negatively impact their internet connectivity.

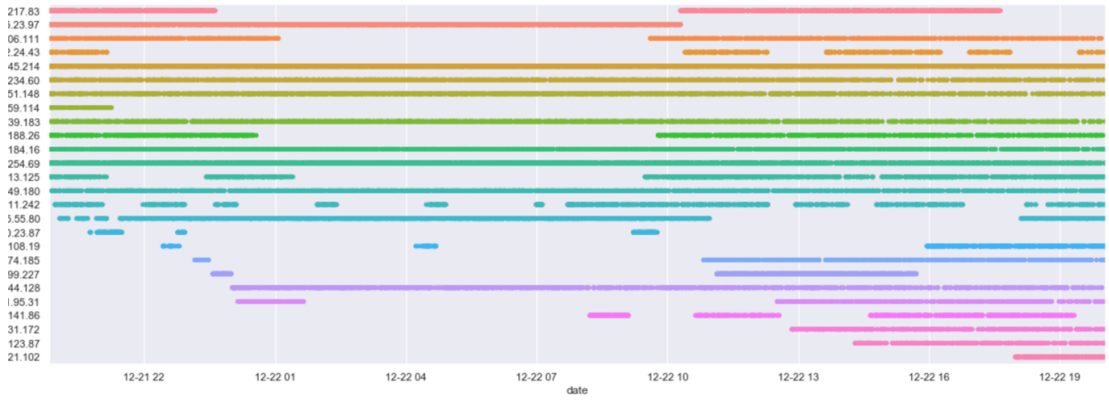


Figure 16: Plot of IP addresses appearing in a 24 hour time span

Additionally, while researching this topic, I have found a bug in one of the VPN's unblocking scheme, allowing me to send arbitrary HTTP(S) requests through the assumed home connection of a private individual. At the end of the research a responsible disclosure will be submitted to the affected parties. I will now list the table with identified issues, as well as steps I can take to minimize them. Most of the legal impact statements are left blank, I am hoping that the commission can fill them in for me. Goal I have already re-run this experiment on several

commercial VPN providers, to find out of this is a one-time phenomenon or an institutionalized issue. To my dismay I must report that as of writing, several commercial providers exhibit the same behavior. With my research I am aiming to continue the novel direction of research that Khan et al. in their 2018 IMC paper have begun, which is measuring the claims of commercial VPN providers. This research would solely focus on the question: How do commercial VPN providers circumvent the geographic access control measures put in place by subscription-based video on-demand streaming services?

B.2. Reply from the Committee

From: ethics-comm-eemcs@utwente.nl
Date: 16 Jun 2020, 11:56
Subject: Request for research proposal evaluation [DACS / geo-unblocking]

Dear Etienne,

The ethics committee will send a positive review to the dean concerning your research. For future references you can use the reference number RP 2020-54.

[redacted][16] Underneath the reactions from our ethics advisors, and attached the reaction from our data protection officer.

Good luck with your research!

Kind regards,

[redacted]

Secretary Ethics Committee EEMCS

[redacted] | Research Support Specialist | Faculty of Electrical Engineering, Mathematics And Computer Science | University of Twente | [redacted] | [redacted] |

[16] Personally identifiable information has been redacted.

C. Dutch Ministry of Defense

During regular monitoring of the accumulating research data we noticed the appearance of an autonomous system, related to Dutch military infrastructure. The nature of this communication network lead us to immediately disclose our findings to the network operators.

C.1. Letter to the MinDef

To: dcsc@mindef.nl
Date: [redacted]
Subject: I detected suspicious activity from DTO IPs during my university research

Dear recipient of this email,

My name is Etienne Khan, and I am a researcher at the DACS group of the University of Twente. During my research I have detected suspicious behavior from IPs in your network. With this mail I would like to explain what I am doing and what I have detected.

My research focuses on commercial VPN providers (ie. NordVPN, ExpressVPN, etc.) and their geo-unblocking feature. These providers usually define geo-unblocking, as a method to get access to, for example, the American Netflix library, even though you do not live in the United States. Maybe you have heard about this before?

During my research I have found a method to measure how these providers are "unblocking" foreign streaming libraries all around the world. During these measurements I have identified two distinct IPs from your network providing access to the Dutch version of Netflix.

I am currently not sure if the proxying software is part of a malware or willingly installed by the user, but I have identified at least 150 different Dutch networks, from where proxy traffic can be measured. Considering the nature of your network, I am now disclosing my findings regarding your network to you ahead of my publication.

The Unix timestamps and IPs are:

1. [redacted],Ok,[redacted],defensie telematica organisatie,15466
2. [redacted],Ok,[redacted],defensie telematica organisatie,15466
3. [redacted],Ok,[redacted],defensie telematica organisatie,15466
4. [redacted],Ok,[redacted],defensie telematica organisatie,15466
5. [redacted],Ok,[redacted],defensie telematica organisatie,15466
6. [redacted],Ok,[redacted],defensie telematica organisatie,15466
7. [redacted],Ok,[redacted],defensie telematica organisatie,15466
8. [redacted],Ok,[redacted],defensie telematica organisatie,15466
9. [redacted],Ok,[redacted],defensie telematica organisatie,15466

Thank you for your attention, should you have any other kind of question or feedback, you can contact my supervisors (dr. Anna Sperotto & dr. Roland van Rijswijk-Deij `https://www.utwente.nl/en/eemcs/dacs/staff/`), or me directly.

Kind regards,

Etienne Khan

D. Letter to identified geo-unblocking hosts at the University of Twente

To: [redacted]@student.utwente.nl
Date: 1 Dec 2020, 14:12
Subject: Request to participate in a VPN research

Dear recipient of this email,

My name is Etienne Khan, and I am a researcher at the DACS group of the University of Twente. In this email I would like to explain to you, why you are being contacted by LISA, the ICT department of the university and my involvement by means of this letter. My research focuses on commercial VPN providers (ie. NordVPN, ExpressVPN, etc.) and their geo-unblocking feature. These providers usually define geo-unblocking, as a method to get access to, for example, the American Netflix library, even though you do not live in the United States. Maybe you have heard about this before? During my research I have found a method to measure how these providers are "unblocking" foreign streaming libraries all around the world. Without going into too much detail, it appears that these providers are routing the VPN customer's traffic through the internet connection of an ordinary person, such as yourself, to mislead the streaming provider (ie. Netflix, Disney+, etc.). I am using the word 'mislead' here, because usually trying to use a streaming provider with a VPN connection enabled, will lead to error messages, prompting the user to disable the VPN. So how does all of this relate to you? During my measurements, I have come across IP addresses from our university, which have been used to unblock Netflix. After consulting my supervisors (dr. Anna Sperotto & dr. Roland van Rijswijk-Deij) we have decided to ask the university's ICT department for help in contacting the user of the IP address, in a privacy preserving way. LISA is reaching out to you because an IP address that was assigned to you appears to have been used as an exit point for a streaming unblocker. It would help my research greatly, if you would be willing to help me by answering a few questions about the unblocking (see below). I note that the unblocking itself is not prohibited in any way, the sole purpose of reaching out is because we want to learn more about how these unblocking services work. This is the background to my research, and it would help me a lot, if you could answer my questions. Your participation is completely voluntary, your identity so far has been kept from me and will remain being kept from me, unless you specifically agree to reveal yourself to me. You can

choose to withdraw at any point in time, without fear of negative consequences. My research has been approved by the ethics committee for EEMCS with reference number RP 2020-54. If you are willing to help, could you please answer the following questions?

1. Before receiving this email, were you aware that your device has been tunneling internet traffic for third parties?

   If not:

   1.1. Do you use commercial VPNs?

   1.2. Could you think of any reason, why your device is behaving this way?

   1.3. Are you interested in investigating the root cause of this behavior, together with me, the principal researcher? (If you are not willing to reveal your identity, are you willing to communicate through a third party?

   If yes:

   1.1. Do you use commercial VPNs?

   1.2. Which software is responsible for this behavior?

   1.3. Were you approached by a third party, with the explicit request to tunnel traffic?

   1.4. Do you receive any kind of compensation or benefits by doing this?

2. Would you mind being contacted again in the future?

Thank you for your attention, should you have any other kind of question or feedback, you can contact the sender of this mail, my supervisors (`https://www.utwente.nl/en/eemcs/dacs/staff/`), or me directly.

Kind regards,

Etienne Khan

E. ICT.OPEN2021

Halfway through December, shortly before the Christmas period, we have been approached by our supervisors with the question if we would like to submit the preliminary findings to the ICT.OPEN2021 research poster competition. ICT.OPEN2021 is part of the Dutch Digital Conference from 8 to 10 February 2021, organized by NWO (Nederlandse Organisatie voor Wetenschappelijk Onderzoek, the national research council of the Netherlands) and the Ministry of Economic Affairs and Climate Policy. The Dutch Digital Conference connects science, industry, societal organizations and government. During several online events, visitors share their knowledge and work together on innovative digital solutions.

Part of this event is the aforementioned poster competition where young researchers (PhD and postdoc level) can present their work. Additionally, poster presenters are eligible for three poster prizes of €500, €250 and €100.

We feel very honored, to have been able to participate with a master's research thesis. Our poster submission, winning the second prize, can be seen on the following page.

STRANGER VPNS

99% Match 2021

Author: Etienne Khan

Contact: etienne@dns.ph

Affiliation: Design and Analysis of Communication Systems (DACS)

Keywords: VPN, Geo-unblocking, Residential Proxy

The story so far...

What?

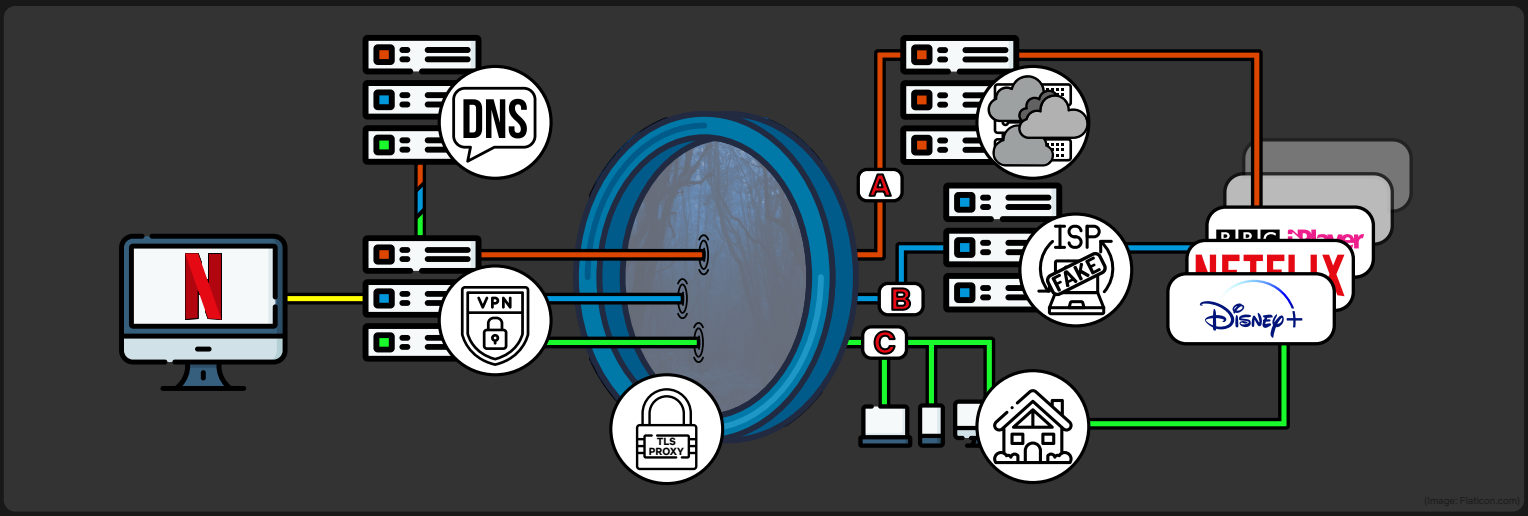
Streaming platforms offer a large variety of content, but its availability depends on the location of the viewer. Some commercial VPN providers **advertise to lift these restrictions** by leveraging their globally distributed server network, even though many streaming platforms employ **VPN detection mechanisms**.

Why?

The commercial VPN ecosystem is **"highly opaque"**, a lack of independent and peer-reviewed evaluation means that functions such as geo-unblocking are **effectively black boxes**. The purpose of this research is to look into these black boxes and see how they work.

How?

By purchasing subscriptions to **6 large VPN providers**, which advertise with unblocking capabilities, and setting up a measurement testbench for **longitudinal data collection**.



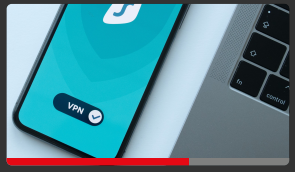
Preliminary Results

1

The Three Main Methods

- **A** Using **"obscure" hosting** providers
- **B** Creation of **fake** (residential) **ISPs**
- **C** With **residential proxies**

2



Unblocking Capabilities

So far, more than **38,500 unique IP addresses** located in over **400 unique Autonomous Systems** have been recorded. Of those, **0.89% are IPv6 addresses (~340)**.

3

Notable Unblocking Exit-Networks

Apart from **residential ISPs**, exit-networks have also been identified in **enterprise networks**, **research networks**, and **government networks**.

UNIVERSITY OF TWENTE.

DACS
Design and Analysis of Communication Systems

References

[1] W. N. Price and I. G. Cohen, "Privacy in the age of medical big data." *Nature Medicine*, vol. 25, no. 1, pp. 37–43, 2019.

[2] S. Kirchgaessner, P. Lewis, D. Pegg, S. Cutler, N. Lakhani, and M. Safi. (2021, Jul) Revealed: Leak uncovers global abuse of cyber-surveillance weapon. [Online]. Available: https://www.theguardian.com/world/2021/jul/18/revealed-leak-uncovers-global-abuse-of-cyber-surveillance-weapon-nso-group-pegasus

[3] L. Verhagen. (2021, Jan) Datalek bij ggd: Gegevens van miljoenen nederlanders in criminele handen. [Online]. Available: https://www.volkskrant.nl/nieuws-achtergrond/datalek-bij-ggd-gegevens-van-miljoenen-nederlanders-in-criminele-handen~b7f17bea/

[4] C. Kahn. (2021, Jul) Cuban government blocks the internet in an attempt to thwart protesters. [Online]. Available: https://www.npr.org/2021/07/14/1015895923/cuban-government-blocks-the-internet-in-an-attempt-to-thwart-protesters

[5] K. Paul. (2021, May) Facebook under fire as human rights groups claim 'censorship' of pro-palestine posts. [Online]. Available: https://www.theguardian.com/media/2021/may/26/pro-palestine-censorship-facebook-instagram

[6] O. Valentine. (2018, Jul) Vpns are primarily used to access entertainment. [Online]. Available: https://blog.gwi.com/chart-of-the-day/vpns-are-primarily-used-to-access-entertainment/

[7] A. M. Research. (2021, Jan) Virtual private network (vpn) market to reach $75.59 bn, globally, by 2027 at 14.7% cagr: Amr. [Online]. Available: https://www.prnewswire.com/news-releases/virtual-private-network-vpn-market-to-reach-75-59-bn-globally-by-2027-at-14-7-cagr-amr-301206042.html

[8] Alphonso. (2021, Jan) Leading brands on cnn in the united states as of 1st quarter 2018. [Online]. Available: https://www.statista.com/statistics/910145/leading-brands-cnn-ad-spending/

[9] M. T. Khan, J. DeBlasio, G. M. Voelker, A. C. Snoeren, C. Kanich, and N. Vallina-Rodriguez, "An empirical analysis of the commercial vpn ecosystem," in *Proceedings of the Internet Measurement Conference 2018*, 2018, pp. 443–456.

[10] T. Spark. VPN TIER LIST. [Online]. Available: https://www.vpntierlist.com/

[11] J. Rolfe. (2020, Jan) Raid 2020 (NES) - Angry Video Game Nerd (AVGN). [Online]. Available: https://youtu.be/fuOmmIOdPg8

[12] Design and Analysis of Communication Systems, University of Twente, Enschede, 2021. [Online]. Available: https://www.utwente.nl/en/eemcs/dacs/

[13] S. Zander, "On the accuracy of ip geolocation based on ip allocation data," *Centre for Advanced Internet Architectures, Technical Report 120524A*, 2012.

[14] J. A. Muir and P. C. V. Oorschot, "Internet geolocation: Evasion and counterevasion," *ACM Comput. Surv.*, vol. 42, no. 1, Dec. 2009. [Online]. Available: https://doi.org/10.1145/1592451.1592455

[15] V. N. Padmanabhan and L. Subramanian, "An investigation of geographic mapping techniques for internet hosts," *SIGCOMM Comput. Commun. Rev.*, vol. 31, no. 4, p. 173–185, Aug. 2001. [Online]. Available: https://doi.org/10.1145/964723.383073

[16] (2021) Proxy Detection & VPN Identification. [Online]. Available: https://www.digitalelement.com/solutions/user-context/vpn-proxy/

[17] (2021) Additional Audience Insights. [Online]. Available: https://www.digitalelement.com/solutions/user-context/more-insights/

[18] Y. Rekhter, S. Hares, and T. Li, "A Border Gateway Protocol 4 (BGP-4)," RFC 4271, Jan. 2006. [Online]. Available: https://rfc-editor.org/rfc/rfc4271.txt

[19] R. Housley, J. Curran, G. Huston, and D. R. Conrad, "The Internet Numbers Registry System," RFC 7020, Aug. 2013. [Online]. Available: https://rfc-editor.org/rfc/rfc7020.txt

[20] CAIDAl. (2021) AS Rank. [Online]. Available: https://asrank.caida.org/asns

[21] M. Luckie, B. Huffaker, and k. claffy, "Learning regexes to extract router names from hostnames," in *Proceedings of the Internet Measurement Conference*, 2019, pp. 337–350.

[22] J. Chabarek and P. Barford, "What's in a name? decoding router interface names," in *Proceedings of the 5th ACM workshop on HotPlanet*, 2013, pp. 3–8.

[23] Akamai. Edgescape. [Online]. Available: https://developer.akamai.com/edgescape

[24] D. Element. Ip geolocation. [Online]. Available: https://www.digitalelement.com/geolocation/

[25] Maxmind. Geoip2. [Online]. Available: https://www.maxmind.com/en/solutions/geoip2-enterprise-product-suite

[26] Hexasoft. Ip2location. [Online]. Available: https://www.ip2location.com/

[27] M. Gouel, K. Vermeulen, O. Fourmaux, T. Friedman, and R. Beverly, "IP Geolocation Database Stability and Implications for Network Research," *Network Traffic Measurement and Analysis Conference*, 2021.

[28] E. Kline, K. Duleba, Z. Szamonek, S. Moser, and W. A. Kumari, "A Format for Self-Published IP Geolocation Feeds," RFC 8805, Aug. 2020. [Online]. Available: https://rfc-editor.org/rfc/rfc8805.txt

[29] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida, "Constraint-based geolocation of internet hosts," *IEEE/ACM Transactions on Networking*, vol. 14, no. 6, pp. 1219–1232, 2006.

[30] E. Katz-Bassett, J. P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe, "Towards ip geolocation using delay and topology measurements," in *Proceedings of the 6th ACM SIGCOMM Conference on Internet Measurement*, ser. IMC '06. New York, NY, USA: Association for Computing Machinery, 2006, p. 71–84. [Online]. Available: https://doi.org/10.1145/1177080.1177090

[31] Akamai. Facts & figures. [Online]. Available: https://www.akamai.com/company/facts-figures.jsp

[32] Y. Shavitt and N. Zilberman, "A geolocation databases study," *IEEE Journal on Selected Areas in Communications*, vol. 29, no. 10, pp. 2044–2056, 2011.

[33] Webwire. Independent Performance Assessment Validates Accuracy of Akamai's IP Location Service. [Online]. Available: https://www.webwire.com/ViewPressRel.asp?aId=104305

[34] Apple. Private relay geolocation feed. [Online]. Available: https://mask-api.icloud.com/egress-ip-ranges.csv

[35] Google. Google corp geofeed. [Online]. Available: https://www.gstatic.com/geofeed/corp_external

[36] B. Huffaker, M. Fomenkov, and k. claffy, "Geocompare: a comparison of public and commercial geolocation databases - technical report," Cooperative Association for Internet Data Analysis (CAIDA), Tech. Rep., 2011-05.

[37] Netflix. Watching tv shows and movies through a vpn. [Online]. Available: https://help.netflix.com/en/node/114701/

[38] ——. Netflix says 'you seem to be using an unblocker or proxy.'. [Online]. Available: https://help.netflix.com/en/node/277

[39] GeoComply. Shift72 protects high value content from geo-piracy with geoguard vpn and proxy detection. [Online]. Available: https://www.geocomply.com/news/shift72-protects-prime-content-with-geoguard-solution/

[40] D. Element. Digital element commemorates 20th anniversary. [Online]. Available: https://www.digitalelement.com/digital_element_20th_anniversary/

[41] FOX4. Our european visitors are important to us. [Online]. Available: https://fox4kc.com

[42] T. T. Project. Tor bulk exit list. [Online]. Available: https://check.torproject.org/torbulkexitlist

[43] PrivateVPN. Getting started. [Online]. Available: https://privatevpn.com/support/getting-started

[44] OpenVPN. Reference manual for openvpn 2.4. [Online]. Available: https://openvpn.net/community-resources/reference-manual-for-openvpn-2-4/

[45] Amazon. Netflix on aws. [Online]. Available: https://aws.amazon.com/solutions/case-studies/netflix/

[46] M. Cotton, G. Huston, and L. Vegoda, "IANA IPv4 Special Purpose Address Registry," RFC 5736, Jan. 2010. [Online]. Available: https://rfc-editor.org/rfc/rfc5736.txt

[47] IANA. IANA IPv4 Special-Purpose Address Registry. [Online]. Available: https://www.iana.org/assignments/iana-ipv4-special-registry/iana-ipv4-special-registry.xhtml

[48] D. Fifield, J. Jian, and P. Pearce. SNI proxies. [Online]. Available: https://www.bamsoftware.com/computers/sniproxy/

[49] D. Stenberg. (1997) cURL. [Online]. Available: https://curl.se/

[50] Frontier Communications Parent, Inc. Internet, Phone & TV Service Provider | Frontier. [Online]. Available: https://frontier.com

[51] D. Element. NetAcuity - VPN Proxy Database. [Online]. Available: https://www.digitalelement.com/wp-content/uploads/2018/09/VPN-Proxy-Datasheet.pdf

[52] Telia Company AB. Markets and brands. [Online]. Available: https://www.teliacompany.com/en/about-the-company/markets-and-brands/

[53] Telia Carrier. IP Transit. [Online]. Available: https://www.teliacarrier.com/products-and-services/internet-and-cloud/ip-transit.html

[54] PrivateVPN. PrivateVPN. [Online]. Available: https://web.archive.org/web/20210101002159/https://privatevpn.com/

[55] ——. PrivateVPN. [Online]. Available: https://web.archive.org/web/20210101211920/https://privatevpn.com/

[56] allabolag. PVDataNet AB. [Online]. Available: https://www.allabolag.se/5568951486/pvdatanet-ab

[57] PVDataNet. The future of data centers industry. [Online]. Available: https://pvdatanet.com/index.htm

[58] Norisab. About us. [Online]. Available: https://norisab.com/about.html

[59] RIPE Database. RIPE Database search service. [Online]. Available: https://apps.db.ripe.net/db-web-ui/lookup?source=ripe&key=MM51507-RIPE&type=role

[60] RIPE. Local internet registries offering service in sweden. [Online]. Available: https://www.ripe.net/membership/indices/SE.html

[61] V. C. Perta, M. V. Barbera, G. Tyson, H. Haddadi, and A. Mei, "A glance through the vpn looking glass: Ipv6 leakage and dns hijacking in commercial vpn clients," *Proceedings on Privacy Enhancing Technologies*, vol. 2015, no. 1, pp. 77–91, 2015. [Online]. Available: https://doi.org/10.1515/popets-2015-0006

[62] M. Ikram, N. Vallina-Rodriguez, S. Seneviratne, M. A. Kaafar, and V. Paxson, "An analysis of the privacy and security risks of android vpn permission-enabled apps," in *Proceedings of the 2016 Internet Measurement Conference*, ser. IMC '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 349–364. [Online]. Available: https://doi.org/10.1145/2987443.2987471

[63] Z. Weinberg, S. Cho, N. Christin, V. Sekar, and P. Gill, "How to catch when proxies lie: Verifying the physical locations of network proxies with active geolocation," in *Proceedings of the Internet Measurement Conference 2018*, ser. IMC '18. New York, NY, USA: Association for Computing Machinery, 2018, p. 203–217. [Online]. Available: https://doi.org/10.1145/3278532.3278551

[64] P. Winter, R. Padmanabhan, A. King, and A. Dainotti, "Geo-locating bgp prefixes," in *2019 Network Traffic Measurement and Analysis Conference (TMA)*, 2019, pp. 9–16.