Seven Years in the Life of Hypergiants' Off-Nets

Petros Gigis
University College London

Matt Calder
Microsoft & Columbia University

Lefteris Manassakis
FORTH-ICS

George Nomikos
FORTH-ICS & Lancaster University

Vasileios Kotronis
FORTH-ICS

Xenofontas Dimitropoulos
FORTH-ICS & University of Crete

Ethan Katz-Bassett
Columbia University

Georgios Smaragdakis
TU Delft

ABSTRACT

Content Hypergiants deliver the vast majority of Internet traffic to end users. In recent years, some have invested heavily in deploying services and servers *inside* end-user networks. With several dozen Hypergiants and thousands of servers deployed inside networks, these *off-net* (meaning outside the Hypergiant networks) deployments change the structure of the Internet. Previous efforts to study them have relied on proprietary data or specialized per-Hypergiant measurement techniques that neither scale nor generalize, providing a limited view of content delivery on today's Internet.

In this paper, we develop a generic and easy to implement methodology to measure the expansion of Hypergiants' off-nets. Our key observation is that Hypergiants increasingly encrypt their traffic to protect their customers' privacy. Thus, we can analyze publicly available Internet-wide scans of port 443 and retrieve TLS certificates to discover which IP addresses host Hypergiant certificates in order to infer the networks hosting off-nets for the corresponding Hypergiants. Our results show that the number of networks hosting Hypergiant off-nets has tripled from 2013 to 2021, reaching 4.5k networks. The largest Hypergiants dominate these deployments, with almost all of these networks hosting an off-net for at least one – and increasingly two or more – of Google, Netflix, Facebook, or Akamai. These four Hypergiants have off-nets within networks that provide access to a significant fraction of end user population.

CCS CONCEPTS

• **Networks** → **Network measurement**.

KEYWORDS

Hypergiants, Content Delivery Networks, TLS, Server Deployment.

ACM Reference Format:
Petros Gigis, Matt Calder, Lefteris Manassakis, George Nomikos, Vasileios Kotronis, Xenofontas Dimitropoulos, Ethan Katz-Bassett, and Georgios Smaragdakis. 2021. Seven Years in the Life of Hypergiants' Off-Nets. In *ACM SIGCOMM 2021 Conference (SIGCOMM '21), August 23–27, 2021, Virtual Event, USA.* ACM, New York, NY, USA, 18 pages. https://doi.org/10.1145/3452296.3472928

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
SIGCOMM '21, August 23–27, 2021, Virtual Event, USA
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8383-7/21/08.
https://doi.org/10.1145/3452296.3472928

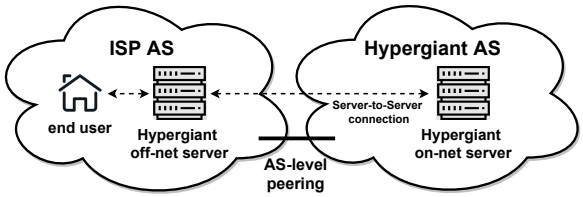


Figure 1: Hypergiant's off-net footprint (in an ISP) and on-net footprint (in the Hypergiant's AS).

1 INTRODUCTION

The vast majority of traffic to Internet users comes from a small number of content providers, cloud providers, and content delivery networks that are heavy traffic outbound, including Google, Netflix, Facebook, and Akamai. These providers, dubbed Hypergiants (HGs) by Labovitz et. al [63], deliver content to billions of users around the world. In 2019, more than half of Internet traffic originated from only 5 HGs [31, 84, 103], a significant consolidation of traffic since 2009, when it took the largest 150 ASes to contribute half the traffic, and 2007, when it took thousands of ASes [63].

To deliver high quality user experience in the face of ever increasing demand for content, HGs invest heavily in their infrastructure. They construct datacenters [53, 95] and roll out fiber to build their backbone [10, 16, 61]. They peer at colocation facilities and Internet Exchange Points (IXPs) worldwide [47, 89, 112]. They also peer directly with eyeball networks, bypassing transit providers to improve user performance and cut costs [10, 11, 27, 39, 63, 68, 80]. For example, Google peers with more than 7.5k networks [11] (ca. 2020), by establishing peerings at more than 100 colocation facilities and 150 IXPs [49].

HG's Off-net Footprint. HGs operate their own networks and datacenters, with servers assigned IP addresses from their own ASes. Some HGs also install servers inside eyeball or other networks, to serve users in those networks or their customers [52, 68, 78, 80]. These servers are assigned IP addresses of the hosting network. Since 2000, Akamai has deployed their servers in hundreds of networks around the globe [68, 80]. We refer to these servers as *off-nets* because they are outside (off) the HG's own network, in contrast to the *on-nets* that a HG hosts on its own network (see Fig. 1). More recently, other HGs have followed this model, e.g., Google launched

the Google Global Cache [52], Netflix has Open Connect [78], and Facebook [88] and Alibaba [7] operate their own CDNs.

Despite the dominant role that HGs play in delivering Internet content, the research community has lacked general and scalable methods to track their growth and impact on the Internet topology.

Why measure Hypergiant Off-nets? Being able to track the expansion of the Hypergiants inside other networks, especially eyeball networks, has implications on the modeling of Internet structure and Internet traffic flow. By deploying off-nets, HG's content is localized within the hosting network, with less traffic crossing network boundaries. This view challenges the research community's mental model of the value of peering and exchanged traffic between networks and can impact the scope and considerations of net neutrality regulation. It has also implications on network performance, as crossing network boundaries comes at a cost [27, 43, 68]. Moreover, operating servers within a network improves the strategic position of Hypergiants as they are able to control both the origin server within their network as well as the servers within other networks and, thus, optimize content delivery to end users. Looking forward, it is important to understand if major Hypergiants, responsible for 90% of the traffic consumed by end-users, have already taken steps towards delivering services in emerging networks (*e.g.,* 5G) that require close proximity to mobile users and performance guarantees. The research community lacks a good understanding of the global expansion strategies of major Hypergiants and how much of the Internet population can be served locally. Such insights can inform studies in other fields, including economics, political science, and regulation. Section 8 revisits some of these topics in light of our results uncovering the off-net footprints of major Hypergiants.

Challenges and Previous Work. Off-net servers of large HGs (e.g., Google, Netflix, Facebook, Akamai, and many others) typically use IP addresses announced by the hosting network (by ISPs rather than by the HG), making it impossible to identify the server as belonging to the HG using traditional techniques such as inspecting BGP feeds. Alternate approaches have been used, but they either require access to distributed vantage points and so have limited coverage, or they are tailored to a particular HG, so lack generality and are fragile to changes that the HG might make.

The first category of earlier approaches relies on issuing DNS queries from many locations since HGs can direct users to a particular server by resolving DNS queries to the server's IP addresses. These approaches either use a distributed measurement platform [87, 101], open recursive resolvers that (generally mistakenly) will respond to queries from arbitrary hosts [54, 104], or crowd sourced requests [3, 73]. Approaches to study YouTube's infrastructure have used 5 vantage points in different networks [102] and a combination of open DNS resolvers and PlanetLab [2]. However, none of these techniques has resulted in truly global coverage, which becomes more of a problem as HGs expand their footprints and use anycast [22], and studies that (ab)use open resolvers also raise ethical concerns.

The second category of earlier approaches get around the need for distributed vantage points via DNS-based techniques tailored to individual HGs. Studies have emulated issuing DNS queries from around the world by using the DNS Extension Client-Subnet (ECS), which allows a DNS query to include the client's IP prefix, allowing researchers to issue queries to a HG that appear to come from arbitrary locations/prefixes [21, 100]. However, many HGs do not support ECS, and even ones that do may reply only to ECS queries from whitelisted resolvers [25]. Further, even Google, the subject of earlier ECS-based mapping, now only responds to DNS queries for domains such as `www.google.com` with IP addresses of on-net servers, and so ECS-based mapping efforts no longer uncover Google off-nets. Other studies have mapped Facebook [13–15] and Netflix [17] off-nets by exploiting patterns in the naming scheme for off-net DNS records, then exhaustively trying queries based on those patterns. This approach is fragile and tedious, as hostname patterns may change, requires tailored patterns per HG so can be difficult to scale, and is not general, as some HGs do not have exploitable naming conventions.

Our Approach. We present the first approach for identifying off-nets that is both general, working across Hypergiants, and complete, achieving global coverage of their off-net footprints. Additionally, our approach is amenable to using existing public datasets, enabling us to apply it in a longitudinal study uncovering the growth in off-net deployments.

Our approach relies on two key observations. First, a Hypergiant's off-net servers host the Hypergiant's Transport Layer Security (TLS) certificate(s) and must provide the certificate(s) in response to queries. Recent years have seen a dramatic increase in use of Transport Layer Security (TLS), such that the majority of Internet traffic today is encrypted [31]. Adoption of encryption has been particularly high among HGs, with the percentage of Google traffic that is encrypted increasing from from 50% in 2014 to 95% in 2020 [50].

A TLS certificate validates the identity of a service run in a server, and so a server possessing a Hypergiant's certificate indicates it is a server for the Hypergiant. If a server outside the Hypergiant's network has the certificate, it is an off-net for the Hypergiant, an observation we validate later in the paper. Because TLS certificates and message exchanges are standardized, TLS scans of the IP address space provide an approach to identify off-nets that will work for any Hypergiant (that uses TLS) and can cover all publicly-addressable servers.

Second, because the approach relies on standard TLS scans, we can use existing certificate corpuses to perform historical and longitudinal analysis. Such corpuses are readily available for commercial and research use, *e.g.,* from Rapid7 [86] and Censys [35].

In combination, the wide adoption of TLS and the available certificate datasets provide an opportunity to infer the off-net footprints of all HGs and to greatly enhance the community's understanding of Internet content delivery.

Our Contributions:

- We develop a generic methodology to infer all HGs' off-net footprints by analyzing corpuses of scanned certificates. We augment our methodology with analysis of HTTP(S) header corpuses to differentiate a HG service hosted on third party platforms (*e.g.,* Netflix web servers running in AWS) from a HG service running on its own servers (*e.g.,* Netflix Open Connect video caches).
- By applying our methodology on certificate corpuses that span more than seven years (2013-2021), we find that the number of ASes that host HG off-net installations has tripled, reaching 4.5k

in April 2021. We validate our results by surveying HG operators. Those that replied indicated that we correctly uncovered 89-95% of ASes hosting their off-nets.
- Of networks that host any HG off-nets, our analysis reveals that the large majority host at least one of the four largest HGs (Google, Netflix, Facebook, and Akamai, the four largest in terms of number of networks in which they have off-nets).
- ASes that already host at least one HG tend to host more over time. Most off-nets are in small and medium ASes, which is not surprising since most ASes in general are small, but a disproportionate share of large ASes host them as well.
- Hypergiants have rapidly expanded their off-net footprints in Europe, Asia, and Latin America, the last seeing exponential growth.
- As a result of this expansion, a significant fraction of the end user population can be potentially served by the off-net deployments of Google, Netflix, Facebook, and Akamai.
- Our analysis unveils different strategies by HGs. While a number of Hypergiants have expanded significantly, we also observe shrinking of deployments, most notably in the case of Akamai.

Artifacts. To support future research, we make our software and results publicly accessible to the research community through our project website [44]:
https://github.com/pgigis/sigcomm2021-hypergiants-offnets

What this paper is not about: As we do not know in detail the business strategies, deployment planning, peering arrangements, and the performance and cost goals of individual HGs, our study is not a head-to-head comparison of different HGs. It is possible that a HG with a smaller off-net footprint can still serve more users or provide better performance. Performance evaluation of different HG off-net footprints is out of the scope of this work.

2 BACKGROUND

In HTTPS, HTTP traffic is encrypted using the cryptographic protocol Transport Layer Security (TLS), which secures communication using public certificates that are exchanged and verified. The standard format of public key certificates is X.509 [32]. The certificates contain several fields [32, 90]. As we will explain in more detail in Section 4, we use the following fields and properties to establish which organization and site(s) a certificate belongs to and whether the certificate is valid:

Subject Name. This field identifies the entity associated with the certificate via a number of (sub)fields. This paper uses the Organization entry, naming the organization associated with the certificate (*e.g.,* Google LLC).

dNSName. The DNS name dNSName extension lists the domains that this certificate certifies (*e.g.,* *.google.com, *.google.com.br, *.googlevideo.com, ...).

Server Name Indication (SNI). SNI is a TLS protocol extension that allows a server to serve multiple certificates for different hostnames, all under a single IP address [34]. During the TLS handshake phase, the client provides the hostname that it wants to reach, and the server replies back with the corresponding certificate. If a client does not include this extension, the server replies with its default certificate.

Validity Period. This uses the NotBefore and NotAfter fields to define the time window within which a certificate should be considered as valid. The values depend on the policy of the owner (*e.g.,* Netflix used short-lived ones [83]).

Certificate Authority. It indicates whether the certificate is a Certificate Authority (CA) or an end entity one.

Certificate chains and verification. To reflect the hierarchical chain(s) of trust from CAs down to certificate-owning organizations, certificates are typically organized in chained lists. A certificate chain is essentially an ordered list of certificates, containing a TLS Certificate and CA Certificates, that enable the receiver to verify that both the sender and all involved CAs are trustworthy. The chain begins with the end entity (EE) certificate, and each certificate in the chain is signed by the entity identified by the next certificate in the chain. Any certificate that sits between the EE Certificate and the Root Certificate is called an Intermediate Certificate. The first Intermediate Certificate is the signer/issuer of the EE Certificate. The Root CA Certificate is the signer/issuer of the penultimate Intermediate Certificate and is a CA-signed certificate (typically pre-installed client-side) that terminates the chain. The signatures of all certificates in the chain must be verified up to the Root CA Certificate.

3 CHALLENGES

At first glance, it may seem that scanning for TLS certificates immediately solves the problem of locating all off-nets – if an IP address outside of a Hypergiant has "the" Hypergiant's certificate, it is an off-net server for that Hypergiant; if it does not, it is not. However, a number of challenges arise, mainly due to the complex and heterogeneous deployment strategies of different Hypergiants:

It is not trivial to determine which certificates to look for, as there is not necessarily one certificate that definitively identifies each Hypergiant. In fact, different Hypergiants deploy very different certificate management strategies (see Appendix A.3). Further, serving infrastructure can reflect relics of business history. For example, LinkedIn and Github have been acquired by Microsoft but might use different serving infrastructure, either their own or third-party. We want our technique to be general enough to accommodate these strategies without requiring significant per-Hypergiant tuning or compromising coverage (uncovering the HG off-net footprint) or accuracy (confidence on the ownership of the server).

The presence of a Hypergiant certificate on a server outside that Hypergiant does not guarantee the server is an off-net content server of the Hypergiant. A number of deployment models can lead to other servers having the certificate:

- Some Hypergiants use their own infrastructure for some services and third-party CDNs for others (*e.g.,* Twitter images come from Akamai and Verizon, but some other content comes from their own infrastructure, and Netflix uses Amazon for web frontends but its own CDN for video). Some Hypergiants (*e.g.,* Apple and Microsoft [94]) have their own infrastructure but also use third-party CDN servers for resilience, capacity, and/or to extend their footprint. These servers may have certificates from a Hypergiant (possibly in addition to certificates from the CDN) and provide the Hypergiant's services, even though they are not part

of the Hypergiant's off-net footprint in terms of the underlying hardware.
- A certificate for a Hypergiant may exist on a server that is not serving infrastructure for the Hypergiant. Cloud providers offer on-premise versions of products such as AWS Outposts, Azure Stack, and Google GKE, which are managed by the cloud provider but do not host public services for it. However, these servers may host a certificate for the cloud provider on a management interface. Similarly, HG certificates may exist on servers used for aspects of their business other than content serving, such as payroll.
- Some Hypergiants like CloudFlare issue TLS certificates to customers of their proxy services, and so a customer server offering a CloudFlare-issued certificate could be mistaken for a CloudFlare off-net.

A simple scan of the non-Hypergiant IP address space may not uncover all off-nets. For Hypergiants serving content over anycast [22], the user-facing IP address for on-net and off-net servers is the same, complicating differentiating one from another, and queries to that interface will reach a particular anycast instance based on the source of the query. Therefore, simply scanning the IP address space from one or a few locations is not enough to uncover every instance of the anycast IP address [29], potentially leaving some of the HG footprint uncovered.

4 METHODOLOGY

We develop a methodology that uses TLS certificate scans as a building block, supplementing them with techniques we develop to address the challenges mentioned in Section 3. First, we learn a Hypergiant's TLS fingerprints by scanning its on-nets (§4.2). Second, we search for the TLS fingerprint in scans of off-net IP addresses to identify candidates (§4.3). Third, we learn the Hypergiant's HTTP(S) header fingerprint by again scanning on-nets (§4.4). Fourth, we confirm the off-net candidates by scanning them for the HTTP(S) header fingerprints (§4.5). Our approaches address most of the challenges, but we discuss their remaining limitations in Section 7.

4.1 Validating Certificates

Throughout, we only use valid certificates. As recommended in prior studies [23, 28], we verify the intermediate/root certificates of each certificate chain against a list of well-trusted root and intermediate certificates which form the WebPKI (extracted from the Common CA Database [76]). We discard any certificates that (at the time they were gathered) were expired, based on the `NotAfter` and `NotBefore` fields. We also discard all self-signed end-entity certificates as they can be issued by anyone to mimic valid HG certificates. During the period of our study, more than one third of the hosts returned invalid certificates that we excluded.

4.2 Learning Hypergiant TLS Fingerprints

A Hypergiant may not have a single defining TLS certificate, for example if it operates different services with different certificates, and so we first learn the fingerprints that identify a particular Hypergiant, in order to later apply the fingerprints to Internet-wide scans. The input to this step is the name of a Hypergiant (e.g., "google") and TLS scans of all IP addresses announced by that Hypergiant (Section 4.6 provides details on the scans we use in this paper). The intuition is that servers in this IP space with end entity (EE) certificates matching the Hypergiant name are extremely likely to be on-net servers of the Hypergiant and so provide a reliable fingerprint for the Hypergiant's serving infrastructure. We are interested in the EE certificates, as they include information for the server owner, while intermediate/root certificates can contain third-party organization information.

From the EE certificates found in the Hypergiant's address space, we identify Hypergiant's on-net servers by performing a case insensitive search of the Hypergiant's name in the TLS `Organization` field of the `Subject Name`, as organizations tend to use their primary organization name to prove the identity and validity of their certificates [33]. Any organization can potentially obtain a Domain-Validated (DV) [1] certificate with, e.g., "google" in the `Organization` field of the `Subject Name`, as the field is not validated or authoritative, and so the `Organization` on its own is not a reliable fingerprint. To supplement it, we extract the list of DNS names (the TLS `dNSName` field, which is authenticated) from the end-entity certificates of the on-net servers, creating a set of DNS names served by the Hypergiant.

4.3 Using Fingerprints to Identify Candidate Off-nets

We then use the fingerprint – specifically the set of DNS names – to search for the presence of certificates from the Hypergiant on IP addresses outside the Hypergiant, as these are its candidate off-nets. We again search for the name of a Hypergiant, e.g., "google", in the TLS `Organization` field of the `Subject Name`. For each matching certificate, we check whether all of the DNS names in the certificate's `dNSNames` field are in the Hypergiant's set of DNS names from on-net certificates we found in the previous step. If they are, the IP address providing the TLS certificate is a candidate off-net. By requiring that all DNS names in the certificate be present in on-net certificates, we filter out cases where the HG is a certificate provider (e.g., Cloudflare) and also cases where the Hypergiant shares a certificate with another organization.

4.4 Learning Hypergiant HTTP(S) Fingerprints

We identify fingerprints in Hypergiant HTTP(S) headers as a basis for excluding off-net candidates that have a certificate from the Hypergiant but are not in fact among its off-net servers (§3). Large content providers and CDNs often use HTTP response headers for debugging purposes, and we inspect these headers to create a per-Hypergiant fingerprint, using responses from on-net servers in Rapid7 HTTP and HTTPS scans from September 2020. We filtered out common standard headers (*e.g.,* `Cache-Control` and `Content-Length`). Since the servers of a particular Hypergiant are likely to share debugging headers, we identified the 50 most frequent header name-value pairs and the most frequent header names for each Hypergiant's on-net servers.

Next, we performed manual classification and validation to find header fingerprints that identify the Hypergiant's web servers. There is a small number of Hypergiants, so we found that examination on a per-case basis was suitable for our work. We leave

Hypergiant	Header Name	Header Value	Documented
Akamai	Server	AkamaiGHost	Yes [5]
Cloudflare	CF-Request-Id		Yes [30]
Google	Server	gws*	Disclosed [48, 58]
Facebook	X-FB-Debug		Yes [38]

Table 1: Examples of headers used to identify HG servers. Empty header values indicate that only the header name is used to match. Entries ending with * indicate a prefix match.

automation of this step for future work. For most frequently occurring headers, HG-specific headers were easily identifiable either from a unique header name or value containing an abbreviated name of the Hypergiant. For nearly 80% of cases, we found public documentation or disclosure confirming the use of these headers by HGs. Table 1 shows several examples, and Appendix A.5 provides the full list. We also verified the presence of the headers with independent tests on content (*e.g.,* `google.com`) for each HG. An interesting case is Netflix, as we discovered that a fraction of its servers responded with the default ngnix header. For our analysis, we consider a server with a Netflix certificate and the default ngnix HTTP(S) header as a Netflix off-net.

4.5 Confirming Candidates Using HTTP(S)

We apply these HTTP(S) header fingerprints to the off-net candidates from Section 4.3 and classify as off-nets any that match the Hypergiant's fingerprint. To assign an IP to an AS we use standard IP-to-AS mapping techniques described in Appendix A.1. We also annotate HGs' on-nets and off-nets as described in Appendix A.2.

4.6 Datasets

Certificate datasets. Rapid7 collects `X.509` certificates observed in IPv4-wide scans on port 443 [66]. We use datasets from once every three months from Oct. 2013 to Apr. 2021, which include 127,812,006 unique certificates. We supplement with port 443 scans from Censys [35] from Nov. 2019 to Apr. 2021. We exclude certificates that we cannot translate to the `X.509` format and those missing critical information.

HTTP(S) headers. We use the corpus of available HTTP(S) headers from Rapid7 from Oct. 2013 to Apr. 2021.

List of hypergiants. We compile a list of HGs using previously published surveys [17, 18, 31, 63, 111], then select the 23 that claim on their website to have a CDN and for which we were able to identify a certificate with a matching `Organization`. Examined HGs: Akamai, Alibaba, Amazon, Apple, Bamtech, Highwinds, CDN77, Cachefly, Cdnetworks, Chinacache, Cloudflare, Disney, Facebook, Fastly, Google, Hulu, Incapsula, Limelight, Microsoft, Netflix, Twitter, Verizon and Yahoo.

5 VALIDATION

We find that our scans and measurement techniques are accurate, then validate our results against information from Hypergiants and results from earlier approaches, finding that our results are trustworthy.

Comparison of Scanning Corpuses. Most results in our paper rely on Rapid7 scans, and so we first evaluate its completeness compared to Censys and an active scan we conducted (Nov. 21-25, 2019) of the entire publicly-routable non-bogon IPv4 address space for `SSL/TLS` certificates on port 443. We use a modified version of the `certigo` tool [96] to perform `TLS` handshakes with servers to fetch their certificates. Our scan fetches 13,156,080 unique end-entity certificates.

Table 2 compares our scan and Rapid7 and Censys scans from November 2019. The number of IP addresses with certificates in Rapid7 and Censys is very similar. However, our `certigo` scan found around 20% more addresses, which we attribute to two causes. First, both Rapid7 and Censys have to respond to complaints and remove IP addresses from their scans [12, 28, 109]. As both scans have run for years, more address space is excluded over time. A second reason for this difference is that our scan took almost four days to execute, which may trigger less rate limiting than the other, faster scans.

However, when we turn our attention to the total number of ASes that host at least one HG (column 6), the numbers across all three datasets are very similar, as they are for the four Hypergiants with the largest footprints (Google, Netflix, Facebook, and Akamai) reported in the last four columns. Another observation is that the number of IP addresses per HG is not relevant to the size or the distribution of the corresponding HGs' off-nets, as each HG has a different strategy on how to assigns IP to servers [41]). We have confirmed that some HGs have only a few front-end IP addresses for multiple servers, and others have multiple IP addresses for one server. For instance, in our active scan (Nov. 2019) we collected Facebook certificates from 33,769 IP addresses in 1,708 off-net ASes. At the same campaign, we collected Akamai certificates from many more IP addresses (105,686) although Akamai's off-net footprint is smaller, with 1,194 off-net ASes. Thus, for the rest of the paper, we will focus on the off-net AS footprint of each Hypergiant.

Ethical Considerations. In our scan, we remain "good Internet citizens" by following best practices [36] to avoid triggering any kind of alarm. We maintain a blacklist and use clients with appropriate rDNS names, websites, and abuse contacts. Therefore, this work does not raise any ethical issues.

Active Measurement Validation. We provide additional validation of our inferences using active measurements. To accomplish this we use the ZGrab2 [114] tool which provides rapid capture of HTTP(S) banners including HTTP headers and TLS certificate validation. We provide an input list of (IP address, domain) pairs and ZGrab2 correctly sets the HTTP Host header and TLS SNI field while performing a GET request for the default document.

In this analysis, we assert that, if we correctly infer a server to be an off-net for a particular Hypergiant, it should not serve requests for domains which the Hypergiant does not host (i.e. TLS validation should fail). To test this, for each IP address that we inferred as an off-net for a particular Hypergiant, we randomly select 10 other Hypergiants and, for each, scan the IP address requesting one of the 50 most popular domains served by the other Hypergiant. Surprisingly, we found that only 89.7% of the inferred off-nets did not validate for the randomly selected domains. Of the 10.3% that correctly validated requests, we inferred 97% as belonging to Akamai, and the request domains (LinkedIn, KDDI, Disney) were for ones known to be also served by third-party CDNs such as Akamai. This result highlights a challenge in understanding the content delivery

Scan (abbreviation)	Date	#IPv4 IPs w/ certs	#ASes w cert	#ASes unique	# ASes w/ Hypergiant Certificates any	Google	Netflix	Facebook	Akamai
Rapid7 (R7)	Nov. 18, 2019	35,009,714	57,769	84	3,788	3,137	1,760	1,737	1,235
Censys (CS)	Nov. 19, 2019	34,235,590	58,183	211	3,974	3,355	1,689	1,746	1,248
Certigo (AC)	Nov. 21-25, 2019	41,357,388	59,178	519	3,802	3,149	1,715	1,762	1,236

Table 2: Statistics for the three scan corpuses of certificates in our study in November 2019.

ecosystem: Large content providers may select a combination of self-hosted and external CDNs for redundancy and additional capacity (§3). Nevertheless, Akamai is the exception in our analysis, as it does not create its own content or have its own users, and its platform delivers the content of other companies, very much like a "cloud provider" for delivery of content.

In this analysis, we assert that servers outside of HG address space should not serve HG domains unless we inferred them to be off-nets. Using the November 2020 Rapid7 and Censys datasets, we selected a random 25% sample out of 57 million IP addresses with responsive web servers that we did not infer to be Hypergiant on-nets. Then for each selected IP address, we select 10 random HG domains as described in the previous analysis. From our random sample, we found 0.1% (17,029) IP addresses in 844 ASes with valid TLS responses. Of those, 98% were servers we had correctly inferred as HG off-nets. We believe the remaining 2% are customer origins of CDN-hosted sites.

Validation from Hypergiants. We surveyed HG operators following a similar approach to earlier work on mapping ISPs [97]. The survey questions are presented in Appendix A.4. Four HGs replied to our request for validation, including some of the four largest. All four agreed that the estimation of the off-net footprint is very good. One HG operator indicated that 6% of ASes we identified as hosting the HG's off-nets were not on the HG's list, and 11% from the HG's list were not uncovered by our technique (while also indicating that the HG's list may not be 100% correct). For the other three HGs, we underestimated the HG footprint by 5% (one HG) or around 10% (two HGs). Our technique may miss or misidentify ASes hosting HG off-nets because of errors in IP-to-AS mapping, because of ASes that have opted out of TLS scans (*e.g.,* Table 2 shows that different scans reveal different coverage), or because of churn between when we measure and when validation was done. The off-nets we missed were in a mix of different types of networks. Our results are in-line with Akamai's public reports for the duration of our study [4].

Comparison to Earlier Results. Our technique works over existing datasets, which enables the comparison of our results with prior studies using different methodologies.

Google: We compare our results with the latest results from a previously published approach to uncovering Google off-nets [21], which reported 1445 ASes hosting Google off-nets in April 2016. Of the 1445 ASes, our approach identified 1421 (98%), while also identifying an additional 283 (68 of which the earlier technique identified prior to April 2016).

Facebook: To the best of our knowledge, the only previous work which reports numbers for Facebook's CDN belongs to a team that participated in a hackathon [13] in March 2018 and posted updated results in August 2018, in November 2019 [14], and in April 2021 [15]. The team mapped Facebook servers globally by guessing DNS names of off-nets based on Facebook naming conventions and global airport codes. Our technique uncovered 1153 of the 1201 ASes (96%) in the team's 2018 data, 1599 of the 1704 ASes (94%) in the 2019 data, and 2068 of the 2187 ASes (95%) in the 2021 data. We have applied the same IP-to-AS mapping in both datasets.

Netflix: A previous study [17] reported that on May 15 2017, 743 ASes hosted Netflix Open Connect servers. In April 2017, we report 769 ASes.

	Hyper-Giant Name	Number of ASes with HG off-nets 2013/10 (only certs)	Max [Snapshot]	2021/04 (only certs)
1.	Google	1044 (1105)	3810 [2021/04]	3810 (3835)
2.	Facebook	0 (8)	2214 [2021/04]	2214 (2229)
3.	Netflix	47 (143)	2115 [2021/04]	2115 (2288)
4.	Akamai	978 (1013)	1463 [2018/04]	1094 (1107)
5.	Alibaba	0 (0)	184 [2018/01]	136 (301)
6.	Cloudflare	0 (2)	110* [2021/01]	110* (137)
7.	Amazon	0 (147)	112 [2017/07]	62 (218)
8.	Cdnetworks	0 (4)	51 [2019/01]	11 (31)
9.	Limelight	0 (1)	42 [2020/04]	32 (32)
10.	Apple	0 (113)	6 [2020/04]	0 (267)
11.	Twitter	0 (101)	4 [2021/04]	4 (180)

Table 3: List of the examined HGs according to the Rapid7 dataset (Oct. 2013-Apr. 2021), sorted by the max # ASes hosting the HG's off-nets (validated by both certificates and headers). (* See the last paragraph of Section 6.1 for a discussion of Cloudflare.)

6 HGS' OFF-NET FOOTPRINTS & GROWTH

This section discusses the footprint and growth of HGs' off-net deployments. We characterize their growth by type of network, region, and coverage of Internet users. We also comment on the network providers' hosting strategies for HGs.

6.1 Hypergiant Statistics

We first consider the Rapid7 certificate dataset. In Figure 2 we show the number of IP addresses with a certificate during the period of our study, from Oct. 2013 to Apr. 2021. We report the number directly from the raw Rapid7 data, i.e., before validating the certificate. Then, we apply our methodology from §4.1-§4.3 to infer the IP addresses that use certificates associated with any of the HGs we study. In Figure 2 (refer to the right y-axis) we plot the percentage of IP addresses we infer serving any of the HGs, either hosted in each HG (dashed line) or in a non-HG AS (dotted line). At the start of 2021, only around 3.8% of the IP addresses seen with valid certificates in Rapid7 are associated with any of the HGs we study, either hosted in one of HGs' ASes or in a non-HG AS and, thus, can potentially be HG off-nets.

Table 3 reports the number of ASes that host each HG off-net footprint at the beginning of our study period (Oct. 2013) and at the end of the study period (Apr. 2021) after validation with headers (§4.5).

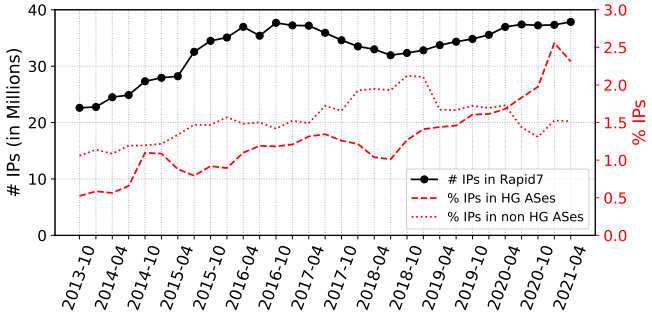


Figure 2: # IP addresses (millions) hosting TLS certificates in raw Rapid7 dataset files (*left y-axis*). Of those, % that host a HG certificate, broken down by whether the IP address belongs to a HG AS or not (*right y-axis*).

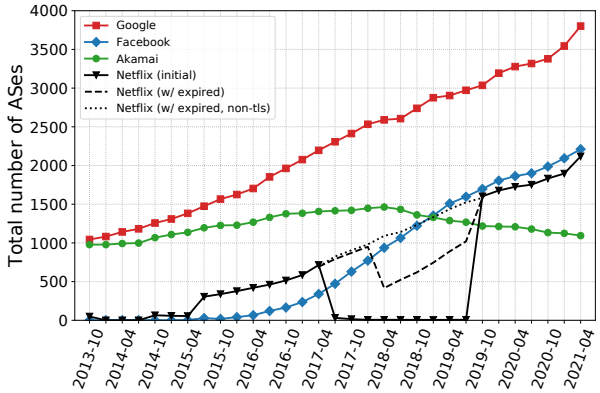


Figure 3: Off-net footprint growth for top-4 HGs over time.

The middle columns of the table gives the maximum number of ASes observed hosting off-nets for a HG and the timestamp when that maximum deployment occurred. The table ranks HGs based on their maximum AS footprint. For half of the HGs (Microsoft, Hulu, Disney, Yahoo!, Chinacache, Fastly, Cachefly, Incapsula, CDN77, Bamtech, and Highwinds), our methodology inferred no off-net footprint during the period of our study, and so the table excludes them.

The deployment strategies of HGs differ. In Section 5, we explained that the absolute number of IP addresses is a good comparison metric. The percentage of IP addresses with certificates (of the top-4 HGs, see Table 3) that were hosted in non-HGs ASes is very high for some HGs, *e.g.,* Google, Netflix, Facebook, and very low for others *e.g.,* Amazon. However, the total number of IP addresses is only a small percent of the dataset of IP addresses with certificates. We also notice that there are two distinct HG groups. First, there are the four largest HGs, namely Google, Netflix, Facebook, and Akamai, which have off-nets in more than 1,000 ASes at the beginning of 2021. For some of HGs, e.g., Google and Akamai, the size of the off-net footprint with server installations (after the validation with headers) is very close to the size of the service-present off-net footprint (as inferred by the certificates alone; see values in parentheses). In some other HGs, e.g., Alibaba, this is not the case, as they run services by relying on servers operated by other HGs or datacenter providers. CloudFlare poses an interesting case, since our manual investigation reveals that it does not have an off-net footprint, but, because it issues and installs certificates in clients that operate in other networks (and to support its DNS service `1.1.1.1`), Cloudflare is misidentified as having off-nets.

6.2 Longitudinal Growth

Figure 3 plots the off-net footprint growth (after the validation with HTTP(S) headers) of the top-4 HGs (Google, Netflix, Facebook, and Akamai) based on our analysis of the Rapid7 certificate dataset. The off-net footprint of these HGs is growing substantially, with the exception of Akamai. Facebook has shown rapid growth since it launched its own CDN in the summer of 2016 [60].

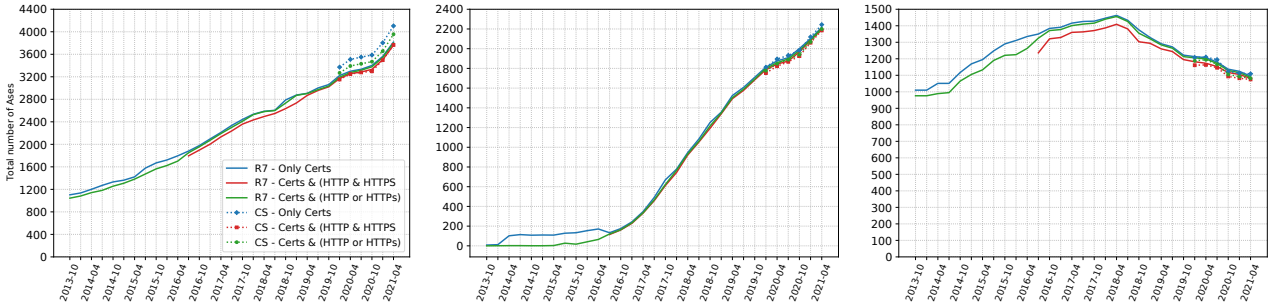
The case of Netflix is the most complex one, requiring manual investigation. Although Netflix's off-net footprint grew constantly after 2015 (we conjecture this may be due to peering disputes with ISPs [71] and the strategic decision to launch Open Connect), we observed a significant fraction of IP addresses responding with an expired certificate after April 2017. This is visible in Figure 3 (solid). When we ignore the expiration date of this certificate, we can restore the activity of Netflix as shown by the Netflix dashed line. In Oct. 2019 the default certificate of these IP addresses changed back to a valid one. We also validate this by using PTR records which at that time contained information about Netflix's footprint. We also observed that 26.8% of the IP addresses serving a certificate for Netflix before April 2017 and after July 2019 stopped responding to `HTTPS` requests on port 443.

To further investigate, we downloaded and studied the responses to `HTTPS GET` (port 443) and `HTTP GET` (port 80) requests from Rapid7. We found that these IP addresses were in fact active during this period, but on `HTTP` instead of `HTTPS`. We conjecture that Netflix moved from `HTTPS` to `HTTP` to cope with high demand as encryption requires additional resources, a challenge Netflix has admitted [98]. However, the Netflix `SNI` policy might also have changed. By restoring these IP addresses as well, we plot (dotted line) the number of ASes that hosted Netflix off-nets between Oct. 2017–Nov. 2019. For the rest of the paper, we will use the envelope of these two lines (solid, dotted) to refer to the ASes that form the Netflix off-net footprint.

We want to study the impact of using only the certificate datasets in the number of ASes we identify as off-nets, compared to using certificates plus HTTP and/or HTTPS headers, as discussed in Section 4. As shown in Figure 4, the differences are minimal, as all straight and dotted lines seem to converge. Rapid7 offers HTTPS data since July 2016, and we have Censys data since October 2019. However, in the case of Google, using the Censys dataset we are able to identify more ASes, possibly because Censys employs sophisticated scanning techniques. This fact is also apparent in Table 2.

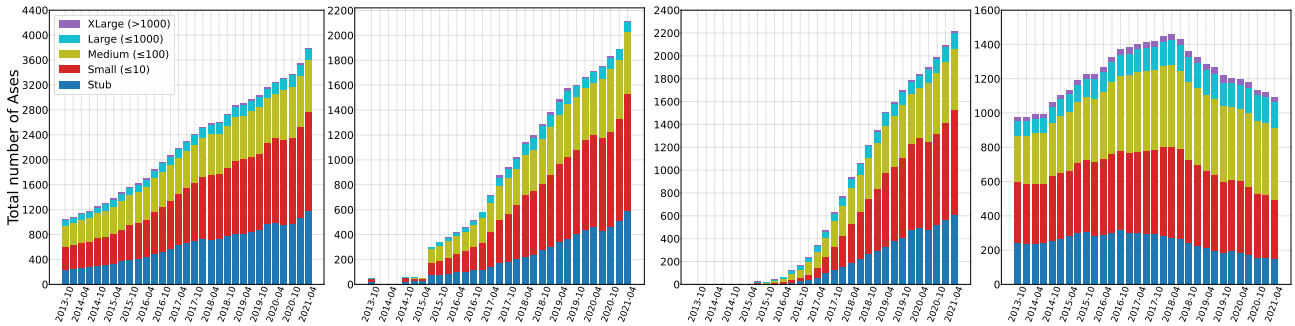
6.3 Growth by Network Type

We are also interested in understanding the "demographics" of ASes that host HGs. To this end, we label ASes that host HGs following a similar approach to the one from previous related work [21], i.e., by characterizing the ASes based on their AS customer cone size. To do this, we employed the CAIDA AS Relationships Dataset, specifically the provider-peer customer cone inferred for each AS [20]. We obtain more than 7 years of monthly snapshots of the dataset



(a) Google (b) Facebook (c) Akamai

Figure 4: Comparison of Rapid7 (R7) and Censys (CS) datasets (certification and headers) for Google, Facebook, and Akamai, across time. Recall, the HTTPS headers are available only after Summer 2016. (note: y-axis scales differ)



(a) Google (b) Netflix (c) Facebook (d) Akamai

Figure 5: Growth of top-4 HGs' off-net footprints grouped by AS customer cone size. (note: y-axis scales differ)

matching our study timeline. We consider 5 categories of ASes, separated by an order of magnitude in terms of their customer cone size. Stub ASes have no customer cone (other than themselves), Small ASes have customer cones ≤ 10 ASes, Medium ASes have customer cones ≤ 100 ASes, Large ASes ≤ 1000 ASes, and XLarge ASes > 1000 ASes.

In Figure 5 we plot the top-4 HGs' off-net footprint in the form of stacked bars. Each bar refers to one of the AS categories: Stub, Small, Medium, Large, and XLarge. Smaller ASes, namely, Stub, Small, and Medium ASes contribute most (between 93-96%) to the growth of Google, Netflix, and Facebook. For Akamai's off-net footprint, the contribution of Stub ASes declines since 2018, while the contribution of Small and Medium ASes remains the same. However, the sum of the 3 categories also remains high, reaching 84%. To better understand the dynamics and how the numbers we see might be biased by the pre-existing ratios of every category, we compute the number of ASes in the entire CAIDA dataset for all categories from 2013-2021. Even though the number of active ASes has substantially increased, from around 45k in 2013 to more than 71k in 2021, the percentage of ASes in each of the aforementioned categories is surprisingly stable. Specifically, Stub ASes are by far the most numerous, with around 85% of all ASes being Stubs. Small ASes are also common, around 12% of all ASes. The rest of the categories are smaller. Medium ASes have a share of 2.6%, Large ASes less than 0.5%, and XLarge ASes less than 0.1%. These are striking differences compared with the percentages reported in Figure 5a, 5b, and 5c for Google's, Netflix's, and Facebook's off-net footprint, respectively. Indeed, the percentage of Stub ASes ranges from 27% to 31%, the percentage of Small ASes is between 41% to 44%, and for Medium ASes it ranges from 22% to 24%. Thus, the demographics of ASes that host the top HGs do not agree with the overall demographics of ASes in the Internet. In the case of Akamai's off-net footprint, the percentage of Stub ASes is even smaller, 13%. Although the share of Large and XLarge ASes in the Internet is a bit more than 0.5%, more than 5% of the ASes that host HGs belong to this category (over 16% in the case of Akamai). Deploying in these large ASes can help HGs serve many users, a subject we elaborate on in Section 6.5.

6.4 Regional Growth

To investigate the HGs' off-net footprint growth in different regions, we assign each AS to one country. We are aware that this may be misleading, especially for XLarge and Large ASes, as they may operate in multiple countries. However, studying a snapshot of the APNIC dataset [64] (see Section 6.5 for more information on this dataset), we observe that 95% of the 26K ASes that are included in this dataset have only one country of operation. To map ASes to countries, we used CAIDA's AS Organizations Dataset [19] (Appendix A.2), resulting in an AS-to-country dataset that spans 7 years and covers 99.9% for the ASes of our study. We compare our AS-to-country dataset with the APNIC dataset [64] and find they agree for 97% of overlapping ASes. However, the APNIC dataset includes many fewer ASes.

The mapping of ASes to countries/regions may be influenced by geopolitics. For example, Hong Kong may appear as part of China

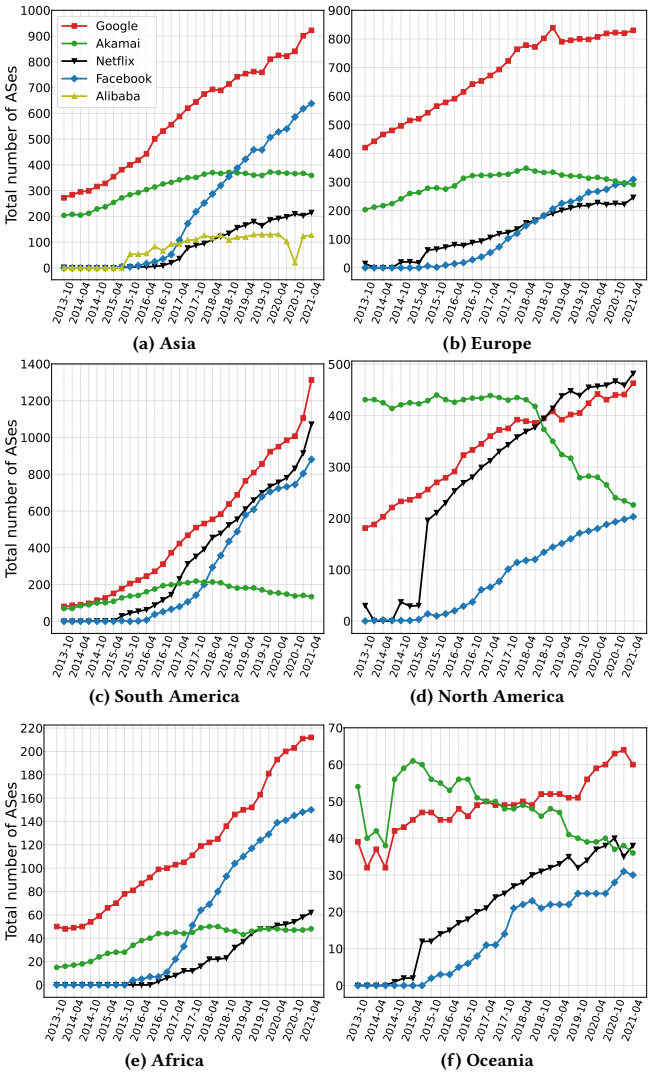


Figure 6: Growth of top-4 HGs per continent over time.

or as a separate region, depending on the source. For these cases, we manually investigate the geographical origin of prefixes, *e.g.*, by utilizing BGP geo-tagged communities [46]. In Figure 6 we plot the growth per continent for the top-4 HGs. In Figures 6a, 6b, and 6c, we present the regions with the highest growth, Asia, Europe, and South America. In 2013, Google had already established a substantial off-net footprint [21], so its growth is linear in both Europe and Asia. Nevertheless, it expands its footprint by 400-600 ASes in each region between 2013 and 2021. Netflix's off-net footprint growth aligns well with the offer of Netflix streaming services in the various regions. Facebook has an aggressive expansion in all regions since the launch of it's CDN. Akamai seems to increase its off-net foorprint in all three regions at similar paces between 2013-2017, before contracting somewhat starting around 2018.

What is really striking is the exponential growth of Google, Netflix, and Facebook in South America. These HGs expand their off-net footprints in more than 800 (and in the case of Google 1,200) ASes from 2013 to 2021. The growth of the other HGs is slower, with the exception of Alibaba that has strong regional growth in Asia. After its launch in late 2014, Alibaba's footprint gradually increases to more than 100 ASes in Asia. A closer investigation by analyzing HTTP(S) headers shows that Alibaba deploys its own hardware servers mainly in Asia and relies on other HGs in other regions.

In Figures 6d, 6e, and 6f, we present areas with lower growth, *i.e.,* North America, Africa, and Oceania. The off-net footprint growth of Google, Netflix, and Facebook is between 200 and 400 ASes in North America, 60-150 ASes in Africa, and 20-30 ASes in Oceania. We attribute this to consolidation in the network market in North America and the relatively small network market in Africa and Oceania. Nevertheless, both Google's and Facebook's off-net footprints include many ASes in Africa. Appendix A.7 investigates the growth of off-net footprints per network type in different regions.

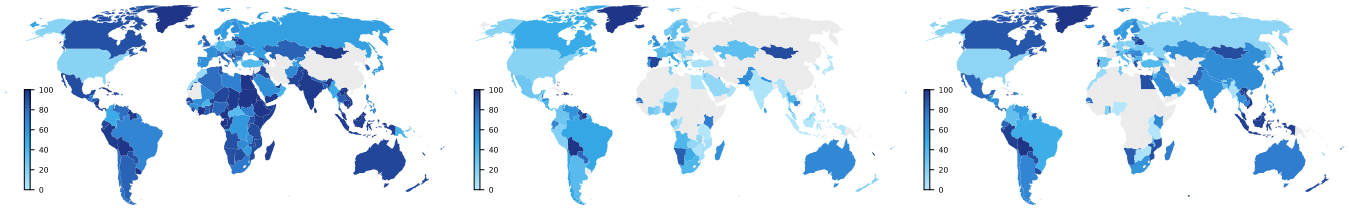
We also noticed a slowdown during the COVID-19 pandemic, but growth continued when the economy opened again in Summer 2020 and especially in the first months of 2021. Anecdotal evidence confirms that additional capacity was allocated in peerings during the pandemic (e.g., for Facebook [62]), as it was more difficult to increase capacity at off-nets inside eyeball networks which can require sending engineers in the field during the lockdown.

6.5 Internet User Population Coverage

Next, we estimate the coverage of Internet user population in a country that can access HGs services located inside their network provider. To estimate this, we need to assign to each AS that hosts at least one of the top-4 HGs its market share in the country where it operates. For this, we use the APNIC dataset [64] that has been used in past studies [45, 59] to characterize ASes based on their Internet user population market share.

APNIC conducts measurement campaigns [65] and publishes the related results on a daily basis. These datasets can be used to estimate the Internet user population percentages per AS, both IPv4 and IPv6, at a country level. We download daily snapshots and we keep only the ASes that have been present in the dataset for at least 25% of each month (one week) to avoid mis-inferences. This filtering reduces the total number of ASes in the dataset from 26k to almost 9k, reducing the coverage of the ASes present in our study to less than 80%. However, by applying such filters we choose to err on the side of accuracy, considering our results as lower bounds of user population percentages. We store monthly snapshots of these data since October 2017.

To estimate the Internet user population coverage that has access to HG servers hosted in its network provider, we add the market share of all the ASes that host HGs and operate in the country and we assign it as a percentage for this country. Figure 7 plots the Internet user percentage per country that has access to off-net HG servers for Google, Netflix, and Akamai in April 2021 using the Rapid7 dataset. For these top HGs, the Internet user coverage per country has not changed dramatically from 2017 to 2021. A closer investigation shows that, in all three cases, the HGs were already hosted by large eyeball networks and by other network providers with high market share in 2017. These observations are inline with our analysis in Section 6.3 regarding the demographics of off-net footprint growth. Although the number of ASes that host



(a) Google (b) Netflix (c) Akamai

Figure 7: Percentage of a country's Internet users in ASes hosting off-net servers of a HG (April 2021).

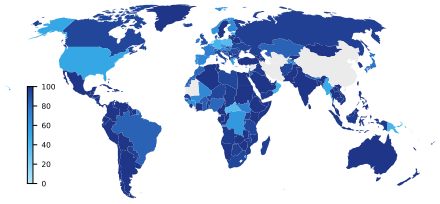
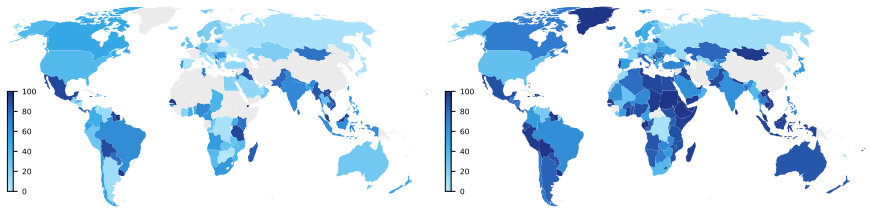


Figure 8: Percentage of a country's Internet users within the customer cones of ASes hosting Google off-net servers (April 2021).



(a) (b)

Figure 9: Percentage of a country's Internet users in ASes hosting Facebook's off-net servers, 10/2017 vs. 04/2021.

Akamai declines, its Internet population coverage does not seem to be affected, as Akamai is still present in networks with large end user population. The top HGs' Internet population coverage varies across regions. For example, Google has strong presence in Africa, and Akamai has high Internet population coverage in Asia. We also verified that the number of countries in which Google claims to have edge caches [51] agrees with our results.

A HG can potentially serve even more of the Internet population by using an off-net to serve not just the users within the AS hosting an off-net but also the users within the customer cone of the AS. Previous studies show that this is the case for Google [21, 27]. In Figure 8 we plot the potential Internet user coverage of Google's off-net footprint, if customer cone users are served by Google servers hosted by their providers. In this case, Google covers more than 50% of the user population in 201 countries, as worldwide coverage increase from 57.8% to 68.2%. User coverage in Europe increases from 58.8% to 77.5%, i.e., an increase of 31.8%, and North America increases by 43.9%, from 49% to 70.6%. Countries with the highest increase are Turkey (from 39.1% to 99%), Colombia (from 48.9% to 98.2%) and Russia (from 54.7% to 94.7%). For Facebook, Netflix and Akamai refer to Appendix A.6.

The Internet population coverage of Facebook's off-net footprint increased significantly between 2017-2021. Figure 9 plots its coverage per country in Oct. 2017 and in Apr. 2021. Facebook's off-net footprint has expanded aggressively to large, medium and small ASes around the globe. As we discussed in Section 6.4, Facebook's footprint growth in absolute number of ASes was especially apparent in South America, Asia, and Africa. Smaller increases are also visible in Europe and Asia.

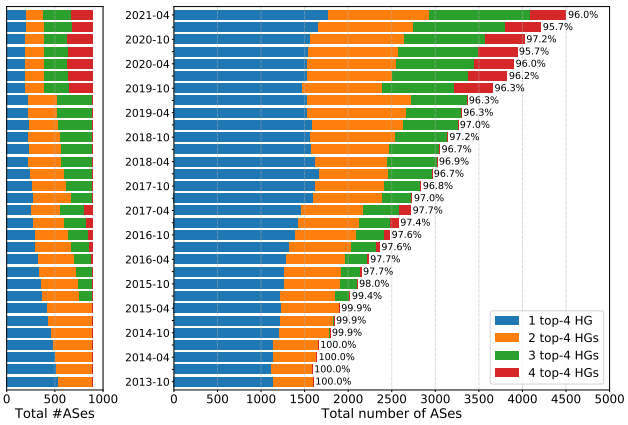
Due to the varying landscape of the telecommunications sector across the world, the growth of Facebook's off-net footprint in various regions yields different increases in the percent of users that can be served. For example, in Africa an increase of the off-net footprint by about 99 ASes yields an increase of more than 115% in user population coverage, from 34.7% to 74.8%. In Europe the increase of the off-net footprint of 206 ASes yields an increase of user population coverage of 136%, from 16.9% to 39.8%. In contrast, in South America, the increase in user coverage was only 32% (from 51.6% to 68%), although the increase of the Facebook's off-net footprint was more than 739 ASes. Facebook has also announced that it had plans to expand in Africa and other developing regions [91]. Our results show that, indeed, there is a significant expansion of Facebook there. The increased coverage by Facebook is also related to the expansion strategy of its CDN in the last two years, discussed in Section 6.4.

Our results show that some regions remain under-covered, although the specifics vary across HGs. The reasons may vary, including politics and business, and we prefer not to speculate on the root causes. However, our analysis can provide insights on the potential increase of user population coverage. For example, Facebook could significantly increase coverage in the US from 33.9% to 61.8%, by deploying off-net servers in only following 5 ASes (AS7018, AS21928, AS20115, AS20057, and AS22394).

6.6 Network Providers' Hosting Strategies

We next turn our attention to the network providers' strategies for hosting HGs. Our results show that, in terms of which networks host HG off-nets, the footprints of HGs overlap, especially those of the four largest HGs. We also demonstrate that a network that already hosts one large HG is likely to later host more.

For all ASes that are part of at least one of the four largest off-net footprints, those of Google, Netflix, Facebook, and Akamai, Figure 10b plots the distribution (across time) of the number of those four HGs that they hosted. The number of such ASes has almost tripled between 2013 and 2021. A striking observation is that the majority of the ASes that host HGs (more than 97%, see percentages in the plot), host at least one of the top-4 HGs. Thus, there is only a tiny percentage of ASes that host only HGs outside the top-4. The



(a) (b)

Figure 10: # ASes that host at least one top-4 HGs (Google, Netflix, Facebook, Akamai) in (a) every snapshot and (b) more than one snapshot.

top-4 HGs have increasingly similar footprints, being present in the same networks. In 2021, more than 70% of ASes that host an off-net host 2-4 top-4 HGs, compared to less than 30% of these ASes in 2013.

We also investigate if networks that have already hosted top-4 HGs are willing to host more. We focus on networks that throughout 2013-2021 hosted at least one top-4 HGs. In total, there are 1,002 such networks. In Figure 10a we plot the distribution of top-4 HGs these networks host. In 2013, there were only 450 ASes that hosted 2 or more top-4 HGs. In 2021, this number increased to more than 800. Moreover, although none of these networks hosted all top-4 HGs in 2013, more than 250 of them host all top-4 HGs in 2021. We provide additional results for networks that start hosting HGs later than 2013 in Appendix A.8, but the conclusions are consistent with these observations. We conclude that networks that host top-4 HGs are, typically, willing to host more as time passes.

7 LIMITATIONS

SNI. The main dataset (Rapid7) that we use for our longitudinal study reports only the default certificate of an IP address running on port 443 using the `HTTPS` protocol. A server may host multiple sites on the same IP, but it reports only the default certificate if no SNI value is present in client requests. This practice is widely adopted and used by HGs to provide service to their customers [41, 79].

Certificates in IPv6 addresses. The corpuses of available certificate datasets are for the IPv4 address space, but there are a very small number of IPv6-only mobile operators for which our approach will not work. We plan to investigate this as part of future work. While our inference approach is IP protocol-agnostic, we lack IPv6 data to conduct longitudinal analysis, and achieving broad coverage in scans of the enormous IPv6 space brings new challenges.

HGs' Operational Practices. Although our methodology is generic, there are some unusual operational practices by some HGs that complicate our approach. For example, CloudFlare offers a proxy service in front of customers and issues a certificate that enables secure communication between the customer server and CloudFlare's middleware [74]. When these certificates are hosted by customers' backend servers in different networks, it appears like CloudFlare has off-nets there. To mitigate this issue, we have to manually identify and remove these certificates in our study. We noticed that CloudFlare includes an additional entry in the `TLS dNSNames` field of their free customer certificates (matching `(ssl|sni)[0-9]*.cloudflaressl.com`), enabling us to filter on it. Although this process filters out the vast majority of Cloudflare customer certificates, additional manual investigation is required, as CloudFlare offers a paid service for dedicated certificates, and it also allows enterprise and business Tier customers to upload their own custom TLS certificates.

Anycast. In the case of IP anycast, the same IP address (belonging to the HG AS) is used to serve content from all sites. From discussions with two operators, we learned that their anycast deployments include off-nets, which additionally use the BGP `no-export` community to keep the announcement local. The use of this same IP address will make all sites appear to belong to the HG's AS. However, we learned it is common also assign each off-net a unicast IP address belonging to the hosting AS, to help with debugging. While these IP addresses are generally not given out by the HG's authoritative DNS to serve production traffic, they do respond in the same way as the production anycast IP. Our approach will discover these unicast IP addresses correctly, but there is no guarantee that operators will configure their networks in this way.

Reverse Proxies and Cache Misses. There are instances in our HTTP(S) scans dataset where the edge site terminating the client's TCP connection belongs to a third-party HG but the request gets forwarded to a backend server in a different HG. This forwarding behavior is usually the result of a CDN cache miss going to the origin or split TCP [26, 82] to an application server. In these cases, the response received by the client will include both the origin HG and edge HG headers. This can confuse our inference confirmation with headers, but in practice we observe this only in 4% of measurements. Of these, 99% are Akamai (64%) and Cloudflare (35%) edges with origins primarily deployed in Amazon S3. This insight allows us to prioritize third-party CDN headers, like Akamai and Cloudflare, as the edge site in the presence of a conflict.

Missing Headers. Some networks, such as Netflix and Hulu, only include debug or headers when serving logged-in users. These are missing from our input datasets and prevent us from performing header confirmation for these HGs.

In-Rotation Servers. Because our datasets "skip" the DNS control-plane of HG content delivery, we cannot currently distinguish if a particular IP address is actively in-use by the HG for serving content. A given IP may be drained [9] but responsive or may belong to a testing environment that always responds when queried.

8 DISCUSSION

An Increasingly Private Internet. Our study shows that many HGs have already installed substantial serving infrastructure deep inside networks, especially eyeball networks. This can be a blessing if these installations can improve end user performance. For example, it has been reported that round trip times between end-users and these servers can be even less than a millisecond [85, 103]. Such low delays are a step towards addressing the future requirements and a way for Hypergiants like Google and Facebook to play an

important role in future services, including 5G. The extended serving infrastructure can also better deal with increased traffic, e.g., during the COVID-19 pandemic [42], as was reported recently by Google [55] and Akamai [69]. At the same time, these deployments mean that a substantial fraction of traffic demand can be served locally without crossing inter-domain links or private interconnections [80, 85, 89, 110, 112, 113]. This "zero AS-hop" Internet has regulatory implications. Servers which are located inside a network to exclusively serve users of that network, e.g., IPTV, "private" CDN clusters, can be considered specialized services [67]. In this setting, it is debatable if network neutrality regulations apply [99], since exceptions may be granted for such specialized services.

When ISPs allow the deployment of servers within their network, they can lose negotiation power in peering agreements with HGs, because the servers reduce the amount of traffic traversing peering interconnections [68, 72, 106]. Smaller networks may see an opportunity to host such servers and reduce their upstream or downstream traffic, thus improving the service they offer to their customers and reducing upstream provider cost and reliance (increasingly bypassing the public Internet) [39, 63, 67]. Nevertheless, HGs and ISPs have to negotiate who will cover the operational cost of hosting content servers (power, bandwidth, maintenance), that can be another source of dispute.

Unintended Consequences. Widespread adoption of TLS– intended to protect users' privacy and support secure protocols – had the unintended consequence of providing the basis for revealing HGs' footprints. Our study shows that it is feasible to infer HGs' off-net footprint, using only publicly available data. We view this outcome as important to understand the dynamics that underpin the phenomenal growth of network traffic and shape the Internet's topology, but this information may have other uses – some less benign. First, knowledge of the network locations and IP addresses of HG servers makes it easier for attackers to be effective. They can use this information to better orchestrate DDoS attacks or target specific servers that deal with critical services or have high financial interest for them. Servers that are deployed outside HG networks may be more vulnerable to attacks, as the security measures and available capacity may not meet the standards of HG datacenters. Authoritarian regimes can also collect intelligence about deployments in their countries or in other countries for their benefit, such as for surveillance purposes. Second, knowledge of HGs' off-net footprints can be utilized for business intelligence by competitors. Knowing where servers are deployed is considered a business secret, as competitors may decide to place their servers in the same location or not to achieve their own strategic objectives. It also allows for inference of deployment and expansion strategies of HGs over time, which could inform investment decisions and planning of competitors. It can also provide the opportunity to new cloud providers to expand and prioritize their server deployment in a more optimized way.

Hide-and-Seek. HGs may want to hide their footprints from our detection methodology for confidentiality and security reasons. There are a number of possible approaches:

(1) Increasing the bar for deployment identification is rather simple: the default certificate should not disclose information, e.g., by setting a null certificate, and the certificate should be presented only when there is a TLS-SNI request for specific domains. These changes would make existing datasets (e.g., Rapid7) less suitable to our methodology, but they are surmountable at the cost of increased measurement overhead with global scans for fully qualified SNI domains. In our study, we noticed two cases where the null default certificates hamper our detection methodology. One is the case of Netflix between 2017-04 and 2019-10, when a large number of Netflix servers identifiable by certificate due to use of HTTP (not HTTPS). The second case are on-net servers of Google, which only answer TLS-SNI requests for specific first-party domains (e.g. www.google.com).

(2) HGs could instruct off-net servers to respond only to requests originated by the customer cone of the hosting network, although the risk of blackholing legitimate traffic may not be worthwhile.

(3) HGs can modify their certificate content by altering fields that we currently use to infer ownership and to extract fingerprints. For example, remove the `Organization` entry from the `Subject Name` of the EE certificate or use unique domain names per off-net deployment.

(4) Anonymizing headers or using bot-detection to exclude headers also helps in hiding the identity of HGs' off-nets, as it blinds our validation technique. However, headers are commonly used for debugging, so removing them may increase diagnostic complexity for operators.

(5) IPv6-only HG servers would be invisible to IPv4 global Internet scans used by Censys and Rapid7 but increases the risk of outages for IPv4-only clients.

Our methodology relies on the fact that HGs include company information in their TLS certificates to help establish their identifies to users. Despite these possible approaches, we expect that the central idea behind our methodology will continue to work, as HGs will always need to provide organizational information to prove their identity.

9 CONCLUSION

Hypergiants are responsible for a significant fraction of the traffic delivered to end users, and they contributed to the notable consolidation and privatization of Internet infrastructure. HGs have been expanding their infrastructures towards and into eyeball networks, including via off-net servers inside those networks, to be as close as possible to the end users. However, as these Hypergiants grew in importance, their serving infrastructures become less and less visible to traditional measurement techniques, and mapping their expansions was an unmet challenge. In this work, we developed a generic methodology to measure their expansion, leveraging more than 7 years (2013-2021) worth of information extracted from corpuses of TLS certificate scans and other active measurements. We observe that the number of ASes hosting HGs' off-nets has more than doubled during this period, with the vast majority of them hosting at least one of the top-4 HG (Google, Netflix, Facebook, and Akamai), and tending to host more of them over time. Recently, growth has been particularly fast in Europe, Asia, and, especially, Latin America. Consequently, these large Hypergiants can serve large fractions of the world's Internet users directly from within the users' networks. Our study opens interesting research directions on Internet privatization, content delivery, and security practices.

ACKNOWLEDGEMENT

We would like to thank our shepherd, Olivier Bonaventure and the anonymous reviewers for their valuable feedback. We also greatly appreciate the Hypergiant operators who validated our results. We are grateful to Rapid7 [86] and Censys [35] for providing us research access to their datasets. This work and its dissemination efforts are partially supported by the European Research Council (ERC) Starting Grant ResolutioNet (ERC-StG-679158) and by NSF awards CNS-1836872 and CNS-2028550.

REFERENCES

[1] Josh Aas, Richard Barnes, Benton Case, Zakir Durumeric, Peter Eckersley, Alan Flores-López, J. Alex Halderman, Jacob Hoffman-Andrews, James Kasten, Eric Rescorla, Seth Schoen, and Brad Warren. 2019. Let's Encrypt: An Automated Certificate Authority to Encrypt the Entire Web. (2019).

[2] Vijay Kumar Adhikari, Sourabh Jain, Yingying Chen, and Zhi-Li Zhang. 2012. Vivisecting Youtube: An Active Measurement Study. In *IEEE INFOCOM*.

[3] Bernhard Ager, Wolfgang Mühlbauer, Georgios Smaragdakis, and Steve Uhlig. 2011. Web Content Cartography. In *ACM IMC*.

[4] Akamai. 2020. Facts & Figures. https://www.akamai.com/us/en/about/facts-figures.jsp.

[5] Akamai. 2021. EdgeWorkers User Guide: 500 Internal Server error. https://learn.akamai.com/en-us/webhelp/edgeworkers/edgeworkers-user-guide/GUID-04E378BA-DBD2-40F9-91BB-912E0FFFD2A4.html.

[6] Alibaba. 2020. Alibaba Cloud: EdgeScript. https://static-aliyun-doc.oss-cn-hangzhou.aliyuncs.com/download%2Fpdf%2F126597%2FEdgeScript_intl_en-US.pdf.

[7] Alibaba. 2021. Alibaba Cloud CDN. https://www.alibabacloud.com/product/cdn.

[8] Amazon. 2021. CloudFront Developer Guide. https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/RequestAndResponseBehaviorCustomOrigin.html.

[9] Amazon. 2021. Configure connection draining for your Classic Load Balancer. https://docs.aws.amazon.com/elasticloadbalancing/latest/classic/config-conn-drain.html.

[10] Todd Arnold, Ege Gurmericliler, Georgia Essig, Arpit Gupta, Matt Calder, Vasileios Giotsas, and Ethan Katz-Bassett. 2020. (How Much) Does a Private WAN Improve Cloud Performance?. In *IEEE INFOCOM*.

[11] Todd Arnold, Jia He, Weifan Jiang, Matt Calder, Italo Cunha, Vasileios Giotsas, and Ethan Katz-Bassett. 2020. Cloud Provider Connectivity in the Flat Internet. In *ACM IMC*.

[12] Shehar Bano, Philipp Richter, Mobin Javed, Srikanth Sundaresan, Zakir Durumeric, Steven J. Murdoch, Richard Mortier, and Vern Paxson. 2018. Scanning the Internet for Liveness . *ACM SIGCOMM Computer Communication Review* 48, 2 (2018).

[13] Anurag Bhatia. 2018. Mapping Facebook's FNA (CDN) nodes across the world! https://anuragbhatia.com/2018/03/networking/isp-column/mapping-facebooks-fna-cdn-nodes-across-the-world/.

[14] Anurag Bhatia. 2019. Facebook FNA node update. https://anuragbhatia.com/2019/11/networking/isp-column/facebook-fna-node-update/.

[15] Anurag Bhatia. 2021. Facebook FNA updates – April 2021. https://anuragbhatia.com/2021/04/networking/isp-column/facebook-fna-updates-april-2021/.

[16] Zachary S. Bischof, Romain Fontugne, and Fabian E. Bustamante. 2018. Untangling the world-wide mesh of undersea cables. In *HotNets*.

[17] Timm Böttger, Felix Cuadrado, Gareth Tyson, Ignacio Castro, and Steve Uhlig. 2018. Open Connect everywhere: A glimpse at the internet ecosystem through the lens of the Netflix CDN. *ACM SIGCOMM Computer Communication Review* 48, 1 (2018).

[18] Timm Böttger, Felix Cuadrado, and Steve Uhlig. 2018. Looking for Hypergiants in PeeringDB. *ACM SIGCOMM Computer Communication Review* 48, 3 (2018).

[19] CAIDA. 2013-2021. The CAIDA AS Organizations Dataset. http://www.caida.org/data/as-organizations/.

[20] CAIDA. 2013-2021. The CAIDA AS Relationships Dataset. http://www.caida.org/data/active/as-relationships/.

[21] Matt Calder, Xun Fan, Zi Hu, Ethan Katz-Bassett, John Heidemann, and Ramesh Govindan. 2013. Mapping the Expansion of Google's Serving Infrastructure. In *ACM IMC*.

[22] Matt Calder, Ashley Flavel, Ethan Katz-Bassett, Ratul Mahajan, and Jitendra Padhye. 2015. Analyzing the Performance of an Anycast CDN. In *ACM IMC*.

[23] Frank Cangialosi, Taejoong Chung, David Choffnes, Dave Levin, Bruce M. Maggs, Alan Mislove, and Christo Wilson. 2016. Measurement and analysis of private key sharing in the https ecosystem. In *ACM CCS*.

[24] CDNetworks. 2021. HTTP Protocol Optimization. https://documents.cdnetworks.com/document/cate/16023/16281.

[25] Fangfei Chen, Ramesh K. Sitaraman, and Marcelo Torres. 2015. End-User Mapping: Next Generation Request Routing for Content Delivery. In *ACM SIGCOMM*.

[26] Yingying Chen, Sourabh Jain, Vijay Kumar Adhikari, and Zhi-Li Zhang. 2011. Characterizing roles of front-end servers in end-to-end performance of dynamic content distribution. In *IMC*.

[27] Yi-Ching Chiu, Brandon Schlinker, Abhishek Balaji Radhakrishnan, Ethan Katz-Bassett, and Ramesh Govindan. 2015. Are We One Hop Away from a Better Internet?. In *ACM IMC*.

[28] Taejoong Chung, Yabing Liu, David Choffnes, Dave Levin, Bruce MacDowell Maggs, Alan Mislove, and Christo Wilson. 2016. Measuring and applying invalid SSL certificates: The silent majority. In *ACM IMC*.

[29] Danilo Cicalese, Jordan Augé, Diana Joumblatt, Timur Friedman, and Dario Rossi. 2015. Characterizing IPv4 Anycast Adoption and Deployment. In *CoNEXT*.

[30] Cloudflare. 2021. Gathering information for troubleshooting sites. https://support.cloudflare.com/hc/en-us/articles/203118044-Gathering-information-for-troubleshooting-sites.

[31] Craig Labovitz. 2019. Internet Traffic 2009-2019. APRICOT 2019.

[32] David Cooper, Stefan Santesson, Stephen Farrell, Sharon Boeyen, Russ Housley and Tim Polk. 2008. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile - RFC-5280. https://tools.ietf.org/html/rfc5280.

[33] Digicert. 2020. What fields are required when generating a Certificate Signing Request (CSR)? https://knowledge.digicert.com/solution/SO16317.html.

[34] Donald Eastlake 3rd. 2011. Transport Layer Security (TLS) Extensions: Extension Definitions. https://tools.ietf.org/html/rfc6066.

[35] Zakir Durumeric, David Adrian, Ariana Mirian, Michael Bailey, and J. Alex Halderman. 2015. A Search Engine Backed by Internet-Wide Scanning. In *CCS*.

[36] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. 2013. ZMap: Fast Internet-wide scanning and its security applications. In *USENIX Security*.

[37] Facebook. 2014. Introducing Proxygen, Facebook's C++ HTTP Framework. https://engineering.fb.com/2014/11/05/production-engineering/introducing-proxygen-facebook-s-c-http-framework/.

[38] Facebook. 2021. Graph API Debugging. https://developers.facebook.com/docs/graph-api/using-graph-api/debugging/.

[39] Peyman Faratin, David D. Clark, Steven Bauer, William Lehr, Patrick Gilmore, and Arthur Berger. 2008. The Growing Complexity of Internet Interconnection. *Communications and Strategies* 72 (2008), 51.

[40] Fastly. 2021. X-Served-By. https://developer.fastly.com/reference/http-headers/X-Served-By/.

[41] Marwan Fayed, Lorenz Bauer, Vasileios Giotsas, Sami Kerola, Marek Majkowski, Pavel Odinstov, Jakub Sitnicki, Taejoong Chung, Dave Levin, Alan Mislove, Christopher A. Wood, and Nick Sullivan. 2021. The Ties that un-Bind: Decoupling IP from web services and sockets for robust addressing agility at CDN-scale. In *SIGCOMM*.

[42] Anja Feldmann, Oliver Gasser, Franziska Lichtblau, Enric Pujol, Ingmar Poese, Christoph Dietzel, Daniel Wagner, Matthias Wichtlhuber, Juan Tapiador, Narseo Vallina-Rodriguez, Oliver Hohlfeld, and Georgios Smaragdakis. 2021. A Year in Lockdown: How the Waves of COVID-19 Impact Internet Traffic. *Communications of the ACM* 64, 7 (July 2021), 101–108.

[43] Tobias Flach, Nandita Dukkipati, Andreas Terzis, Barath Raghavan, Neal Cardwell, Yuchung Cheng, Ankur Jain, Shuai Hao, Ethan Katz-Bassett, and Ramesh Govindan. 2013. Reducing Web Latency: the Virtue of Gentle Aggression. In *ACM SIGCOMM*.

[44] Petros Gigis, Matt Calder, Lefteris Manassakis, George Nomikos, Vasileios Kotronis, Xenofontas Dimitropoulos, Ethan Katz-Bassett, and Georgios Smaragdakis. 2021. Seven Years in the Life of Hypergiants' Off-Nets Artifacts: Project Repository and Website. https://github.com/pgigis/sigcomm2021-hypergiants-offnets.

[45] Petros Gigis, Vasileios Kotronis, Emile Aben, Stephen D Strowes, and Xenofontas Dimitropoulos. 2017. Characterizing user-to-user connectivity with Ripe Atlas. In *Proceedings of the Applied Networking Research Workshop*. 4–6.

[46] Vasileios Giotsas, Matthew Luckie, Bradley Huffaker, and kc claffy. 2014. Inferring Complex AS Relationships. In *IMC*.

[47] Vasileios Giotsas, Georgios Smaragdakis, Bradley Huffaker, Matthew Luckie, and kc claffy. 2015. Mapping Peering Interconnections at the Facility Level. In *CoNEXT*.

[48] Google. 2012. Chicago - Jobs - Google. https://web.archive.org/web/20120415021232/http://www.google.com/intl/en/jobs/uslocations/chicago/index.html.

[49] Google. 2020. Google Peering. https://peering.google.com/#/options/peering.

[50] Google. 2020. Google Transparency Report: HTTPS encryption on the web. https://transparencyreport.google.com/https/overview.

[51] Google. 2021. Google Global Cache. https://peering.google.com/#/options/google-global-cache.

[52] Google. 2021. Google's Edge Network. https://peering.google.com/.

[53] Chi-Yao Hong, Subhasree Mandal, Mohammad Al-Fares, Min Zhu, Richard Alimi, Kondapa Naidu B., Chandan Bhagat, Sourabh Jain, Jay Kaimal, Shiyu Liang, Kirill Mendelev, Steve Padgett, Faro Rabe, Saikat Ray, Malveeka Tewari,

Matt Tierney, Monika Zahn, Jonathan Zolla, Joon Ong, and Amin Vahdat. 2018. B4 and after: managing hierarchy, partitioning, and asymmetry for availability and scale in google's software-defined WAN. In *ACM SIGCOMM*.
[54] Cheng Huang, Angela Wang, Jin Li, and Keith W. Ross. 2008. Measuring and Evaluating Large-scale CDNs. In *IMC*.
[55] Urs Hölzle. 2020. Keeping our network infrastructure strong amid COVID-19. https://www.blog.google/inside-google/infrastructure/keeping-our-network-infrastructure-strong-amid-covid-19/.
[56] IANA. 2021. IANA IPv4 Special-Purpose Address Registry. https://www.iana.org/assignments/iana-ipv4-special-registry/iana-ipv4-special-registry.xhtml.
[57] IANA. 2021. IANA Special-Purpose Autonomous System (AS) Numbers. https://www.iana.org/assignments/iana-as-numbers-special-registry/iana-as-numbers-special-registry.xhtml.
[58] Jonathan Dingman. 2008. Interview with Bharat Mediratta About the Google Web Server. https://web.archive.org/web/20080416222246/http://www.ginside.com/2008/1489/interview-bharat-mediratta/.
[59] Vasileios Kotronis, George Nomikos, Lefteris Manassakis, Dimitris Mavrommatis, and Xenofontas Dimitropoulos. 2017. Shortcuts Through Colocation Facilities. In *ACM IMC*.
[60] Reinhardt Krause. 2016. Akamai Revenue Growth Tanks As Apple, Facebook Shift Traffic. https://www.investors.com/news/technology/akamai-revenue-growth-tanks-as-apple-facebook-shift-traffic/.
[61] Logan Kugler. 2020. How the Internet Spans the Globe. *Comm. of the ACM* 63, 1 (2020).
[62] Craig Labovitz. 2020. Pandemic Impact on Global Internet Traffic. NANOG 79.
[63] Craig Labovitz, Scott Iekel-Johnson, Danny McPherson, Jon Oberheide, and Farnam Jahanian. 2010. Internet Inter-Domain Traffic. In *ACM SIGCOMM*.
[64] APNIC Labs. 2021. AS population data. https://stats.labs.apnic.net/aspop/.
[65] APNIC Labs. 2021. IPv6 measurement campaign. https://labs.apnic.net/measureipv6/.
[66] Rapid7 Labs and Project Sonar. 2021. Rapid7 Labs IPv4 SSL Certificates from Project Sonar. https://opendata.rapid7.com/sonar.ssl/.
[67] William Lehr, David D. Clark, Steven Bauer, Arthur Berger, and Philipp Richter. 2019. Whither the public Internet? *J. of Information Policy* 9 (2019).
[68] Tom Leighton. 2009. Improving Performance on the Internet. *Comm. of the ACM* 52, 2 (2009).
[69] Tom Leighton. 2020. Can the Internet keep up with the surge in demand? https://blogs.akamai.com/2020/04/can-the-internet-keep-up-with-the-surge-in-demand.html.
[70] Limelight. 2014. Troubleshooting using HTTP Headers. https://www.youtube.com/watch?v=-0SwWUYXwgs.
[71] Victor Luckerson. 2014. Netflix's Disputes With Verizon, Comcast Under Investigation. https://time.com/2871498/fcc-investigates-netflix-verizon-comcast/.
[72] Matthew Luckie, Amogh Dhamdhere, David Clark, Bradley Huffaker, and kc claffy. 2014. Challenges in Inferring Internet Interdomain Congestion. In *IMC*.
[73] Srdjan Matic, Gareth Tyson, and Gianluca Stringhini. 2019. Pythia: a Framework for the Automated Analysis of Web Hosting Environments. In *WWW*.
[74] Matthew Prince. 2014. Introducing Universal SSL - Cloudflare. https://blog.cloudflare.com/introducing-universal-ssl/.
[75] Microsoft. 2020. Protocol support for HTTP headers in Azure Front Door. https://docs.microsoft.com/en-us/azure/frontdoor/front-door-http-headers-protocol.
[76] Mozilla. 2021. Common CA Database (CCADB). www.ccadb.org.
[77] RIPE NCC. 2021. Routing Information Service (RIS). https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris.
[78] Netflix. 2021. Netflix Open Connect. https://openconnect.netflix.com/.
[79] Erik Nygren. 2017. Reaching toward universal TLS SNI. https://blogs.akamai.com/2017/03/reaching-toward-universal-tls-sni.html.
[80] Erik Nygren, Ramesh K. Sitaraman, and Jennifer Sun. 2010. The Akamai Network: A Platform for High-performance Internet Applications. *SIGOPS Oper. Syst. Rev.* 44, 3 (2010).
[81] University of Oregon. 2021. RouteViews Project. http://www.routeviews.org/routeviews/.
[82] Abhinav Pathak, Y. Angela Wang, Cheng Huang, Albert Greenberg, Y. Charlie Hu, Randy Kern, Jin Li, and Keith W. Ross. 2010. Measuring and Evaluating TCP Splitting for Cloud Services. In *PAM*.
[83] Bryan Payne. 2016. PKI at Scale Using Short-lived Certificates. In *USENIX Enigma Conference Presentation*.
[84] Enric Pujol, Ingmar Poese, Johannes Zerwas, Georgios Smaragdakis, and Anja Feldmann. 2019. Steering Hyper-Giants' Traffic at Scale. In *CoNEXT*.
[85] Enric Pujol, Philipp Richter, Balakrishnan Chandrasekaran, Georgios Smaragdakis, Anja Feldmann, Bruce Maggs, and Keung-Chi Ng. 2014. Back-Office Web Traffic on The Internet. In *ACM IMC*.
[86] Rapid7 Research. 2021. Project Sonar: Gaining Insights into Global Exposure. https://www.rapid7.com/research/project-sonar/.
[87] Mario A. Sanchez, John S. Otto, Zachary S. Bischofand David R. Choffnes, Fabian E. Bustamante, Balachander Krishnamurthy, and Walter Willinger. 2013. Dasu: Pushing Experiments to the Internet's Edge . In *NSDI*.
[88] Brandon Schlinker, Italo Cunha, Yi-Ching Chiu, Srikanth Sundaresan, and Ethan Katz-Bassettt. 2019. Internet Performance from Facebook's Edge. In *ACM IMC*.
[89] Brandon Schlinker, Hyojeong Kim, Timothy Cui, Ethan Katz-Bassett, Harsha V. Madhyastha, Italo Cunha, James Quinn, Saif Hasan, Petr Lapukhov, and Hongyi Zeng. 2017. Engineering Egress with Edge Fabric: Steering Oceans of Content to the World. In *ACM SIGCOMM*.
[90] Sectigo. 2021. What Is an X.509 Certificate & How Does It Work? https://sectigo.com/resource-library/what-is-x509-certificate.
[91] Rijurekha Sen, Sohaib Ahmad, Amreesh Phokeer, Zaid Ahmed Farooq, Ihsan Ayyub Qazi, David Choffnes, and Krishna P. Gummadi. 2017. Inside the Walled Garden: Deconstructing Facebook's Free Basics Program. *ACM SIGCOMM Computer Communication Review* 47, 5 (2017).
[92] Pavlos Sermpezis, Vasileios Kotronis, Petros Gigis, Xenofontas Dimitropoulos, Danilo Cicalese, Alistair King, and Alberto Dainotti. 2018. ARTEMIS: Neutralizing BGP hijacking within a minute. *IEEE/ACM Trans. Networking* 26, 6 (2018).
[93] Muhammad Shuaib Siddiqui, Diego Montero, Marcelo Yannuzzi, Rene Serral-Gracia, and Xavi Masip-Bruin. 2014. Route leak identification: A step toward making Inter-Domain routing more reliable. In *IEEE DRCN*.
[94] Rachee Singh, Arun Dunna, and Phillipa Gill. 2018. Characterizing the Deployment and Performance of Multi-CDNs. In *ACM IMC*.
[95] Rachee Singh, Manya Ghobadi, Klaus-Tycho Foerster, Mark Filer, and Phillipa Gill. 2018. RADWAN: Rate Adaptive Wide Area Network. In *ACM SIGCOMM*.
[96] Square Open Source. 2021. `certigo`. https://github.com/square/certigo.
[97] Neil Spring, Ratul Mahajan, and David Wetherall. 2002. Measuring ISP Topologies with Rocketfuel. In *ACM SIGCOMM*.
[98] Randall Stewart and Scott Long. 2016. Improving High–Bandwidth TLS in the FreeBSD kernel. *FreeBSD Journal* (2016).
[99] Volker Stocker, Georgios Smaragdakis, and William Lehr. 2020. The State of Network Neutrality Regulation. *ACM SIGCOMM Computer Communication Review* 50, 1 (2020).
[100] Florian Streibelt, Jan Boettger, Nikolaos Chatzis, Georgios Smaragdakis, and Anja Feldmann. 2013. Exploring EDNS-client-subnet adopters in your free time. In *ACM IMC*.
[101] Ao-Jan Su, David Choffnes, Aleksandar Kuzmanovic, and Fabian Bustamante. 2006. Drafting behind Akamai (travelocity-based detouring). In *ACM SIGCOMM*.
[102] Ruben Torres, Alessandro Finamore, Jin-Ryong Kim, Marco Mellia, Maurizio M. Munafo, and Sanjay Rao. 2011. Dissecting video server selection strategies in the youtube CDN. In *ICDCS*.
[103] Martino Trevisan, Danilo Giordano, Idilio Drago, Maurizio Matteo Munafo, and Marco Mellia. 2018. Five Years at the Edge: Watching Internet from the ISP Network. In *ACM CoNEXT*.
[104] Sipat Triukose, Zhihua Wen, and Michael Rabinovich. 2011. Measuring a Commercial Content Delivery Network. In *WWW*.
[105] Twitter. 2021. Using Twirl. https://developer.twitter.com/en/docs/tutorials/using-twurl.
[106] Amin Vahdat, David Clark, and Jennifer Rexford. 2016. A purpose-built global network: Google's move to SDN. *Comm. of the ACM* 59, 3 (2016).
[107] Verizon Digital Media Services. 2020. Response. https://docs.vdms.com/cdn/Content/HTTP_and_HTTPS_Data_Delivery/Response.htm.
[108] Pierre-Antoine Vervier, Olivier Thonnard, and Marc Dacier. 2015. Mind Your Blocks: On the Stealthiness of Malicious BGP Hijacks. In *NDSS*.
[109] Gerry Wan, Liz Izhikevich, David Adrian, Katsunari Yoshioka, Ralph Holz, Christian Rossow, and Zakir Durumeric. 2020. On the Origin of Scanning: The Impact of Location on Internet-Wide Scans. In *ACM IMC*.
[110] Florian Wohlfart, Nikolaos Chatzis, Caglar Dabanoglu, Georg Carle, and Walter Willinger. 2018. Leveraging Interconnections for Performance: The Serving Infrastructure of a Large CDN. In *SIGCOMM*.
[111] Jing'an Xue, David Choffnes, and Jilong Wang. 2017. CDNs Meet CN An Empirical Study of CDN Deployments in China. *IEEE Access* 5, 1 (2017).
[112] Kok-Kiong Yap, Murtaza Motiwala, Jeremy Rahe, Steve Padgett, Matthew Holliman, Gary Baldus, Marcus Hines, Taeeun Kim, Ashok Narayanan, Ankur Jain, Victor Lin, Colin Rice, Brian Rogan, Arjun Singh, Bert Tanaka, Manish Verma, Puneet Sood, Mukarram Tariq, Matt Tierney, Dzevad Trumic, Vytautas Valancius, Calvin Ying, Mahesh Kallahalla, Bikash Koley, and Amin Vahdat. 2017. Taking the Edge off with Espresso: Scale, Reliability and Programmability for Global Internet Peering. In *ACM SIGCOMM*.
[113] Bahador Yeganeh, Ramakrishnan Durairajan, Reza Rejaie, and Walter Willinger. 2019. How Cloud Traffic Goes Hiding: A Study of Amazon's Peering Fabric. In *IMC*.
[114] ZGrab2. 2021. Fast Go Application Scanner. https://github.com/zmap/zgrab2.

A APPENDICES

Appendices are supporting material that has not been peer-reviewed.

A.1 IP-to-AS Mapping

To address the challenge of mapping IP addresses to ASes with high accuracy and coverage, we combine data from two well-known publicly available datasets, namely RIPE RIS [77] and RouteViews [81]. These datasets contain control-plane information of BGP announcements (BGP RIBs and updates), enabling the mapping of an AS (or more than one ASes in Multi-Origin/MOAS cases) to the IP prefixes that they announce in BGP. In compliance with the timeline of our study, we obtain more than 7 years (October 2013 - April 2021) of daily data from each dataset. We aggregate them in monthly snapshots and we filter out reserved IP prefixes [56] and ASes [57]. Moreover, since some of the information (such as the origin AS of the prefix) seen in BGP might be "tainted", e.g., due to BGP hijacks [92] or route leaks [93], we preserve only the IP-to-AS mappings that consistently appeared for more than 25% of the total time per monthly snapshot (i.e., more than a week – less than 2% of BGP hijacks last longer than a week according to [108]). We then merge the two datasets. In case they contain conflicting information on the AS mapping for the same prefix, we consider all associated ASes as valid mappings and treat the case as BGP MOAS. This process results in a coverage of 75.8% of the publicly routable IPv4 address space, on average across time.

A.2 On-Net Hypergiant Footprint

To identify the AS(es) of each HG, i.e., the on-net HG footptint, throughout the 7-year duration of our study, we use the CAIDA AS Organizations dataset [19]. CAIDA uses `WHOIS` information available from Regional and National Internet Registries to infer mappings of ASes to the organizational entities that operate them. AS-to-organization mappings are available every quarter from Oct. 2009 onwards, fully covering our study timeline. In order to extract the HG ASes across time, we are interested in the reverse mapping (organization-to-AS(es)), since we only know the HG's organization name. We track organization IDs (which may change across time) by parsing corresponding organization name literals, which we manually filter. We treat ASes that do not belong to the HG organization but host IP addresses that serve its associated certificates as hosts of its off-net footprint (§4).

A.3 Characteristics of HG Certificates

We present indicative interesting characteristics of HG-served certificates.

Certificate Numbers and IP Groups. The total number of certificates differs among HGs, from a few 100s (Google, 2021) to many 1000s (Facebook, 2021). The number tends to increase over time, albeit at different rates per HG. Each certificate can be served by multiple IP addresses. Figure 11 shows the coverage of the top ten IP groups (each serving the same certificate) for Google and Facebook over time. The top ten groups include over 90% of Google's certificate-serving IP addresses, with over 50% of them serving the certificate that certifies `*.googlevideo.com` among other DNS names for Google's off-net services. Facebook started with heavy aggregation in 2014 and ended up with a disaggregated pattern in 2021.

Expiration Times. The validity period for HG `TLS` certificates vary from weeks to years, varying both across HGs and across time. For example, Microsoft's certificates have a median duration of 1 year (2013-2016), between 1 and 2 years (2016-2017) or 2 years (2018-2019). Google generally uses certificates with median duration of 3 months. Netflix's median expiry times oscillate between 8 months and 2 years. However, median Netflix expiry times dropped within 2019, reaching 35 days, corresponding to its strategic shift towards short-lived certificates, first announced in 2016 [83].

A.4 Survey Questions

The survey was for the analysis of data on November 30, 2020. The questions are as follows:

1. *Overall, how do you rate the estimation of the off-net footprint of your HG?:*
   - Excellent
   - Very good
   - Good
   - Poor

2. *Do we overestimate or underestimate the off-net footprint of your HG?*
   - Overestimate
   - Underestimate
   - Estimation is quite accurate

3. *What is our estimation error of the off-net footprint?*
   - 1%
   - 5%
   - 10%
   - 20%+

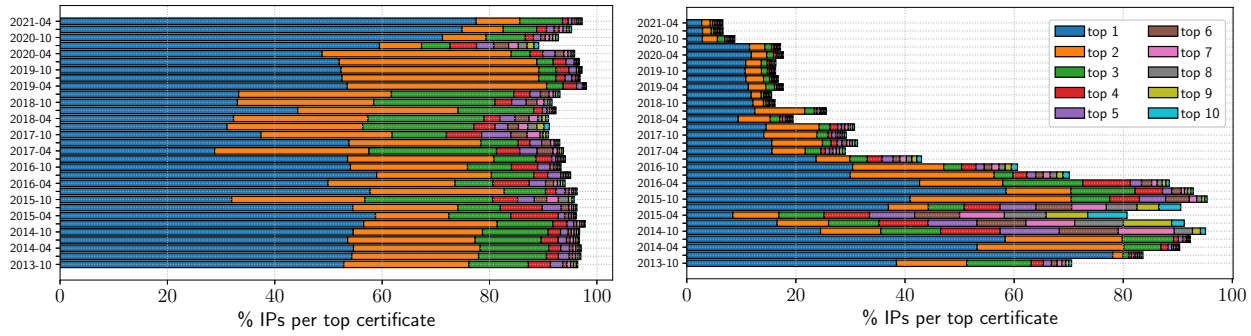
4. *Do we miss any AS when we report the off-net footprint of your HG? If yes, what type of ASes do we miss?*
   - Only a few ASes are missing
   - Datacenter ASes
   - Eyeball ASes
   - Transit ASes
   - free text – report the type(s) of ASes

A.5 List of Keywords and Headers

In Table 4 we present the keywords and headers we used in our study to identify HGs and validate the installation of their servers in off-nets (see Section 4.4). In addition to the HGs listed in Table 4, we used the following Hypergiant keywords: *Bamtech, CDN77, Cachefly, Chinacache, Disney, Highwinds*, and *Yahoo*. For the latter list of HGs, we were not able to identify unique HTTP(S) headers to extract fingerprints.

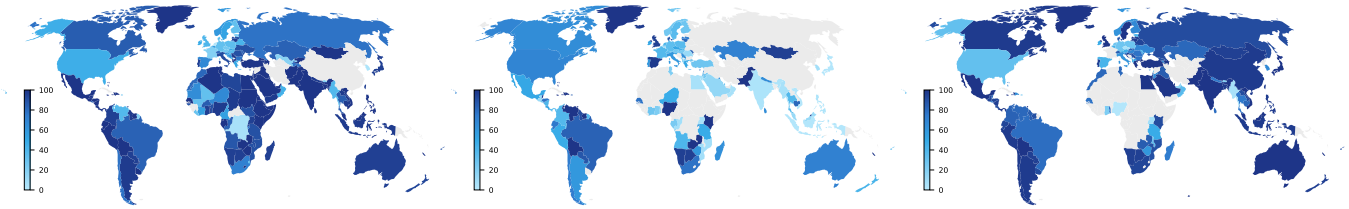
A.6 User Population Coverage based on Customer Cone

Section 6.5 examined how much of the Internet user population can be served from Google's off-nets, if they serve users within the hosting networks and their customer cones. Figure 12 plots



(a) Google (b) Facebook

Figure 11: Coverage of top-10 IP groups serving the same certificate for Google and Facebook; the coverage of a group is the percentage of associated IPs over the total IP population of the HG for the corresponding snapshot.



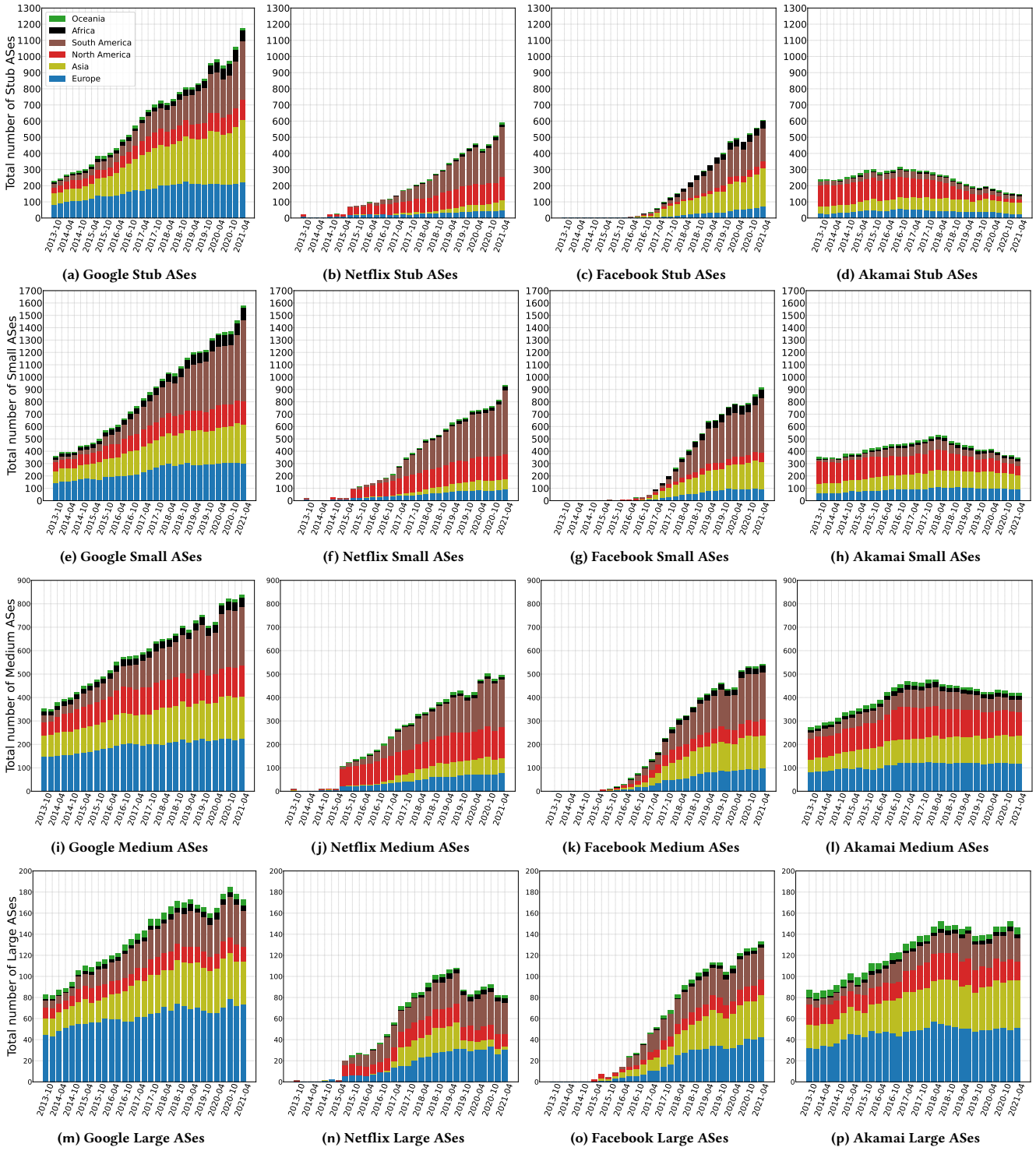
(a) Facebook (b) Netflix (c) Akamai

Figure 12: Percentage of a country's Internet users within the customer cones of ASes hosting Facebook/Netflix/Akamai off-net servers (April 2021)

Hypergiant & Keyword	Header Name:Value Pairs	Documentation
Akamai	Server:AkamaiGhost, Server:AkamaiNetStorage, Server:Ghost (only in China)	Yes [5]
Alibaba	Server:tengine*, Eagleid:, Server:AliyunOSS*	Yes [6]
Amazon	x-amz-id2:, x-amz-request-id:, Server:AmazonS3, Server:awselb*, X-Amz-Cf-Id:, X-Amz-Cf-Pop:, X-Cache:Hit from cloudfront, x-amzn-RequestId:	Yes [8]
Apple	CDNUUID:	No
Cdnetworks	Server:PWS/*	Yes [24]
Cloudflare	Server:Cloudflare, cf-cache-status:, cf-ray:, cf-request-id:	Yes [30]
Facebook	Server:proxygen*, X-FB-Debug:, X-FB-TRIP-ID:	Yes [37, 38]
Fastly	X-Served-By:cache-*	Yes [40]
Google	Server:gws, Server:gvs*, X-Google-Security-Signals: X_FW_Edge:, X_FW_Cache:	Disclosed[48, 58]
Hulu	X-Hulu-Request-Id:, X-HULU-NGINX:	No
Incapsula	X-CDN:Incapsula	No
Limelight	Server:EdgePrism*, X-LLID:	Yes [70]
Microsoft	X-MSEdge-Ref:	Yes [75]
Netflix	X-Netflix.*:, X-TCP-Info:, Access-Control-Expose-Headers:X-TCP-Info	No
Twitter	Server:tsa_a	Yes [105]
Verizon	Server:ECacc*	Yes [107]

Table 4: List of keywords and headers used to verify HGs' server installation in our study. Empty header values indicate that only the header name is used to match. Entries ending with * indicate a prefix match.

the equivalent user coverage for Facebook, Netflix, and Akamai. Serving into the customer cone (rather than just serving the hosting networks) noticeably expands Facebooks coverage in parts of Africa, Asia, Europe, and South America (Figure 12a compared to Figure 9b), expanding service from 49.9% to 63.2%, i.e, a 26.8% increase of Internet users. For Netflix, serving within customer cones slightly increases population coverage for countries in South America, North America, and Africa (Figure 12b compared to Figure 7b), that increase the user population coverage from 16.3% to 26%, i.e.,



(a) Google Stub ASes (b) Netflix Stub ASes (c) Facebook Stub ASes (d) Akamai Stub ASes

(e) Google Small ASes (f) Netflix Small ASes (g) Facebook Small ASes (h) Akamai Small ASes

(i) Google Medium ASes (j) Netflix Medium ASes (k) Facebook Medium ASes (l) Akamai Medium ASes

(m) Google Large ASes (n) Netflix Large ASes (o) Facebook Large ASes (p) Akamai Large ASes

Figure 13: Growth of the top-4 HG per continent and per network type over time.

59.4% increase. The coverage for Akamai increases significantly for countries located in Asia, Europe, and South America (Figure 12c compared to Figure 7c), resulting in a 49.1% increase, from 51.7% to 77%. The dramatic increase in Akamai's coverage when considering users within the customer cones of ASes hosting Akamai off-nets makes sense given Akamai's observed strategy of shrinkings its footprint within small ASes in favor of large ASes (§6.3).

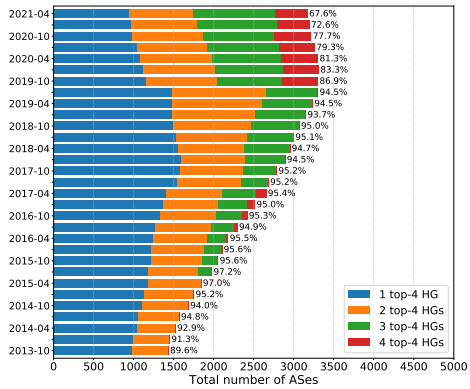
A.7 Off-Net Growth per Network Type and per Region

Combining Sections 6.3 and 6.4, in order to get insights on the growth per network type (Stub, Small, Medium, Large) in the different regions, we plot in Figure 13 the number of different network types that host off-net Google, Netflix, Facebook, and Akamai servers. Our results indicate that the expansion of HG off-nets into more stub ASes slows down in all regions and for all top-4 HGs until early 2020, at the beginning of the COVID-19 pandemic. After the summer of 2020, expansion picks up across all HGs and regions, with the exception of Akamai. The fraction of stub ASes with Akamai off-nets shrinks by around 80% in North America, but doubles in Asia, suggesting that large CDNs can flexibly rearrange their off-net footprint within a few years to better achieve their objectives (potentially choosing not to replace servers as they age out).

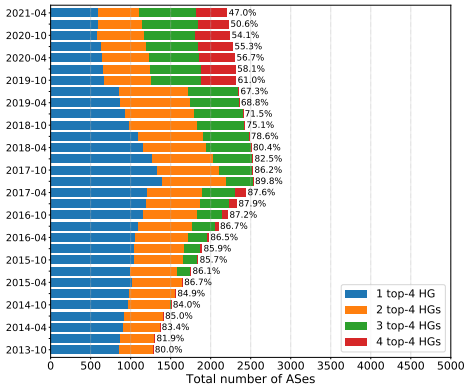
The aggressive growth of Google, Netflix, and Facebook is also visible in South America, as well as in Asia by Facebook and Akamai. Akamai's off-net footprint decreases by more than 50% in small ASes over the years. We observe similar growth for the top-4 HGs in Medium ASes as we observe in Small ASes. A noticeable exception is Akamai, which has expanded its footprint in Medium ASes in Asia and South America. Overall, our results suggest that Akamai is shifting its s off-net footprint away from Stub and Small networks towards Medium and larger networks in Asia and North America.

A.8 Willingness by Networks to Host HGs

We further investigate how the symbiosis of HGs and networks (see Section 6.6) evolves, especially for the four Hypergiants with the largest footprints. In Figures 14a and 14b we present the total number of ASes that host at least one top-4 HG in at least 25% and at least 50% of the dataset snapshots respectively, and what percentage they represent of the total ASes that host ≥1 of the top-11 HGs in at least one data snapshot (percentages as shown at the end of sub-bars). Both figures show that the majority of ASes chose to host only one top-4 HG until late 2019, where we see a shift, as more and more ASes select to host up to all the top-4 HGs. A steady rise of the number of ASes with >2 HGs takes place again until late 2019, when we notice ASes are starting to host more and more HGs outside of the big 4. This trend coincides with the beginning of the COVID-19 pandemic, as the content providers are adjusting their deployment strategies in order to meet the suddenly increased user traffic demand. In addition, even if the percentage of ASes (for the same timestamp between Fig. 14a and Fig. 14b) that hosts one to four top-4 HGs across years varies between 10% and 20%, we notice a similar trend with respect to the symbiosis of HGs. That is, until 2016, we constantly observe that more ASes are willing to host more HGs servers with a peak between 2017 and 2018. After that



(a) # ASes that host ≥1 HG in 25% of the snapshots.



(b) # ASes that host ≥1 HG in 50% of the snapshots.

Figure 14: Total number of ASes that host at least one of the top-4 HGs (Google, Netflix, Facebook, Akamai).

period, they reach a plateau in terms of the number of ASes with a slight decline in 14b. Last but not least, our analysis shows that about 5% of the total number of ASes, on average, in each snapshot, are newcomers, *i.e.,* ASes never seen in past snapshots. Overall, there is a clear symbiotic pattern due to the fact that along with newcomers there are ASes which strategically decide to host more HG servers over the years. Nevertheless, there are a few other ASes that stop appearing as HG hosts.