Monetizing Spare Bandwidth: The Case of Distributed VPNs

YUNMING XIAO, Northwestern University, USA
MATTEO VARVELLO, Nokia Bell Labs, USA
ALEKSANDAR KUZMANOVIC, Northwestern University, USA

Residential Internet speeds have been rapidly increasing, reaching averages of ~100 Mbps in most developed countries. Several studies have shown that users have way more bandwidth than they need, only using about 20-30% on a regular day. Several systems exploit this trend by enabling users to monetize their spare bandwidth, *e.g.,* by sharing their WiFi connection or by participating in distributed proxy or VPN (dVPN) services. Despite the proliferation of such systems, little is known on how such marketplaces operate, what are the key factors that determine the price of the spare bandwidth, and how such prices differ worldwide. In this work, we shed some light on this topic using dVPNs as a use-case. We start by formalizing the problem of bandwidth monetization as an optimization between a buyer's cost and seller's income. Next, we explore three popular dVPNs (Mysterium, Sentinel, and Tachyon) using both *active* and *passive* measurements. We find that dVPNs have a large and growing footprint, and offer comparable performance to their centralized counterpart. We identify Mysterium (in the US) as the most concrete realization of a bandwidth marketplace, for which we derive a value of spare Internet bandwidth ranging between 11 and 14 cents per GB. We also show that both buyers and sellers utilize ad-hoc "rules-of-thumb" when choosing their prices, which results in a sub-optimal marketplace. By applying our optimization, a seller's income can be tripled by setting a price lower than the default one which allows to attract more buyers. These observations motivate us to create *RING*, a first and concrete system which helps sellers to automatically adjust their prices and traffic volumes across multiple marketplaces.

CCS Concepts: • **Networks** → **Network services**; **Network security**; **Wide area networks**; **Public Internet**; **Peer-to-peer networks**; **Network performance evaluation**; • **Applied computing** → **Economics**.

Additional Key Words and Phrases: Distributed VPN; Network Measurement; Traffic Characterization; Bandwidth Monetization; Optimization

ACM Reference Format:
Yunming Xiao, Matteo Varvello, and Aleksandar Kuzmanovic. 2022. Monetizing Spare Bandwidth: The Case of Distributed VPNs. *Proc. ACM Meas. Anal. Comput. Syst.* 6, 2, Article 33 (June 2022), 27 pages. https://doi.org/10.1145/3530899

1 INTRODUCTION

According to Ookla [23], average Internet speeds in American homes grew 20x in the last 10 years, from 8 Mbps (2010) to 180 Mbps (2021). A similar trend is observable worldwide [22]. Several recent studies [27, 40, 52, 53] suggest that the added cost for faster Internet speeds — *e.g.,* $50 monthly to boost from 200 to 300 Mbps with Comcast Xfinity [5] — is not worth it to most residential users, as only a median of 5% of bandwidth is used. For example, a family of 4 concurrently streaming HD videos only requires ~20 Mbps, not to mention unused bandwidth at night.

Authors' addresses: Yunming Xiao, Northwestern University, USA; Matteo Varvello, Nokia Bell Labs, USA; Aleksandar Kuzmanovic, Northwestern University, USA.

2476-1249/2022/6-ART33 $15.00
https://doi.org/10.1145/3530899

Motivated by the above notion of *spare* bandwidth, several systems offer mechanisms which allow users to monetize their unused bandwidth, *e.g.,* by sharing their WiFi connection or by participating in distributed proxy or VPN (Virtual Private Network) services [8, 11, 13, 15, 21, 24]. Effectively, these applications are building *bandwidth marketplaces* where spare bandwidth is auctioned. For example, Alice (France) is willing to pay Bob (US) $1 to tunnel her video traffic and avoid geo-blocking.

While such systems are attracting a considerable number of users, both as clients (buyers) and providers (sellers) of spare bandwidth, little to nothing is known about the properties of such marketplaces, and the dominant factors that affect them. Understanding the properties of existing marketplaces helps to shed light on the future market evolution as well as designing more efficient ones, which is a topic we explore at the end of this paper. Further, existing marketplaces can help us understand which value buyers and sellers associate with (spare) Internet bandwidth today.

We identified *distributed* VPNs (dVPNs) — a new form of VPN with no central authority — as the most concrete examples of such marketplaces. That is because, in essence, a dVPN is a system which allows users to sell their spare bandwidth. The first contribution of this paper is an investigation of the dVPN ecosystem. We run *active* measurements to characterize their footprint, pricing schemes, and performance. We further run *passive* measurements by contributing multiple nodes (both at residential and cloud locations) while experimenting with traffic filters and pricing.

The analysis of six months of dVPNs data shows that the three major dVPNs (Mysterium [11], Sentinel [21], Tachyon [24]) compose together a fast-growing network footprint of thousands of nodes (sellers) located all over the world. These nodes offer download speeds comparable with ProtonVPN [16], a popular centralized VPN. Location-wise, the US is the most attractive market, with the majority of the traffic being destined to services located in the US and served by our US node (10 of the 16 TB we served over 6 months). Traffic is mostly HTTP(s) and "safe", with only 2% being categorized as potentially malicious by McAfee domain classification [2].

The second contribution of this paper is a formalization of the bandwidth monetization problem by considering a *single-vendor* bandwidth marketplace. Next, we analyze the price ranges that create the most efficient marketplace, both in terms of the sellers' revenue, and the number of buyers that are willing to join such marketplaces at the given price range. We find that the value of Internet spare bandwidth in the US ranges between 11 and 14 cents per GB. However, neither the buyers nor the sellers are optimizing their costs or incomes. This leaves an opportunity for a seller to maximize income by reducing the price and attracting more buyers. Given our measurements show that the bandwidth demand is less than the supply, if more sellers adopt such optimization, a buyer's cost and the spare bandwidth value can be potentially reduced.

Motivated by the latter observation, we extend our modeling to a *multi-vendor* bandwidth marketplace. We then realize such a marketplace with RING, the third contribution of this paper. RING is a software which offers its users fine-grained control on *how* and *where* to monetize their spare bandwidth. RING achieves this by incorporating multiple dVPNs (Mysterium, Sentinel, Tachyon) along with traffic policies which optimize the user's income. We demonstrate RING's functioning via a one-week controlled experiment where it helped increase a node revenue by 63%. We plan to open-source and release RING to the public.

2 PROBLEM FORMALIZATION AND MOTIVATION

In this section, we formalize the bandwidth monetization problem in a single-vendor marketplace. This will help us to comprehensively analyze the existing marketplaces and quantify if, and how sub-optimal, they may be. We assume that a *seller* at location l offers her spare bandwidth to potentially multiple concurrent buyers. We further assume that the seller's Internet connection is characterized by some (spare) speed r, and a data cap D in a period of time T. For example,

Comcast Xfinity offers download speed up to 1,200 Mbps, depending on the monthly price tag, for a maximum of 1.2 TB per month [31]. Note that r is the minimum between download and upload speeds, or $r = min\{r_u, r_d\}$. This is because a seller is not an end-point but a "middle-point", which is required to utilize both her upload and download bandwidth. For example, when a buyer downloads a 1 MB file from the Internet, for the seller, this translates to 1 MB of download data which then needs to be uploaded to the buyer. Thus, the seller carries twice as much data as the buyer, and the actual speed depends on where the bottleneck is between a seller's download and upload bandwidth.

We consider a charging scheme defined by the pair (x, y), which represents the *amount of data consumed* and *duration*, *e.g.,* (Gigabytes, seconds). This is reasonable and representative of what we have observed in several bandwidth marketplaces (see Table 1). In addition, the seller may or may not be willing to carry some "dangerous" traffic, *e.g.,* contacting IP addresses labeled unsafe by services like the Safebrowsing list [35]. Generally speaking, we assume a seller defines a blocklist $A = \{dst1, dst2, \ldots\}$, which includes the set of destination IP addresses which should be blocked.

We call S the set of sellers participating in a marketplace. Each seller $s_j \in S$ posts her service in the marketplace defined by a location l_j, price settings x_j, y_j, rate limit r_j and blocklist A_j. Buyers can see seller details $s_j = (l_j, x_j, y_j, r_j, A_j)$ and decide to buy, hence connect, or not. In the following, we formalize the optimizations from both a buyer and a seller perspective.

Buyers' Perspective – Assume a buyer is looking for bandwidth with average speed b for a duration u. The buyer is further looking for bandwidth from sellers within a set of locations L, and her traffic is directed to a destination set DES. The chosen seller $s_j \in S$ must satisfy $r_j \geq b, DES \subseteq A_j, l_j \in L$. The price P the buyer needs to pay to seller s_j is:

$$P = x_j \cdot b \cdot u + y_j \cdot u \tag{1}$$

The buyer naturally would like to minimize her cost, given equal performance. Let $S(b, L, DES) \subseteq S$ be a sellers set which matches a buyer's constraints. To minimize the buyer's cost, a seller can be selected by optimizing the following objective function:

$$\min_{s_j \in S(b,L,DES)} P(S) = \min_{s_j \in S(b,L,DES)} (x_j \cdot b \cdot u + y_j \cdot u) \tag{2}$$

Intuitively, the process consists of filtering the sellers by given constraints (b, L, DES). Then, for a demanded bandwidth b, the buyer should select the seller among the left sellers minimizing $x_j \cdot b + y_j$.

Sellers' Perspective – We assume that connection decisions of each buyer are independent events, and that the number of buyers is large. Under this assumption, the arrival process of bandwidth buyers would follow the Poisson distribution, *i.e.,* $\mathbf{n} \sim Poisson(N)$, where $\mathbf{n}$ is the number of buyers and N is the mean number of buyers within time T. Intuitively, N is a function of (x, y, r, l, A) since it depends on the service price, quality, location, and seller's blocklist.

Let $\mathbf{B}$ and $\mathbf{U}$ be the random variables of bandwidth and duration of incoming traffic sessions, respectively. The expectation of income $\mathbf{I}$ of a seller within a period of time T can then be formalized as follows:

$$\mathbb{E}[I(x, y, r, l, A)] = x \cdot \mathbb{E}[\mathbf{n} \cdot \mathbf{B} \cdot \mathbf{U}] + y \cdot \mathbb{E}[\mathbf{n} \cdot \mathbf{U}] \tag{3}$$

The seller would naturally like to maximize her income. We optimize the seller's income by adjusting the unit prices x, y and bandwidth speed limit r, *i.e.,* we maximize the following objective function:

Table 1. Summary of current dVPN solutions.

dVPN	Client Platform	Node Platform	Open Source	Payment Scheme	Tunneling Protocol	Comments
Mysterium	Android, Mac, Windows	ARM, x86	Yes	Data transferred, connection time	OpenVPN, WireGuard	–
Sentinel	Android, Mac, Windows, Linux	ARM, x86	Yes	Data transferred	WireGuard	–
Tachyon	Android, Mac, iOS	x86	No	Data transferred, staking reward	Tachyon	Still in development
Privatix	Android, Mac, Windows, Linux	ARM, x86	Yes	Data transferred	OpenVPN	Cannot join [36]
Orchid	Android, Mac, iOS	ARM, x86	Yes	Not yet released	OpenVPN, WireGuard	Still in development
Lethean	X	ARM, x86	Yes	Data transferred	OpenVPN	Dead project

$$\max_{x,y,r}(x \cdot \mathbb{E}[\mathbf{n} \cdot \mathbf{B} \cdot \mathbf{U}] + y \cdot \mathbb{E}[\mathbf{n} \cdot \mathbf{U}]) \;\; s.t. \;\; \mathbb{E}[\mathbf{n} \cdot \mathbf{B} \cdot \mathbf{U}] \leq D/2 \tag{4}$$

3 DATA COLLECTION

Distributed VPNs (dVPNs) — a new form of VPN with no central authority — are realizations of single-vendor bandwidth marketplaces. That is because, in essence, a dVPN is a tool which allows users to sell their (spare) bandwidth. In this section, we first provide some background on existing dVPNs. Next, we describe the methodology we have designed to perform both *active* and *passive* measurements of dVPNs.

3.1 Background

DVPN users can have two roles, either concurrently or disjointly: *node* and *client*. A user acts as a *node* when it forwards traffic on behalf of other users and requests some form of compensation for this. A user acts as a *client* when it pays for its traffic to be tunneled via a dVPN node of her choice.

Over the years, there have been many attempts at building dVPNs. For instance, VPN Gate [29] is a dVPN originated as a research project [49] to achieve blocking resistance to censorship firewalls. It is however not useful for our study since its users are not allowed to arbitrarily charge for their bandwidth, thus not making it a good approximation of an actual bandwidth marketplace.

With the above in mind, appropriate candidates for our study are recent dVPNs which originated in conjunction with the rise of blockchain [45]. In particular, dVPN nodes can directly determine their prices and users are granted visibility on information like price charged, *e.g.,* cryptocurrency per GB, node location, and expected bandwidth (Mbps). Examples of such dVPNs are: Mysterium [11], Sentinel [21], Privatix [15], Tachyon [24], Orchid [13], and Lethean [8].

Figure 1 depicts the working procedure of such dVPNs. First, dVPN nodes register their "offering" (*e.g.,* location and cost per GB) at the dVPN broker. When a client wants to use the dVPN service, she requests the currently available offerings from the dVPN broker and selects a node to establish a VPN tunnel to. The tunnel is established directly between the client and the node. In the meantime, transactions are generated per the agreement between client and node, and are executed and recorded on the blockchain. It thus follows that recent dVPNs fit in the category of decentralized applications (dapps) [38].

Table 1 provides an overview of the above dVPNs. The table shows that Android is the most common client platform. With respect to the node, all dVPNs except Tachyon are open source and offer executables for both ARM and x86. The payment scheme of most dVPNs is based on

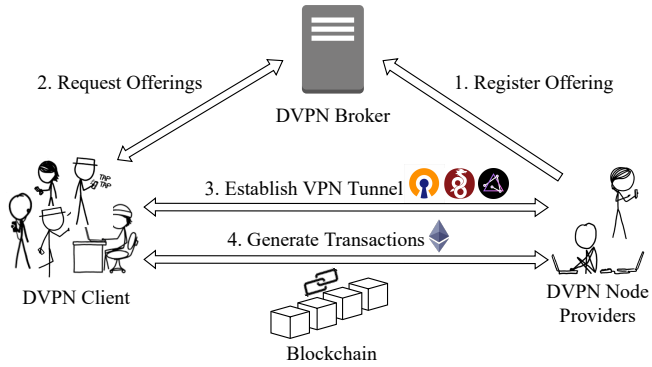


Fig. 1. Visualization of dVPN functioning.

how much data is transferred through the node. Mysterium also charges clients by how long they connect to a node. In addition, Tachyon pays a *staking* reward; "staking" refers to the fact that the seller needs to put down some amount of cryptos (a stake) before they can receive rewards. In terms of VPN tunneling, OpenVPN [12] and WireGuard [30] are the most popular protocols adopted. The only exception is Tachyon, which uses a proprietary protocol also named "Tachyon" [7].

In green, we have highlighted the dVPNs which we have selected for both our active and passive monitoring: Mysterium, Sentinel, and Tachyon. These dVPNs were selected since they offer stable client and node implementations, and a payment scheme which is representative of a bandwidth marketplace. Privatix (orange) was instead only studied *actively*, *i.e.,* by leveraging its client to test its current nodes, since our (multiple) attempts to join the Privatix network as nodes have not been successful. Finally, Orchid and Lethean (red) could not be studied for the following reasons. Orchid has not opened node registration to regular users but only to partners; further their client was quite unreliable during automation. Lethean has no working client and its *staking* account, from which users need to acquire funds, is currently unavailable.

3.2 Methodology

We here describe our methodology to explore the dVPNs selected above and collect the data needed to populate our model of a generic single-vendor bandwidth marketplace. Our rationale is to both *actively* and *passively* collect data from a dVPN. Active experiments consist of automating a dVPN client to send traffic via the available nodes. This is useful to learn about their footprint, pricing, and performance. Passive experiments consist of contributing bandwidth to such dVPNs by running several nodes. We are interested in characterizing how much traffic a typical dVPN node carries, and thus how much revenue (cryptocurrency) it could generate. Further, we want to explore the traffic *characteristics*, *e.g.,* presence or lack of harmful traffic, to help asses the *risk* associated with running a dVPN node.

3.2.1 Active. As per Table 1, our active experiments rely on Android dVPN clients because Android is the common platform among all dVPNs we aim to investigate. We have further confirmed that there are no significant differences between the information (node locations and prices) available through the different client platforms. Our testbed consists of an Android device (a Samsung S9 running Android 10) controlled by a Raspberry Pi 4. The Android device is used to run the dVPN apps, while the Raspberry Pi realizes the automation, *e.g.,* launch a dVPN app and select a node to connect to. We chose the Raspberry Pi for its convenience and given that its task is simple and

more powerful hardware is not required. The Android device connects to a fast WiFi (80 Mbps upload/download bandwidth) and is located in North America.

We automate dVPN usage via the Android Debugging Bridge (ADB [32]), a rich Android protocol which allows to automate app operations like launching, scrolling, and GUI interaction. At a high level, we use ADB to instrument each dVPN app to automatically iterate through its available nodes, while attempting a connection. We rely on visual inspection of screen recordings to verify and learn how to iterate through all states each dVPN app can reach, *e.g.,* connection ready or more random states like *rate the app*, which we then translate into automation scripts.

We use several techniques to both gather information about a dVPN and enforce correct crawling functionalities. For example, we monitor Android network interfaces to verify successful "connect" and "disconnect" operations. We also rely on Android logging (`logcat`) where developers often log information like state changes, node IPs, payments, etc. We further use screenshots, both XML of the information on screen (via `uiautomator`) and actual images coupled with OCR processing, to collect statistics which are only available on screen. When available, we also resort to public APIs, as in the case of Sentinel [37] which curiously even reports the CPU consumption measured at its dVPN nodes.

We use the above automation to build two active measurement campaigns: *discovery*, and *speedtest*.

Discovery – The goal of this measurement is to discover nodes offered by a dVPN, along with any public information like pricing and advertised bandwidth. This implies quickly iterating through the GUI of each dVPN (or query its public API, if available) logging information about node counts and locations. Since this method does not require to connect to each node, it is quite lightweight, and we thus run it daily over 6 months, from December 2020 to May 2021.

Speedtest – The goal of this measurement is to benchmark the *connectivity* (availability, location, and download/upload bandwidths) of the nodes offered by a dVPN. This requires connecting to each node discovered using the above procedure, to then perform a speedtest. Compared to the discovery measurement, this test is more complex and invasive. We thus resorted to run it monthly; further, in presence of very large dVPNs we sample a subset of the nodes by selecting a maximum of 10 nodes per country.

To perform speedtests, we leverage the public service offered by Netflix at *https://fast.com* automated via ADB. First, we configure a target dVPN node to be tested. Then, we launch the Chrome browser and visit the speedtest website. Last, we use `uiautomator` – which dumps content on screen in XML format – to retrieve measured bandwidths, latencies, estimated location, and server used for testing. We also take a screenshot of the page to retrieve the above information via OCR in case of failure of `uiautomator` (which can happen in presence of dynamic content on screen). To avoid very long and expensive tests, we limit the duration of each test to 10 seconds, which implies a maximum upload/download of 100 MB under our (residential) connectivity.

3.2.2 Passive. In these measurements, we run nodes for the main dVPN providers while passively collecting their traffic using Tstat [28], a popular traffic sniffer which automatically analyzes TCP and UDP traffic. Tstat uses the classic 5-tuples[1] to identify TCP *sessions* and UDP *flows*. TCP sessions are identified using TCP connection establishment process. UDP flows are harder to detect since there is no explicit notion of a session. Tstat defines a UDP flow as the set of packets with same 5-tuples with inter-arrival times smaller than 200 seconds. We further call dVPN *session* a collection of TCP/UDP flows between client and node, and *outgoing sessions* the TCP/UDP traffic between dVPN node and destination IPs.

[1] < *IP_SRC*, *IP_DST*, *PORT_SRC*, *PORT_DST*, *PROTO*>

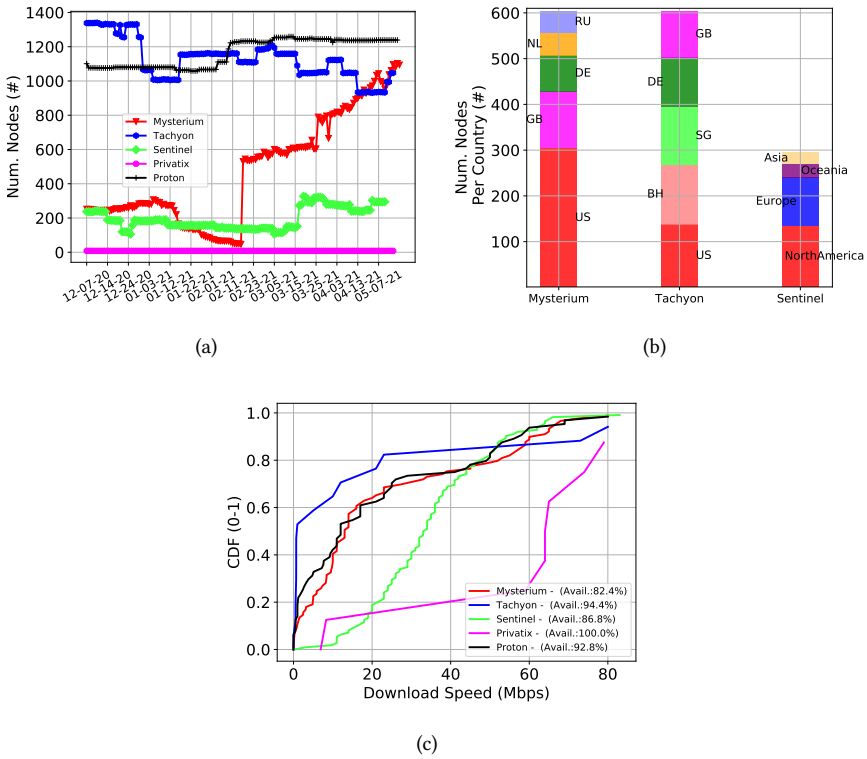


Fig. 2. Footprint and performance characterization of the dVPN ecosystem: (a) Evolution over 6 months of the number of nodes; (b) Histogram of number of nodes per country; (c) CDF of download speed with availability.

Given HTTPS represents the majority of today's Internet traffic [4], we rely on DNS – when not encrypted – and SNI – not yet encrypted even with TLSv1.3 [19] — for coarse traffic characterization, *i.e.,* we identify accessed domains but not, for instance, specific webpages. Next, we adopt McAfee domain classification service [2] which achieves the highest coverage according to [56], *i.e.,* 94% over 4.4 million domains. McAfee provides two attributes per domain: *reputation* and *type*. "Reputation" is calculated dynamically by the TrustScore system [17] and maps to four ratings: minimal risk (<15), unverified (15-30), medium risk (30-50), and high risk (>50). "Type" depends on the content available at a given domain, *e.g., facebook.com* corresponds to social networking. We further use signature matching provided by IPP2P to identify P2P traffic.

We have deployed 10 machines (2 residential and 8 cloud) running nodes for Mysterium, Sentinel, and Tachyon, across 4 countries (US, UK, Italy, and China). With respect to pricing, we always adopt the lowest pricing available to make our nodes appealing to clients. The two residential machines are both located in the US: a Ubuntu desktop located in New Jersey (US-R1),[2] and a Raspberry Pi located in Illinois (US-R2). Next, we have deployed one machine at each of the following cloud providers: AWS [33] (US), OVH [14] (UK), Keliweb [6] (Italy), Alibaba Cloud [1] (China).

In AWS, we further deploy four extra nodes (US-C2 to US-C5) to investigate variable pricing: i) 0.02, 0.05, 0.1, 0.25 MYST per GB, ii) 30, 50, 70, 90 SENT per GB. We did not investigate price

[2]US refers to the machine location, R to its residential characteristic. C will be used for cloud. The integer is used to provide unique naming.

variations with Tachyon for two reasons: i) Tachyon requires the seller to invest (stake) several hundred USD before being able to set the price. In absence of such investment, a node provides free dVPN service. ii) Tachyon clients are not required to actually pay, nor does Tachyon app show any price information for now. It follows that experimenting with variable prices on Tachyon would be expensive and unrealistic, thus we have left it as a potential future work.

Our passive measurements last for 3 months (February to April 2021) and account for ∼16 TB of traffic. IRB at our institution has determined that our work is not considered human research (details in Appendix A). We have also verified that we do not violate the terms of service of Sentinel, Tachyon, and Privatix. Instead, Mysterium's terms of service have restricted conducts that are related to our data collection process. We have thus reached out to Mysterium, and obtained permission to perform and publish this study.

4 DATA ANALYSIS

This section analyzes the data collected via our active and passive experiments. The analysis provides a detailed view of the dVPN ecosystem with respect to its footprint, performance, and traffic characteristics. We then further investigate whether dVPNs are indeed concrete representations of a bandwidth marketplace, and the collected data-set can be used to model the variables contributing to the bandwidth monetization problem discussed in Section 2.

Footprint and Performance – We start by investigating the *footprint* of the current dVPN ecosystem, *i.e.,* how many nodes compose each dVPN and where they are located. Figure 2(a) shows, for each dVPN, the evolution over the last six months (December 2020 – May 2021) of the *total* number of nodes advertised by each dVPN. The figure is further enhanced with data collected from ProtonVPN [16], a popular centralized VPN, given a basic account ($5 per month).

Figure 2(a) shows that, initially, only Tachyon had a footprint comparable with ProtonVPN, *i.e.,* in the order of one thousand nodes. However, Tachyon has lost 36% of its nodes over time while Mysterium's node count has been steadily increasing after February, and it is the largest dVPN with 1,100 nodes by the end of May. Mysterium has been *losing* nodes in January/February (reaching a minimum of 50 nodes), followed by a sudden increase to 530 nodes on 02-17-21. This behavior was an artifact due to Mysterium's migration from their version 1.0 to 2.0, which progressively made part of the network appears to be offline as discussed in [26]. Note also that ProtonVPN has added 100 news nodes in this time span. While Sentinel has also increased its footprint, it currently attracts a relatively small number of nodes compared to Mysterium and Tachyon. Finally, Privatix only counts 8 nodes, which are likely provided by the Privatix team given they are very stable and, in our experience, it is currently impossible to contribute a node to this dVPN.

Next, we report on *where* dVPN nodes are located. Each stacked barplot in Figure 2(b) shows the top 5 countries per dVPN, as per 05-17-21. We omit Privatix whose eight nodes are located in: Toronto, Frankfurt, London, Bangalore, Amsterdam, Singapore, New York, and San Francisco. Note that Sentinel only distinguishes nodes by continent. The figure shows that, irrespective of the dVPN, the US (NorthAmerica for Sentinel) is the country where most nodes are located. Germany (DE) and Great Britain (GB) are two other popular countries among dVPNs. The geolocations of dVPN nodes are reported by the dVPNs, and bias may exist [51]. In addition, 75% of the Mysterium nodes are *residential*, whereas the percentage of residential node drops to 45% for Sentinel and 0% for Privatix. We were instead unable to retrieve such statistic for Tachyon.

Finally, we report on the *performance* – in terms of download speed and availability – when using such dVPNs. Results for upload bandwidths are omitted since they exhibit a similar trend, although about half of the bandwidth available, overall. Figure 2(c) shows the Cumulative Distribution Function (CDF) of the download speed measured each month per dVPN (plus ProtonVPN); each

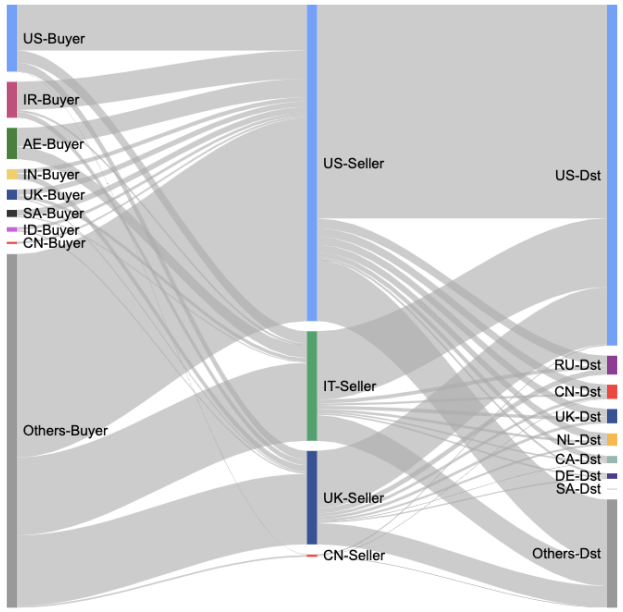


Fig. 3. Visualization of dVPN traffic across Mysterium, Sentinel, and Tachyon. Buyer's locations are shown on the left, our machines where dVPN nodes are run in the center, and traffic destinations on the right.

VPN was tested independently at night, while making sure no local cross traffic was present. The figure shows that only Tachyon is overall slower than ProtonVPN. Mysterium has comparable performance with ProtonVPN while both Sentinel and Privatix significantly improve bandwidth, by up to 3x and 6x. The legend of Figure 2(c) also reports the overall availability of each dVPN which is, on average, comparable with ProtonVPN. High bandwidth and perfect availability offered by Privatix further confirm that its 8 nodes are likely managed by Privatix itself.

Traffic Characterization – Between February and April 2021, our nodes have served ~505 thousand distinct buyers, ~632 thousand dVPN sessions, ~623 million TCP/UDP flows, accounting for about 16 TB of data. Download traffic is the highest contributor, about 10x the amount of upload traffic. Both residential and cloud dVPN nodes have attracted significant traffic over time (tens of thousands of sessions per dVPN), with the exception of the node located in Alibaba cloud (China) which has received no session via Sentinel, 604 sessions via Mysterium, and 1,924 sessions via Tachyon (see the bottom of Figure 3, CN-Seller). This is due to the interference of the great firewall [55].

Figure 3 visualizes from *where* dVPN traffic originates and is destined to, using Maxmind [10] to map `ip_src` and `ip_dst` at the country level. The middle of the plot shows the 10 machines – distributed between US, Italy, UK, and China – which were used for passive data collection. The figure aggregates data across the three dVPNs since no statistically meaningful difference was observed. The figure shows that the US has the most buyers, followed by Iran (IR), United Arab Emirates (AE), India (IN), and the UK, to complete the top 5 buyer locations. The US is also the most popular destination regardless of which node (middle of the plot) is used, accounting for over half of the traffic. Russia (RU) is the second most popular destination, followed by China (CN), UK, and Netherlands (NL).

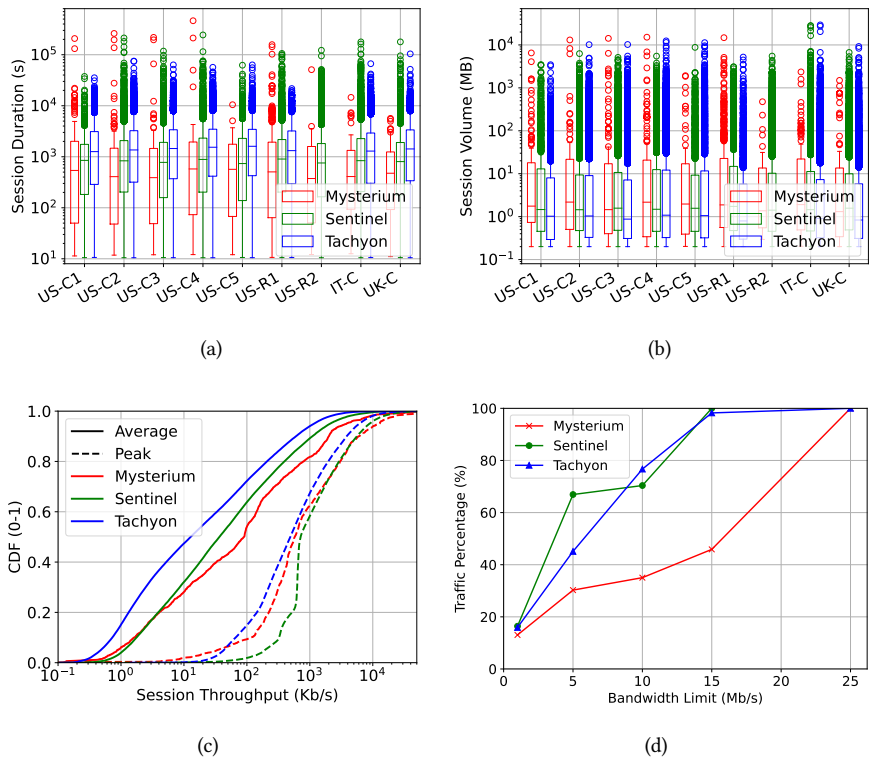


Fig. 4. Analysis of dVPN sessions: (a) duration, (b) volume, and (c) throughput. 9 node locations over 3 months; and (d) traffic percentage (ratio to the maximum observed) over one month as a function of a dVPN node available bandwidth.

It is noteworthy that the traffic destinations (right of Figure 3) may be biased because of the presence of Content Delivery Networks (CDNs), which distribute the content to servers all over the world and allow users to retrieve the content from the closest location. That is to say, the destination of the same content may vary depending on the geolocation of the requesters, which are our nodes in this case. To quantify the impact of CDNs, we i) perform `whois` to check the registration of the destination IPs, and ii) validate whether the IP addresses of the destinations fall in the range of CDN services when the company provides both cloud and CDN services, including Amazon [9], Google [3], and Microsoft [18]. We find that, depending on the node, traffic directed to CDNs ranges between 24 and 32%. Further, the most popular CDN providers, in descending order, are: Facebook, Akamai, and Cloudflare.

Next, we investigate *duration* and *volume* of the 629,156 dVPN sessions handled by our 9 nodes (see Figure 4[3]), omitting the Chinese node due to the lack of traffic discussed above. The figure shows two main results. First, user sessions are quite similar across nodes, with no significant difference apart from the tails (outliers in the boxplots). Second, user sessions are instead quite different among dVPNs, with Mysterium's sessions being overall shorter (median of 10 minutes versus 23 minutes for Tachyon) but carrying more traffic (median of 2 MB versus 1 MB for Tachyon). The latter result also implies that a large number of sessions (50% or more) are mostly idle. Nevertheless,

[3]US-R2 is an ARM-based machine and thus does not support Tachyon, see Table 1.

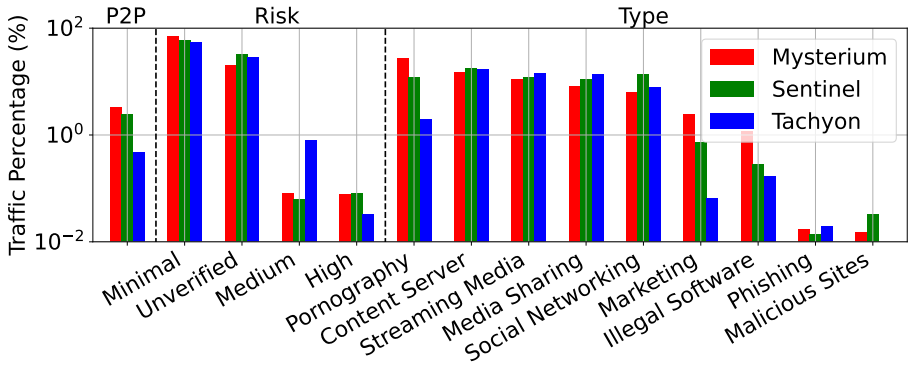


Fig. 5. DVPN traffic composition according to McAfee and IPP2P classification.

many sessions carry a large amount of traffic, even up to multiple GB. Overall, the average sessions have a volume of 120, 60, and 40 MB for Mysterium, Sentinel, and Tachyon, respectively.

To further investigate the previous result, we derive the session *throughput* as the number of bytes transferred during a session divided by its duration. We further compute the throughput for each of the TCP/UDP flow within a session and select the maximum (*peak*) as an approximation of the bandwidth available to a user. Discounting our nodes upload bandwidth (since they are very well provisioned), the session throughput depends on two factors: i) user access bandwidth, ii) application demand, *e.g.,* if a user is reading an article online for a long time, very little traffic would be measured. Given that we did not notice significant difference among our nodes, Figure 4(c) shows CDFs of both session and peak throughput (per dVPN) among all our nodes. The figure shows a large discrepancy between session and peak throughput suggesting that application demand is the main bottleneck, *i.e.,* users are mostly downloading from time to time rather than, for instance, streaming some video.

As a next step, we limit the upload bandwidth of our nodes and investigate the impact on the number of sessions. To do so, we set up, for one month, 5 additional AWS EC2 machines where we limit the bandwidth to 1, 5, 10, 15, and 25 Mbps for each dVPN. Note that the available bandwidth of dVPN nodes is not shown on the buyers' app. However, the buyers will likely switch to another node if the bandwidth of a connected node does not satisfy their needs. While it is natural to think that limiting the bandwidth would result in decreasing the number of sessions, Figure 4(d) shows that buyers from different dVPNs react quite differently. Mysterium buyers are the most sensitive to the bandwidth limits. For example, the number of Mysterium sessions is more than halved when implementing a 15 Mbps limit, which has no impact on both Sentinel and Tachyon buyers. This behavior is likely driven by a more demanding user-base, with overall higher bandwidth requirements (Figure 4(c)). Given all our nodes are equipped with more than 25 Mbps, this result further corroborates the above assumption that our nodes are not the reason of the trends observed above with respect to session characteristics.

Last, we characterize dVPN traffic composition according to classification based on McAfee and IPP2P (see Section 3.2). Figure 5 shows, for each dVPN, the amount of traffic belonging to each category; the dashed vertical lines group traffic in higher level categories (P2P, reputation, and type). Regardless of the dVPN, P2P traffic is extremely low, accounting for less than 3% of the overall traffic. With respect to the traffic *reputation*, the figure shows that the majority of the traffic carries very low risk: 60-70% minimal risk and 20%-30% unverified, or in between minimal and

medium risk according to McAfee classification. Medium and high risk are quite small and account for less than 1%.

With respect to content *type*, no (broad) category dominates the traffic. A big difference among dVPN arises when considering pornography, which accounts for 27% and 12% of the Mysterium and Sentinel traffic respectively, while it only accounts for 2% of the Tachyon traffic. Only a minority of traffic falls into the *malicious* category (less than 2%). In this category, "illegal software" is the most popular sub category, followed by "phishing" and "malicious websites".

5 ON THE VALUE OF SPARE BANDWIDTH

In this section, we first leverage the previous analysis to derive several assumptions which allow us to solve the optimization problem described in Section 2. Then, we derive optimal buyer's cost and seller's income, and conclude by commenting on the value of spare Internet bandwidth.

5.1 Buyer Cost Analysis and Optimization

Methodology – We investigate a buyer's minimal cost following Equation 2 from Section 2. We first need to filter the sellers based on a buyer's *constraints*, and then find the seller asking the lowest price for her bandwidth. We proceed as follows. For a given marketplace, we obtain the list of sellers as reported by our active crawler. Next, we remove the sellers which provide less bandwidth than b, the bandwidth requested by a user, and are not in the desired locations L. As per Equation 2, we should also filter sellers that block access to the set of domains contacted by a buyer. We skipped this step because blocklist usage is not publicly available for any of the dVPNs.[4] Finally, we choose the seller with the lowest cost among the remaining sellers. If the payment scheme of the marketplace is solely based on the amount of data transferred, *e.g.,* Sentinel, then we choose the seller asking the lowest price. If the payment scheme depends on both the amount of data transferred and the connection duration, *e.g.,* Mysterium, we need to consider the bandwidth needs b of the buyer as well. That is, minimizing the objective function $(x \cdot b \cdot u + y \cdot u)$ (Equation 2), where x, y are the prices (of data transferred and connection duration) and u is the connection duration.

With the existing dVPN apps, the buyers first retrieve the list of all nodes and then manually select a node. By default nodes are ordered by price, *i.e.,* the top of the list shows the cheapest nodes. Hence the above strategy can partially be realized with the current dVPN apps, *i.e.,* the buyers can filter nodes by a given country and manually implement such node selection. However, only "signals" (bad, medium, good) of bandwidth but no exact bandwidth numbers of the dVPN nodes are available in the existing dVPN apps.

Results – Figure 6(a) shows the CDF of the seller's prices available to buyers across marketplaces. Prices refer to a sample collected by the active crawler (05/01/2021) and are normalized relative to the lowest price allowed by a dVPN, *e.g.,* given the lowest price for Sentinel is 1 SENT per Gigabyte, then a relative price of 50 indicates 50 SENT per Gigabyte. The figure shows that 40% of Tachyon sellers ask the minimum price (0.22 IPX/GB), and only few (10%) dare to increase the price to 3x the minimum (up to 0.62 IPX/GB). Conversely, most Mysterium and Sentinel sellers (50-90%, respectively) request the default price (50 SENT and 22 MYST), while the remaining sellers equally split between providing *much* lower or higher prices (70x for Mysterium and 85x for Sentinel). It is worth noting that Mysterium sellers treat "price per min" differently from "price per GB", mostly

[4]While we could actively test whether a dVPN node block some high-risk traffic, we opted to avoid this experiment for two reasons. First, it would provide a very coarse approximation of existing blocklists. Second, it involves injecting high-risk traffic, which is unethical and potentially illegal.

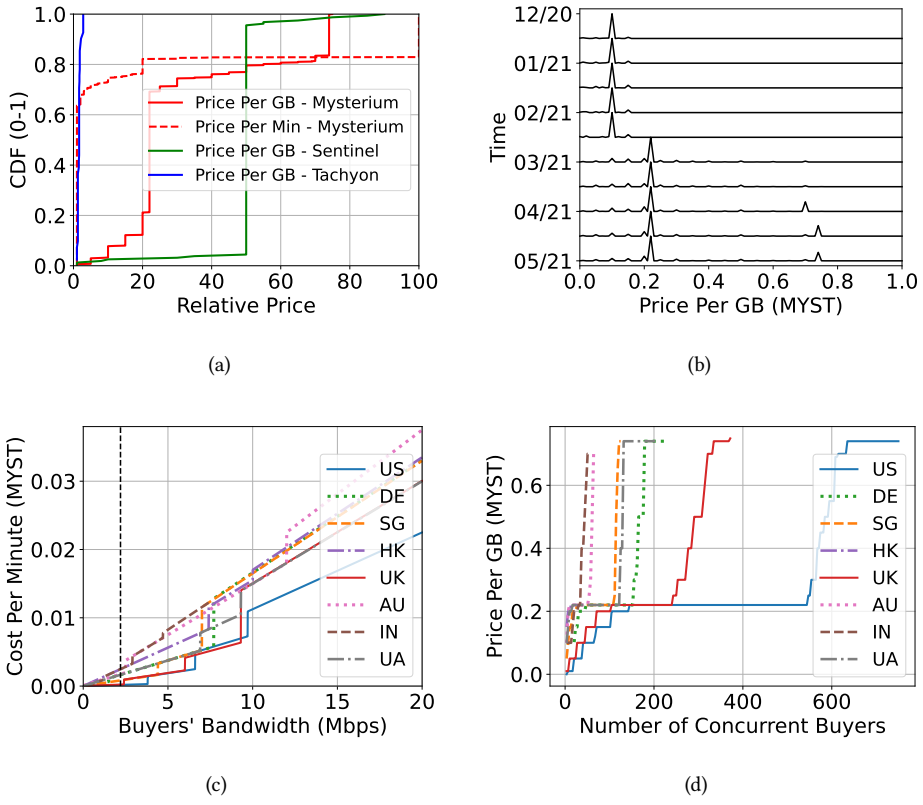


Fig. 6. Price and buyer's cost analysis. (a) CDF of relative price (to the lowest) per marketplace. (b) 6-months PDFs of Mysterium prices. (c) Buyer's minimal cost in Mysterium given a target country and bandwidth. (d) Mysterium's price per GB as a function of the number of concurrent buyers.

requesting the minimum price. This behavior is an indication that complex pricing schemes can be hard to grasp by sellers.

The above result suggests that both Tachyon and Sentinel are not yet realistic bandwidth marketplaces, despite their large footprint (see Figure 2(a)). In fact, their sellers just ask either the minimum (Tachyon) or the default (Sentinel) price unmotivated by the lack of payment as well as cost for the buyers. In the rest of the analysis we only focus on Mysterium since it is currently the most mature bandwidth marketplace.

We start by analyzing the *history* of Mysterium's seller prices. Figure 6(b) shows the historical frequencies (PDF) of MYST per GB over 5 months. Mysterium had a major update in late February 2021 when the default price increased from 0.1 to 0.22 MYST per GB, as indicated by the PDF shift in Figure 6(b). While many sellers (48%) stick to the default price setting, the number of sellers offering cheaper prices, comprised between 0.01 and 0.22 MYST, grew by 11% and currently account for 21% of the sellers. Further, the figure shows a new peak around 0.7 (04/2021) and 0.74 MYST (05/2021) which account for about 18% of sellers.

We now investigate the question: *what is the minimum buyer cost?* We assume Mysterium buyers with variable bandwidth requirements (between 1 and 20 Mbps, which is mostly an upper-bound as per Figure 4) and interest in several countries. We choose countries whose trend in price setting is

representative of most other countries in their *regions*, *e.g.,* Germany (DE) for Europe. We assume a buyer can always select the seller which meets his/her constraints, at the minimum price. We will relax this constraint later.

Figure 6(c) shows the minimal buyer cost as a function of both bandwidth and location. We express the buyer cost as *cost per minute*, to incorporate in a single metric both pricing schemes adopted by Mysterium. The cost per minute is the sum of the cost of the GB transferred in a minute, given a target bandwidth, and the cost for such duration. The figure shows a significant impact of the selected country on a buyer's cost: up to 10x when comparing the cheapest country (US, UK) with the most expensive one (India). Given that many countries offer high bandwidth, there is potential of savings for buyers who are not interested in a specific location, *i.e.,* they leverage a dVPN mostly for privacy.

When focusing on bandwidth requirements, Figure 6(c) shows that the cost mostly grows linearly as the bandwidth increases. This is simply because higher bandwidth requires more data per minute. However, we also observe some non-linear "jumps". For instance, the minimum cost for acquiring less than 3.7 Mbps in the US is 0.01 MYST/GB + 0.00001 MYST/min. When higher bandwidth is requested (between 3.7 and 6.6 Mbps) the minimum price available is 5x higher, or 0.05 MYST/GB + 0.00005 MYST/min. Similar patterns apply to other countries, where there are some cheap sellers with relatively small bandwidth capacity, while it costs more to acquire higher bandwidth. Some countries have relatively low bandwidth offerings, *e.g.,* the highest bandwidth provided in India (IN) is only 7.6 Mbps.

Next, we assume N concurrent buyers. Each new buyer consumes a portion of the available bandwidth at a seller (see Figure 2(c)), and will thus eventually impact the decision of future buyers. For this analysis, we assume each buyer requires 2.2 Mbps (average peak bandwidth from the distribution described by Figure 4(c)). Figure 6(d) shows, for several locations, the minimum cost for the N-th user as a function of N, or the number of concurrent buyers. As the load on the marketplace increases, new buyers are left with more expensive sellers, and thus with a bill which grows, overall, by 70x. The cost increases faster in countries with overall less bandwidth for sale. For example, 100 concurrent buyers are enough to force new buyers to pay 0.7 MYST per GB in SG and UA. Conversely, the US can support up to 600 concurrent buyers before reaching such a high price.

5.2 Seller Income Analysis and Optimization

Methodology – We investigate a seller's maximum income following Equation 4 (Section 2). For the same reason as in Section 5.1, we ignore the impact of blocklists. It follows that to solve the objective function we only need to calculate the expectations of the traffic volume $\mathbb{E}[n \cdot B \cdot U]$ and duration $\mathbb{E}[n \cdot U]$. We approximate the distributions of the variables (n, B, U) by their best fitting functions over the data we have collected. For example, the distribution of the dVPN session duration (Figure 4(a)) is fitted by function $y = \frac{1}{ax^b+c}$.

For the sake of brevity, we leave detailed reasoning and formula derivations in Appendix B. Denoting $f(b)$ as the PDF of the buyers' bandwidth b, and $H(r)$ as the percentage of buyers left under bandwidth limit r compared to no limit, *i.e.,* Figure 4(d), the seller's objective function is:

$$\begin{aligned} &\max_{x,y,r} \mathbb{E}[\mathbf{U}] \cdot H(r) \cdot N'(x,y,l) \cdot (x \int_0^r bf(b)db + y) \\ &s.t.\ \ \mathbb{E}[\mathbf{U}] \cdot H(r) \cdot N'(x,y,l) \int_0^r bf(b)db \leq D/2 \end{aligned} \tag{5}$$

Results – Figure 7 shows the average number of sessions (black markers) and total income (orange markers) they produce for Mysterium sellers under variable pricing and bandwidth limits. Each point further shows minimum and maximum value of each metric as errorbars. The dashed lines

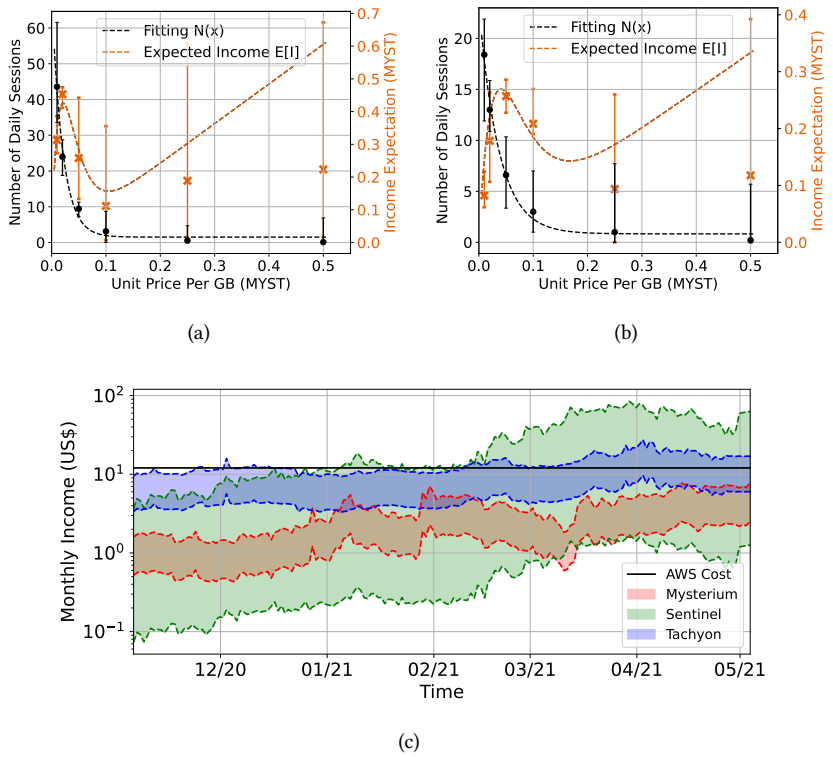


Fig. 7. Mysterium's seller income analysis as a function of price settings assuming a node bandwidth of: (a) 25Mbps and (b) 5 Mbps. (c) Evolution over six months of default and optimized monthly income per marketplace.

represent a fitting function of daily sessions ($N(x) = ae^{x-b} + c$, black) and the theoretical income expectation (Equation 5, orange). These results are derived from a one-month experiment where we fix the location l to be the US, set no data cap $D = \infty$, and allow all traffic types. To bound the number of variables, we investigate 6 different prices per GB (0.01, 0.02, 0.05, 0.1, 0.22, and 0.5) MYST but fix the price per minute to the minimum, *i.e.,* the most popular option as suggested by Figure 6(a). For the nodes bandwidth we select 5 and 25Mbps, which are *low* and *average* bandwidth currently offered by Mysterium nodes in the US.

In presence of high bandwidth (Figure 7(a)), the figure shows that the number of daily sessions quickly decreases as the price increases. For example, a 10x price increase (from 0.01 to 0.1 MYST per GB) causes the average number of daily sessions to drop from 44 down to about 4 sessions, and then eventually near zero as the price keeps increasing. This behavior causes the income to be fitted by a non-linear function: the income grows when the price doubles from the minimal (from 0.01 to 0.02 MYST per GB), then decreases (between 0.02 and 0.22 MYST per GB), and finally flattens out (between 0.22 and 0.5 MYST per GB). A similar trend is observable also when we limit the node bandwidth to just 5Mbps. In this case, the node attracts less than half of the sessions but it is also less penalized by a price increase, *i.e.,* peaking at 0.05, or twice as much as before. It follows that, at its peak value, the daily income amounts to 0.28MYST versus 0.45MYST with 25Mbps, or just a 40% reduction in presence of an 80% bandwidth reduction. This result can be due to many reasons, but mostly indicates that a small portion of Mysterium buyers tend not to optimize their

Table 2. The value of spare Internet bandwidth in the US.

	Time (Hr)	Current MYST ($)	Optimized MYST ($)
Buyer's Cost	Average	0.18 (0.11)	0.02 (0.012)
	Most Popular	-	0.026 (0.016)
	Least Popular	-	0.015 (0.009)
Seller's Price	Average	0.23 (0.14)	0.028 (0.017)

costs, either because they purposely pay more for better experience or they do not optimize their node selections, *e.g.,* filter their nodes by pricing and bandwidth.

We now focus on the accuracy of our theoretical income expectation. The figure shows that when the price per GB is equal or lower than 0.1 MYST, the theoretical results fit the measurement results quite well. However, theoretical and actual incomes diverge when $x > 0.1$, *e.g.,* the theoretical income overestimates the actual income by 3x, compared to the average, with a price of 0.5 GB per MYST. This is because at this price point there are mostly no sessions per day, with the exception of few days which cause high variance, *e.g.,* between 0 and 0.7 MYST with a price of 0.5 GB per MYST and 25 Mbps (Figure 7(a)). Due to such high variance, the fitting function for N(x) is quite coarse in this range.

When comparing the latter result with the current pricing adopted by Mysterium sellers (Figure 6), we find that most sellers are making suboptimal decisions with respect to their income optimization. Figure 7(c) shows, for each marketplace, the lowest (lower bound) and optimized (high bound) incomes over 6 months. Each monthly income is derived using the same strategy as in Figure 7(a); we then derive income value in dollars based on the cryptocurrency value during each day of the month, with the goal to visualize its high volatility. The figure shows that a careful mechanism to optimize seller's price offers significant income increases, comprised between 3x (Mysterium) and 50x (Sentinel). However, even assuming optimal income, the monthly income is mostly below $10 (both over time and across dVPNs) or the cost of renting an AWS instance to act as a dVPN node. However, if the increasing trend of cryptocurrency values continue, hosting a dVPN may become more and more profitable. Similarly, it seems that Mysterium has realized that the default pricing scheme is currently too low, as suggested by the change in the default price as observed in Figure 6(b). The figure also shows that the volatility of cryptocurrency can make one marketplace more profitable than another, over time. For example, Mysterium has been filling the (income) gap with respect to Tachyon over the period of our measurement.

It has to be noted that the above price optimization only applies to a precise period of time. In reality, the optimal price is not fixed but should be evolving over time, influenced by the instantaneous choices of both the buyers and the sellers, as well as the presence of alternative marketplaces. This motivates us to build a system, in the upcoming section, which is capable of adjusting seller's "settings", *e.g.,* pricing and traffic per marketplace, to optimize their income over time.

5.3 Discussion

We finally comment on the value of spare Internet bandwidth leveraging the data we have collected and model we have proposed. We focus on Mysterium – since it has shown to be a mature bandwidth marketplace – and the US, which is currently the largest market (see Figure 2(b)) and the location where we conducted our pricing and bandwidth experiments (see Section 3.2).

By the end of our measurements, Mysterium counts 302 sellers (Figure 2(b)) in the US asking an average price of 0.23 MYST/GB. Using function N(x) from Figure 7(a), *i.e.,* assuming 25 Mbps or the average bandwidth offered by nodes in the US, we estimate that these nodes currently attract, in total, 629 daily buyers, that spend between 0.01 and 0.74 MYST/GB (0.18 MYST/GB on average).[5] It follows that the value of spare Internet bandwidth (in the US) lies between 0.18 MYST/GB (average price paid by the buyers) and 0.23 MYST/GB (average price requested by the sellers), which corresponds to $0.11-0.14 per GB, given the cryptocurrency value then.

Next, we explore the effect of the optimization of buyer's cost and seller's income on the value of spare bandwidth, independently. Figure 4(a) shows that the average buyer session lasts about one hour. Assuming 629 daily buyers, equally distributed throughout the day, then there are, on average, 25 concurrent buyers interested in a US node at any point in time ($629 \times 3,499s/24h \approx 25$). We then apply the methodology described in Section 5.1 (Figure 6(d)), where each buyer selects the cheapest seller who has enough spare bandwidth to satisfy her demand. After this optimization, the buyers will pay between 0.01 and 0.05 MYST/GB (0.02 MYST/GB, on average), or a 10x reduction compared to today (0.18 MYST/GB). We then take a closer look at the time of the day. Our measurements indicate that there are 33 and 18 concurrent buyers during the most and least popular hour, respectively. Table 2 shows the optimized buyer's costs in these scenarios. From the seller's perspective, we have previously derived optimal price setting of 0.028 MYST/GB (Figure 7(a)), which is 8x smaller than the average price (0.23 MYST/GB) that the sellers are requesting today. Note that the table does not show optimal seller's price at different times of the day since we did not observe significant shift. These discrepancies suggest that neither the buyers nor the sellers are optimizing their costs/incomes.

From the above analysis, we conclude that the value of spare US Internet bandwidth lies between $0.11 and $0.14 per GB. However, both buyers and sellers have room to move this price and optimize either their cost or income. This result was obtained considering the optimization of a buyer's cost or seller's income, in isolation. In reality, one would affect another. For example, the value of bandwidth would decrease if more sellers optimize their incomes because the bandwidth demand is currently less than the supply, *i.e.,* we estimate 25 concurrent buyers when the sellers can support over 600 (Figure 6(d)). Further analysis would require many other assumptions, *e.g.,* rationality of the buyers and sellers, and is out of the scope of this paper.

6 A MULTI-VENDOR BANDWIDTH MARKETPLACE

This section translates results from the previous section in a concrete system, *RING*, which helps sellers to *maximize* their income while participating to multiple bandwidth marketplaces. We start with a quick extension of the bandwidth monetization problem in the context of a multi-vendor marketplace. Next, we detail design and implementation of RING. We then conclude the section by showing how RING operates.

6.1 Multi-Vendor Bandwidth Market Optimization

Consider a seller who joins M marketplaces concurrently. For a marketplace $i \in [1, M]$, we denote by B_i and U_i the probability density functions of bandwidth and duration characterizing its buyer sessions. Next, we denote by x_i and y_i the prices for traffic volume and session duration, per marketplace i. Finally, we call r_i the maximum bandwidth allowed per marketplace and D the *total*

[5]More specifically, from the active measurements, we know the number of sellers M given any price x ranging from 0.01 to 0.74 MYST/GB. For instance, $M(0.01) = 4$ means that there are 4 sellers offering at 0.01 MYST/GB. We also know the number of buyers N given a price x per day. For instance, $N(0.01) = 43$ means that a seller with 0.01 MYST/GB attracts 43 buyers per day. Let T_s be the average buyers' session time. We then can estimate the number of concurrent buyers in the US by $(M(0.01) * N(0.01) + M(0.02) * N(0.02) + \ldots + M(0.74) * N(0.74)) * T_s/24h$.

Algorithm 1: Sellers' Income Optimization Heuristic

Input : $DVPNs[1, M]$. For $i \in [1, M]$, Rate Limit r_i, Price x_i, Data Caps D, D_i, Income I_i, Cumulative Consumed Data CC_i, Last Consumed Data LC_i, Left Time T_{left}.

1 $I \leftarrow \sum_i^M I_i$
2 **for** $i \in [1, M]$ **do**
3 $\quad D_i \leftarrow D_i + \alpha(D\frac{I_i}{I} - D_i)$ // Adjust the data cap
4 $\quad$ Data demands $d_i \leftarrow LC_i \cdot T_{left}$
5 $\quad$ **if** $d_i > D_i - CC_i$ **then**
6 $\quad\quad$ Decrease r_i and/or increase x_i // decrease demands
7 $\quad$ **else**
8 $\quad\quad$ Increase r_i and/or decrease x_i // increase demands
9 $\quad$ **end**
10 **end**

Output : Rate limits r_i and price settings x_i.

data cap, *i.e.,* the sum of all data caps D_i per marketplace. The objective function from Equation 4 becomes a cross optimization of the total income from multiple marketplaces (see Equation 6):

$$\begin{aligned} &\max_{x_i, y_i, r_i} \sum_i^M (x_i \cdot \mathbb{E}[\mathbf{n}_i \cdot \mathbf{B}_i \cdot \mathbf{U}_i] + y_i \cdot \mathbb{E}[\mathbf{n}_i \cdot \mathbf{U}_i]) \\ &s.t. \quad \sum_i^M \mathbb{E}[\mathbf{n}_i \cdot \mathbf{B}_i \cdot \mathbf{U}_i] \leq D/2 \end{aligned} \tag{6}$$

In reality, it is challenging to collect the information needed to solve Equation 6. Based on the insights we have gained from the optimization of the seller's income in a single marketplace (see Section 5), Algorithm 1 proposes a local heuristic to approximate the solution of the above optimization.

First, we calculate total (across marketplaces) income for all dVPNs (L1 of the algorithm) in a time interval T, *e.g.,* one hour. The interval should be neither too short, since changing the settings requires rebooting the dVPN (thus interrupting all ongoing sessions), nor too long which slows down algorithm's convergence to the optimal settings. We have tested several time intervals and found that one hour is appropriate. The total data cap D is a user provided constant, which is initially equally distributed among marketplaces (D_i). Each marketplace may generate different income due to, for instance, the current value of its cryptocurrency. We thus adjust the data cap for each marketplace to maximize the income. When a marketplace has generated more income per GB than the average income per GB for all marketplaces ($\frac{I_i}{D_i} > \frac{I}{D}$), this is an indication that its data cap D_i should be increased. Otherwise its data cap should be decreased. We iterate across marketplaces and adjust their data cap (L4), where the *aggressiveness* of data cap reallocation is determined by coefficient α, *i.e.,* when $\alpha = 1$, the data cap is adjusted purely based on what happened in the last hour.

Next, we adjust the bandwidth limits and/or prices to maximize the profit for each marketplace. Section 5 suggests that the key to maximize a seller's income is to adjust bandwidth limit and price such that the buyers' bandwidth demand fulfills the seller's data cap. We calculate the buyers' bandwidth demand based on consumed data in the last hour (L5). Then, we compare the derived bandwidth demand with the leftover data cap (L6-L10). If the bandwidth demand exceeds the data cap, then we either increase the price or decrease the rate limit if the marketplace allows charging for session duration (*i.e.,* $y_i > 0$). In fact, according to Section 5 this would allow to reduce the

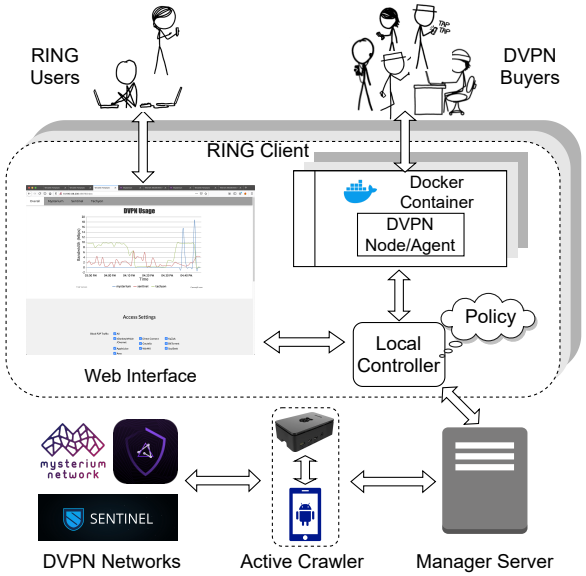


Fig. 8. Visual representation of RING.

bandwidth demand and increase income. Otherwise, we either decrease the price or increase the rate limit to attract more buyers, and thus increase income.

6.2 Design and Implementation

RING's design is motivated by four goals. First, allow a seller to *concurrently* join multiple marketplaces (dVPNs). Second, provide intelligence to *maximize* a seller's income. Third, provide *fine-grained* control on permitted traffic to limit the danger of running a dVPN node. Fourth, *ease of use*: currently, mostly expert users can deploy dVPN nodes due to lack of executables across OSes, complex setup, etc.

Figure 8 shows RING's architecture with its three main components: *client*, *manager server*, and *crawler*. The crawler is the same one described in Section 3.2: it periodically crawls the set of supported dVPNs to fetch information like available nodes, and current pricing. The client controls and monitors multiple dVPN nodes running at the user machine. It fetches up-to-date information about the dVPNs and makes decisions for local bandwidth allocation and price settings. Below, we provide details on RING's client and its traffic control.

RING Client – We adopt Docker containers [34] which allow us to run dVPN nodes concurrently and in isolation. We create Docker virtual network interfaces which eases traffic monitoring and rate limiting per dVPN. We build Docker images from each dVPN up-to-date source code to support Raspberry Pi (ARM), which we envision as the perfect plaform for RING's clients – a small and cheap box to attach to the home router.

RING's client can be managed by a Web interface. This interface makes it possible to customize each dVPN, *e.g.,* by providing crypto-wallet addresses, speed limits, data cap, and allowlists. Further, the interface shows several useful statistics, *e.g.,* bandwidth consumed by each dVPN as shown in Figure 8. The Web interface (locally) communicates with a controller which executes user input, *e.g.,* starts/pauses/stops a specific dVPN or updates the price settings. RING's client can also set timer and choose specific times of the day to be a part of the marketplace. Other important user

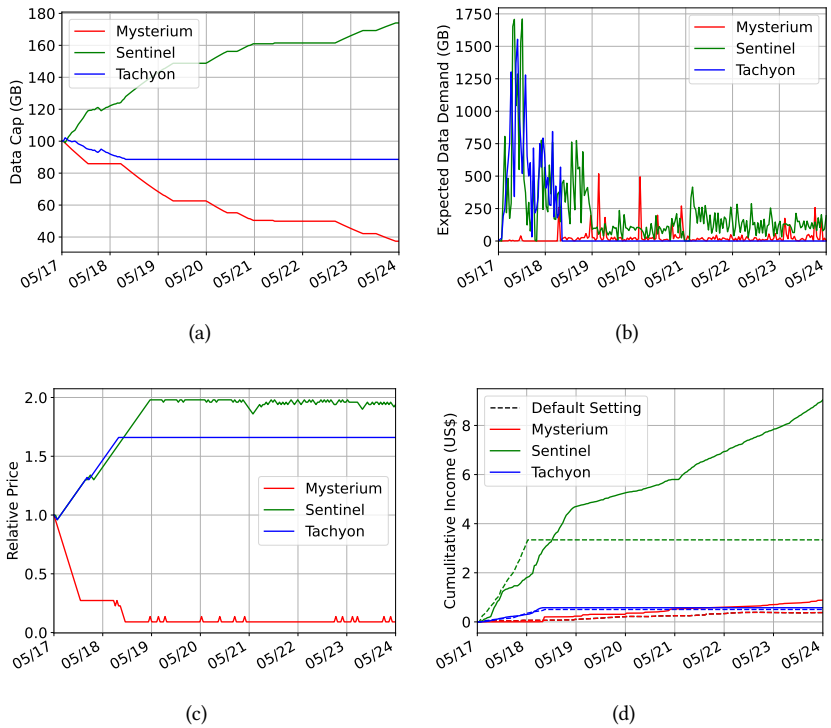


Fig. 9. RING's preliminary evaluation over one week: (a) data cap adjustments, (b) hourly forecast of expected traffic volume, (c) price decisions, and (d) cumulative income.

inputs are *rate limits*, which translate into Linux TC [25] rules, and *accesslists*, which translate into iptables. Next, we offer more details on how accesslists are implemented.

Traffic Control – RING's Web interface allows users to only allow low, medium, high risk traffic [2], or select a set of content categories allowed, *e.g.*, avoid gambling and pornography. An option to block P2P traffic – identified using IPP2P – is also provided. To generate such iptables rules, we rely on the domain classification described in Section 3.2: SNI + DNS data matched using McAFee database [2] which achieves, in our data-set, >95% domain coverage. Such rules are maintained at the manager and updated regularly.

6.3 Preliminary Evaluation

We demonstrate the functioning of RING via controlled experiments (over one week) using two AWS EC2 machines in the US: one machine running RING's heuristic and one using default prices and no control on data cap and bandwidth limit. We assume a 300 GB weekly data-cap, initially equally divided between the three dVPNs. We derive income for each controlled user based on the amount of traffic they carried, their price settings (either default or using RING price adjustments), and the cryptocurrency value. In the case of Mysterium, we also validate such computed income against the official figures reported by Mysterium. This cannot be done for Sentinel and Tachyon which are still in development and do not yet release payments to their users. In the following, we first discuss the decisions made by RING's heuristic and then compare the income generated by the two machines.

Figure 9(a) shows the evolution over one week of the data cap allocation per dVPN realized by RING, which is automatically adjusted based on the computed income per GB. The figure shows that Sentinel's data cap almost doubles over one week, from the starting 100 GB up to 170 GB, at the expense of Tachyon (down to 90 GB) but mostly Mysterium (down to 40 GB). This implies that Sentinel is bringing the most income per GB, as previously shown in Figure 7(c).

Figure 9(b) shows a hourly forecast of the expected traffic volume for the week given the rate measured in the last hour. The figure shows that, within two days, both Sentinel and Tachyon attract significant amounts of traffic which would exceed their weekly data caps. Their prices are then increased (Figure 9(c)) which effectively reduces the demand (Figure 9(b)) while increasing income (Figure 9(d)). The opposite behavior is observed for Mysterium; as previously discussed, Mysterium's default price is too high and a price reduction can attract more traffic (see Figure 7). Because we adjust price and bandwidth limit conservatively, Tachyon runs out of data cap on the second day. With appropriate price setting, Mysterium and Sentinel show data demands (50 and 150 GB) that closely approximate their data caps.

To conclude, Figure 9(d) shows the cumulative income generated using both default settings for each dVPN, and RING's heuristic. Because of the high demand shown in Figure 9(b), Sentinel and Tachyon with default settings run out of data cap on the second and third day respectively, generating a total income of $3.7. In contrast, RING allocates a larger data cap to Sentinel and also increases its price, generating a combined (Sentinel plus Tachyon) income of $10.5. For Mysterium, RING's price adjustment allows to double its income compared to the default setting ($0.9 versus $0.4).

This preliminary evaluation shows that RING achieves higher income than default dVPNs settings. It is noteworthy that the evaluation only reveals partial capability of RING. With the flexibility to deploy more advanced optimization algorithms and the ability to include more dVPNs, RING has a high potential in benefiting the bandwidth sellers. We plan to release RING in the near future and explore the multi-vendor bandwidth marketplace in depth.

7 RELATED WORK

Several research studies quantify the presence of spare bandwidth. In particular, a measurement study showed that a typical U.S. household does not use most of its bandwidth while streaming and gets marginal gains from upgrading speeds [27, 40]. Despite such findings, user studies have shown that reliability and speed are most important for consumers [52, 53]. This inclination towards high speeds, enhanced by the proliferation of bandwidth-hungry applications, further impacted by the necessary variability in traffic demand, leaves often significant portions of bandwidth unused. Hence, monetization opportunities, which we analyzed in this paper, arise.

Data caps are a method ISPs use to protect against heavy-hitters [31] or to limit user activity in resource-constrained networks such as cellular networks [41]. There has been research on the implications of data caps on the user experience. For example, in [44] the authors explored the effects of data caps on home Internet usage in urban South Africa to show that users have three uncertainties with regards to their bandwidth usage: invisible balances, mysterious processes, and multiple users. Our paper considers a different scenario, the one where dVPN nodes operate under data caps, yet because these caps are unlikely to be reached, they monetize spare bandwidth. ISPs are faced with a tradeoff – make their plans less attractive by reducing data caps and negatively affect users and their own revenues, or enable larger caps to attract users and consequently enable bandwidth monetization.

There has been research on understanding the users that are willing to pay more for bandwidth. Necessarily, such a willingness is positively related to income and other technological attributes and negatively related to socio-demographic attributes such as habitat and age [43]. Another study

found that there exists a significant variability in the sense that certain kinds of users are willing to pay substantially more than others [57]. Such a variability exists with dVPNs, which are currently in their "infancy." Hence, there is a narrow group of consumers involved in the spare bandwidth market. Nonetheless, this market is likely to grow in the coming years and become mainstream. In this paper we analyzed the key parameters that affect this emerging market.

There has been work on addressing bandwidth pricing among users and ISPs considering single- and multi-class scenarios [54]. Others have analyzed incentives for creating efficient *inter-ISP* bandwidth marketplaces [42] or pricing schemes among different to accomplish net neutrality [46]. Our work is distinctive because we effectively have Internet consumers both as buyers and sellers of the spare bandwidth market. A more similar scenario is the P2P system BitTorrent, where Internet users trade the download resources with upload capacity, *i.e.,* bandwidth for bandwidth. There has been work on studying the incentives of users in such a P2P system [47, 50]. Yet, our investigated market is different because the trading resources are not the same, *i.e.,* bandwidth trading with money, and the latter resource depends on geolocation, *i.e.,* purchasing power is different in different countires, and is potentially unlimited with respect to our investigated market. Also, unique issues affect such a market, *e.g.,* seller's location and willingness to serve a particular type of traffic.

Network censorship, *i.e.,* blocking traffic originated to and from particular applications or regions [39], is one of the main drivers behind the consumer-consumer bandwidth marketplace we explored in this paper. Typically, buyers come from censored regions and sellers reside in the reminder of the Internet. Our data (details omitted) confirm that this is indeed the case.

8 CONCLUSION

Residential Internet speeds have been increasing much faster than actual user needs, leaving up to 80% of today's residential bandwidth unused. This trend, in conjunction with the adoption of cryptocurrencies, has stimulated the deployment of bandwidth *marketplaces*, *i.e.,* systems where users can monetize part of their connectivity in exchange of some compensation. Distributed Virtual Private Networks (dVPNs) – a new form of VPN with no central authority – are popular instances of these marketplaces, with multiple providers and thousands of worldwide users. In this paper, we have presented the first comprehensive study of bandwidth marketplaces, using dVPNs as their incarnation. We *actively* and *passively* monitored three major dVPNs (Mysterium, Sentinel, and Tachyon) for 6 months, reporting on their footprint, performance, income opportunities, and traffic characteristics. Using this data, we estimated that the value of spare Internet bandwidth in the US ranges between 11 and 14 cents. Still, we found that both buyers and sellers utilize ad-hoc "rules-of-thumb" when choosing their prices, resulting in a sub-optimal marketplace. Indeed, we showed that a seller's income could be increased by setting a lower but optimal price which is likely to attract more buyers. We also predict that the value of spare bandwidth would be reduced when more sellers begin to optimize their income as the current bandwidth supply exceeds the demand. Finally, we formalized how a seller's income could be optimized in a *multi-vendor* marketplace. We also realized this abstraction in RING, the first such marketplace built on top of Mysterium, Sentinel, and Tachyon, which helped increase a node revenue by 63%.

ACKNOWLEDGEMENTS

We would like to thank our shepherd Gareth Tyson and the anonymous reviewers for their insightful feedback and guidance.

REFERENCES

[1] Alibaba Cloud. https://cn.aliyun.com/.
[2] Customer URL Ticketing System. https://www.trustedsource.org/sources/index.pl.
[3] Google CDN ip-range. https://groups.google.com/g/gce-discussion/c/V2n9Ri-T5qg.
[4] HTTPS encryption on the web. https://transparencyreport.google.com/https/overview.
[5] Internet Deals, Plans, and Pricing | Xfinity. https://www.xfinity.com/learn/internet-service/deals.
[6] Keliweb. https://www.keliweb.it/.
[7] Key Features of Tachyon Protocol. https://tachyon.eco/?n=yr8mtzfwee.WhatIsTachyon.
[8] Lethean: Absolute Internet Privacy. https://lethean.io/.
[9] Locations and IP address ranges of CloudFront edge servers. https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/LocationsOfEdgeServers.html.
[10] MaxMind: IP Geolocation and Online Fraud Prevention. https://www.maxmind.com/.
[11] Mysterium network: Censorship free Internet for all. https://mysterium.network/.
[12] OpenVPN. https://openvpn.net/.
[13] Orchid: The Crypto Powered VPN. https://www.orchid.com/.
[14] OVH Cloud VPS. https://www.ovhcloud.com/en/vps/.
[15] Privatix: Next-gen VPN. https://privatix.com/.
[16] ProtonVPN. https://protonvpn.com/.
[17] Reference Guide - McAfee TrustedSource Web Database. https://www.trustedsource.org/download/ts_wd_reference_guide.pdf.
[18] Retrieve the current POP IP list for Azure CDN. https://docs.microsoft.com/en-us/azure/cdn/cdn-pop-list-api.
[19] RFC 8446: The Transport Layer Security (TLS) Protocol Version 1.3. http://tools.ietf.org/html/rfc8446.
[20] RING: the first multi-vendors bandwidth marketplace. https://ringdvpn.com/.
[21] Sentinel: Secure yourself today with Sentinel. sentinel.co.
[22] Speedtest Global Index. https://www.speedtest.net/global-index.
[23] Speedtest Global Index - United States. https://www.speedtest.net/global-index/united-states.
[24] Tachyon VPN. https://tachyon.eco/.
[25] tc(8) — Linux manual page. https://man7.org/linux/man-pages/man8/tc.8.html.
[26] Testnet 1.0 ends as Testnet 2.0 unlocks new milestone. https://mysterium.network/blog/testnet-1-0-ends-as-testnet-2-0-unlocks-new-milestone/.
[27] The Truth About Faster Internet: It's Not Worth It, The Wall Street Journal, 2019. https://www.wsj.com/graphics/faster-internet-not-worth-it/.
[28] Tstat - TCP STatistic and Analysis Tool. http://tstat.polito.it/.
[29] VPN Gate - Public VPN Relay Servers. https://www.vpngate.net/en/.
[30] WireGuard: fast, modern, secure VPN tunnel. https://www.wireguard.com/.
[31] Xfinity Data Plans. https://www.xfinity.com/learn/internet-service/data.
[32] Android Debug Bridge (ADB), 2020. https://developer.android.com/studio/command-line/adb/.
[33] AWS EC2, 2020. https://aws.amazon.com/ec2/.
[34] Docker, 2020. https://www.docker.com/.
[35] Google Safe Browsing, 2020. https://safebrowsing.google.com/.
[36] Privatix Community Update, 2020. https://medium.com/privatix/community-update-c98df60f2c98/.
[37] Sentinel api. 2021. https://api.sentinelgroup.io/client/vpn/list.
[38] A. M. Antonopoulos and G. Wood. *Mastering ethereum: building smart contracts and dapps*. O'reilly Media, 2018.
[39] K. Bock, G. Hughey, L.-H. Merino, T. Arya, D. Liscinsky, R. Pogosian, and D. Levin. Come as you are: Helping unmodified clients bypass censorship with server-side evasion. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, pages 586–598, 2020.
[40] F. Bronzino, P. Schmitt, S. Ayoubi, G. Martins, R. Teixeira, and N. Feamster. Inferring streaming video quality from encrypted traffic: Practical models and deployment experience. *Proceedings of the ACM on Measurement and Analysis of Computing Systems*, 3(3):1–25, 2019.
[41] P. Casas, R. Schatz, F. Wamser, M. Seufert, and R. Irmer. Exploring qoe in cellular networks: How much bandwidth do you need for popular smartphone apps? In *Proceedings of the 5th Workshop on All Things Cellular: Operations, Applications and Challenges*, pages 13–18, 2015.
[42] I. Castro, A. Panda, B. Raghavan, S. Shenker, and S. Gorinsky. Route bazaar: Automatic interdomain contract negotiation. In *15th Workshop on Hot Topics in Operating Systems (HotOS {XV})*, 2015.
[43] L. Cerno and T. Amaral. Demand for internet access and use in spain, 2005. https://eprints.ucm.es/id/eprint/7897/1/0506.pdf.

[44] M. Chetty, R. Banks, A. Brush, J. Donner, and R. Grinter. You're capped: understanding the effects of bandwidth caps on broadband use in the home. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 3021–3030, 2012.
[45] S. Corbet, C. Larkin, B. Lucey, A. Meegan, and L. Yarovaya. Cryptocurrency reaction to fomc announcements: Evidence of heterogeneity based on blockchain stack position. *Journal of Financial Stability*, 46:100706, 2020.
[46] Y. Harchol, D. Bergemann, N. Feamster, E. Friedman, A. Krishnamurthy, A. Panda, S. Ratnasamy, M. Schapira, and S. Shenker. A public option for the core. In *Proceedings of the Annual conference of the ACM Special Interest Group on Data Communication on the applications, technologies, architectures, and protocols for computer communication*, pages 377–389, 2020.
[47] D. Levin, K. LaCurts, N. Spring, and B. Bhattacharjee. Bittorrent is an auction: analyzing and improving bittorrent's incentives. In *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, pages 243–254, 2008.
[48] F. J. Massey Jr. The kolmogorov-smirnov test for goodness of fit. *Journal of the American statistical Association*, 46(253):68–78, 1951.
[49] D. Nobori and Y. Shinjo. Vpn gate: A volunteer-organized public vpn relay system with blocking resistance for bypassing government censorship firewalls. In *11th USENIX Symposium on Networked Systems Design and Implementation (NSDI'14)*, pages 229–241, 2014.
[50] M. Piatek, T. Isdal, T. Anderson, A. Krishnamurthy, and A. Venkataramani. Do incentives build robustness in bittorrent. In *Proc. of NSDI*, volume 7, page 4, 2007.
[51] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye. Ip geolocation databases: Unreliable? *ACM SIGCOMM Computer Communication Review*, 41(2):53–56, 2011.
[52] G. Rosston, S. Savage, and D. Waldman. Household demand for broadband internet service, 2010. https://siepr.stanford.edu/research/publications/household-demand-broadband-internet-service.
[53] S. J. Savage and D. Waldman. Broadband internet access, awareness, and use: Analysis of united states household data. *Telecommunications Policy*, 29(8):615–633, 2005.
[54] N. Shetty, G. Schwartz, and J. Walrand. Internet qos and regulations. *IEEE/ACM Transactions on Networking*, 18(6):1725–1737, 2010.
[55] H. Taneja and A. X. Wu. Does the great firewall really isolate the chinese? integrating access blockage with cultural factors to explain web user behavior. *The Information Society*, 30(5):297–309, 2014.
[56] P. Vallina, V. Le Pochat, Á. Feal, M. Paraschiv, J. Gamba, T. Burke, O. Hohlfeld, J. Tapiador, and N. Vallina-Rodriguez. Mis-shapes, mistakes, misfits: An analysis of domain classification services. In *Proceedings of the ACM Internet Measurement Conference*, pages 598–618, 2020.
[57] H. R. Varian. The demand for bandwidth: evidence from the index project, 2014.

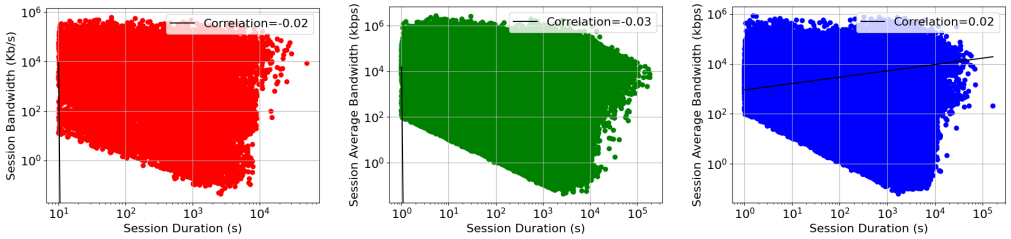
A ETHICS CONSIDERATION

Our work involves human subjects, *i.e.,* users who connect to dVPN nodes hosted by us. We followed the best community practices when conducting our work to make our data collection anonymous. Two identifiers are available for dVPN users: IP addresses and dVPN-specific identifiers. We perform coarse-grain geo-location analysis on the IP addresses which contact our nodes and then discard them. Further, dVPN-specific identifiers are not exposed to our nodes for both Sentinel and Tachyon, and in case of Mysterium we do not record them. Our code is open source [20], and we welcome investigations to verify the above statements. IRB at our institution has determined that our work is not considered human research because we used non-identifiable private information about living individuals and data collected does not contain any accompanying information by which we could identify such individuals.

B SELLERS' OPTIMAL INCOME

We now solve the seller's optimal income as defined by Equation 4 (Section 2). To solve the objective function, we need to: a) calculate the expectation $\mathbb{E}[\mathbf{n} \cdot \mathbf{B} \cdot \mathbf{U}]$ and duration $\mathbb{E}[\mathbf{n} \cdot \mathbf{U}]$, and b) quantify the impact of the blocklist on bandwidth demand. We start off by deriving assumptions based on the measurements.

First, we observe that the session duration is not correlated to the session throughput in any form as illustrated in Figure 10, where the throughput spans all values for a given session duration. Next, we notice that the session duration is not affected by implementing a bandwidth limit r as well.



(a) (b) (c)

Fig. 10. Correlation between session throughput and session duration for (a) Mysterium, (b) Sentinel, and (c) Tachyon.

The independence is verified using the Kolmogorov–Smirnov (KS) test [48]. The null hypothesis is that there is no difference between the two distributions of the session duration. Given the size of the sampled data (number of sessions), the null hypothesis is accepted when the D-statistics is below 0.021, 0.011, 0.008 given significance level $\alpha = 0.001$ for Mysterium, Sentinel, and Tachyon, respectively. As shown in Figure 11, the null hypothesis is accepted in most cases, that is, the session duration $\mathbf{U}$ is independent from the bandwidth limit r and thus the number of buyers $\mathbf{n}$.

Based on these empirical observations, we make the following assumption: the session duration $\mathbf{U}$ is independent from both the number of buyers $\mathbf{n}$ and the buyers' throughput $\mathbf{B}$, which implies

$$\begin{cases} \mathbb{E}[\mathbf{n} \cdot \mathbf{B} \cdot \mathbf{U}] = \mathbb{E}[\mathbf{n} \cdot \mathbf{B}] \cdot \mathbb{E}[\mathbf{U}] \\ \mathbb{E}[\mathbf{n} \cdot \mathbf{U}] = \mathbb{E}[\mathbf{n}] \cdot \mathbb{E}[\mathbf{U}] \end{cases} \tag{7}$$

Let $f(b)$ and $g(u)$ be the PDFs of $\mathbf{B}$ and $\mathbf{U}$, respectively. Following our assumption that the duration does not depend on other variables, the expectation of the duration is a constant:

$$\mathbb{E}[\mathbf{U}] = \int_0^\infty u g(u) du. \tag{8}$$

Given the results from Figure 4(d), the fraction of number of sessions can be expressed as a function of a node's available bandwidth, which we describe as $H(r)$ and model according to Figure 4(d). Then we have:

$$\mathbb{E}[\mathbf{n}] = H(r) \cdot N(x, y, l, A), \tag{9}$$

Further, the volume expectation is the product of the number of buyers with lower throughput than the bandwidth limit ($b \leq r$) and the mean throughput of these buyers:

$$\mathbb{E}[\mathbf{n} \cdot \mathbf{B}] = H(r) \cdot N(x, y, l, A) \int_0^r b f(b) db. \tag{10}$$

As discussed in Section 5.1, we ignore the impact of blocklists since they are not publicly available for any dVPN. This leaves us with the following objective function for seller's optimal income:

$$\begin{aligned} &\max_{x,y,r} \mathbb{E}[\mathbf{U}] \cdot H(r) \cdot N'(x, y, l) \cdot (x \int_0^r b f(b) db + y) \\ &s.t.\ \ \mathbb{E}[\mathbf{U}] \cdot H(r) \cdot N'(x, y, l) \int_0^r b f(b) db \leq D/2 \end{aligned} \tag{11}$$

Bdw Limit	1 (Mbps)	5	10	15
5	.012			
10	.011	.009		
15	.010	.014	.012	
25	.020	.008	.013	.020

(a)

Bdw Limit	1 (Mbps)	5	10	15
5	.0059			
10	.0044	.0040		
15	.0088	.0058	.0059	
25	.0081	.0076	.0046	.0077

(b)

Bdw Limit	1 (Mbps)	5	10	15
5	.0025			
10	.0040	.0022		
15	.0057	.0038	.0029	
25	.0069	.0046	.0017	.0014

(c)

Fig. 11. KS-test statistics of session duration between different bandwidth limits for (a) Mysterium, (b) Sentinel, and (c) Tachyon.

Let us fix the location and ignore the blocklist, and denote $\int_0^r bf(b)db$ by $F(r)$. Then the objective function of the sellers is

$$\begin{aligned} &\max_{x,y,r} \mathbb{E}[\mathbf{U}] \cdot H(r) \cdot N"(x,y) \cdot (x \cdot F(r) + y) \\ &s.t.\ \ \mathbb{E}[\mathbf{U}] \cdot H(r) \cdot N"(x,y) \cdot F(r) \leq D/2 \end{aligned} \tag{12}$$

The solution is as follows. Let $C(x,y,r)$ be the constraint function

$$C(x,y,r) = \mathbb{E}[\mathbf{U}] \cdot H(r) \cdot N"(x,y) \cdot F(r) - D/2. \tag{13}$$

And let

$$\begin{aligned} \mathbb{L}(x,y,r) =& \frac{1}{\mathbb{E}[\mathbf{U}]} \cdot [I(x,y,r,l,A) - \beta C(x,y,r,l,A)] \\ =& H(r) \cdot N"(x,y) \cdot (x \cdot F(r) + y) \\ &- \beta (H(r) \cdot N"(x,y) \cdot F(r) - \frac{D}{2}) \end{aligned} \tag{14}$$

We have

$$\begin{aligned} \nabla_{x,y,r,\beta} \mathbb{L} =& H(r) \cdot \{\frac{\partial N"}{\partial x}[(x-\beta)F(r) + y] + N" \cdot F(r)\}\partial x \\ &+ H(r) \cdot \{\frac{\partial N"}{\partial y}[(x-\beta)F(r) + y] + N"\}\partial y \\ &+ N" \cdot \{\frac{\partial H(r)}{\partial r}[(x-\beta)F(r) + y] \\ &+ rf(r)H(r)(x-\beta)\}\partial r \\ &+ \{H(r) \cdot N" \cdot F(r) - \frac{D}{2}\}\partial \beta \end{aligned}, \tag{15}$$

which, in accordance to the Lagrange multiplier theorem, gives the optimal condition for maximizing the objective as

$$\begin{cases} \dfrac{\partial N"}{\partial x}[(x-\beta)F(r)+y]+N"\cdot F(r)=0 \\ \dfrac{\partial N"}{\partial y}[(x-\beta)F(r)+y]+N"=0 \\ \dfrac{\partial H(r)}{\partial r}[(x-\beta)F(r)+y]+rf(r)H(r)(x-\beta)=0 \\ H(r)\cdot N"\cdot F(r)-\dfrac{D}{2}=0 \end{cases} \tag{16}$$

Received February 2022; revised March 2022; accepted April 2022