Leveraging Interconnections for Performance

The Serving Infrastructure of a Large CDN

Florian Wohlfart[*][§] Nikolaos Chatzis[§] Caglar Dabanoglu[§] Georg Carle[*] Walter Willinger[‡]

[§]Akamai Technologies [*]Technical University of Munich [‡]NIKSUN, Inc.

ABSTRACT

Today's large content providers (CP) are busy building out their service infrastructures or "peering edges" to satisfy the insatiable demand for content created by an ever-expanding Internet edge. One component of these serving infrastructures that features prominently in this build-out is their connectivity fabric; i.e., the set of all Internet interconnections that content has to traverse en route from the CP's various "deployments" or "serving sites" to end users. However, these connectivity fabrics have received little attention in the past and remain largely ill-understood.

In this paper, we describe the results of an in-depth study of the connectivity fabric of Akamai. Our study reveals that Akamai's connectivity fabric consists of some 6,100 different "explicit" peerings (i.e., Akamai is one of the two involved peers) and about 28,500 different "implicit" peerings (i.e., Akamai is neither of the two peers). Our work contributes to a better understanding of real-world serving infrastructures by providing an original account of implicit peerings and demonstrating the performance benefits that Akamai can reap from leveraging its rich connectivity fabric for serving its customers' content to end users.

CCS CONCEPTS

• **Networks → Network architectures**;

KEYWORDS

Content Providers, Content Delivery Networks, Peering

ACM Reference Format:

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 '18: SIGCOMM 2018, August 20–25, 2018, Budapest, Hungary*. ACM, New York, NY, USA, 15 pages. https://doi.org/10.1145/3230543.3230576

SIGCOMM '18, August 20–25, 2018, Budapest, Hungary

ACM ISBN 978-1-4503-5567-4/18/08…$15.00
https://doi.org/10.1145/3230543.3230576

1 INTRODUCTION

Today's large Internet content providers (CP) that include the large content delivery networks (CDN) are faced with the problem of having to serve ever-increasing traffic volumes to a growing number of increasingly heterogeneous end points (e.g., end users, IoT devices) that reside in different types of networks, consume diverse types of content and require ever more stringent performance guarantees. When trying to solve this challenging problem, the *serving infrastructures* that these large CPs maintain take center stage. Here, a CP's serving infrastructure (also referred to as its "(Internet) peering edge" or "peering surface") consists of two main components. The first is its *footprint*; that is, a set of "deployments", where a deployment comprises of one or more clusters of servers. By this definition, deployments may or may not contain servers that are directly involved in serving content to end users. While our focus will be mainly on deployments with such end user-facing server clusters (known as "edge nodes" or "serving sites"), the footprint of a large CP's serving infrastructure typically also contains other types of deployments, and in this paper, we will specify the deployment type unless it is obvious from the context. The second component is its *connectivity fabric*; that is, the set of Internet interconnections or peerings that the content served by this CP has to traverse as it travels from the CP's deployments where it is ingested or resides to the end users where it was requested. Aspects of these large CPs' serving infrastructures that are of particular interest are the extent and structure of their footprints, the details of their connectivity fabrics, and their ability to scale in a cost-effective manner as traffic volumes keep increasing, performance considerations gain in importance, and the required capabilities (e.g., measurements of the network's and infrastructure components' state and performance, directing end user clients to servers) become more demanding.

The serving infrastructures that some of today's large CPs have built and rely on differ in size, design, and operations. For example, in a recent paper [47], Google states that it has "one of the largest peering surfaces in the world" and describes the structure of this "surface" as consisting of a set of edge nodes (i.e., Google-supplied servers known as Google Global Cache (GGC) that are deployed in third-party networks) and a set of interconnected edge PoPs in some

70 metro areas where Google connects to the rest of the world via peering [23]. Moreover, in support of this structure, Google operates a private inter-data center backbone and a separate WAN that connects to external peers and back to its large data centers [47]. Facebook describes its serving infrastructure as consisting of "dozens of PoPs in six continents" where it has "thousands of peers and serves over two billion users" [42].[1] While there have been a number of recent papers that describe various aspects of these and other CPs' serving infrastructures (e.g., see [7, 9, 42, 47]), the focus has been almost exclusively on their footprints and on traffic engineering-related challenges posed by the sheer scale of these infrastructures. At the same time, the connectivity fabrics that these CPs utilize to get content from the various deployments all the way to the end users have received little or no attention and remain largely ill-understood.

In this paper, we provide a detailed account of the serving infrastructure of Akamai, especially its connectivity fabric. Akamai is a large global CDN whose serving infrastructure's footprint consists of a large number of deployments of different types (including deployments in third-party networks). Akamai also operates its own multi-service backbone to support the delivery of its customers' content to end users. We note, however, that despite detailing the serving infrastructure of Akamai, our study is not about which type of serving infrastructure (e.g., one with or without deployments in third-party networks, one with or without a backbone) is better or worse. Instead, our work uses Akamai's existing serving infrastructure as an example to highlight the important but subtle aspects that need to be considered when examining this increasingly important part of a large CP's infrastructure, especially with respect to determining and establishing the exact extent and structure of its connectivity fabric.

To this end, we first report in Section 3 on an in-depth study of the connectivity fabric component of Akamai's serving infrastructure. Our study reveals a bifurcation of all of the interconnections utilized by Akamai into (i) a set of 6.1k "explicit" peerings (e.g., traditional peering options where Akamai is one of the two involved peers) and (ii) a set of 28.5k "implicit" peerings (i.e., traditional peering choices where neither of the involved peers is Akamai). We elaborate on what information sources are required to fully and conclusively account for such a rich connectivity fabric and discuss why relying on publicly available BGP information (and possibly other information obtained from additional active measurement campaigns) provides only an inadequate picture of this densely connected fabric. In the process, we explain why these implicit peerings have gone largely unnoticed in the past (see, however, [13] for an earlier account of implicit peerings at a large IXP) and have prevented researchers from appreciating the full extent of the serving infrastructures of large CPs such as Akamai. In addition, we show that the contributions of Akamai's different deployment types to its sizable connectivity fabric are uneven, with some deployments contributing only explicit peerings and others only implicit peerings, and we provide illustrative examples to explain this observation. Next, in view of Akamai's objective to optimize the performance of content delivery as experienced by the end users, we quantify in Section 4 some performance benefits that Akamai reaps from leveraging its rich connectivity fabric when choosing from among the different options it has for serving its content end users.

At first glance, determining the connectivity fabric of a network A boils down to identifying the number and type of direct peerings that A utilizes to connect to other networks within the larger Internet, together with the equipment that is located in the different deployments and is required to establish and operate those peerings. These peerings can be of the following well-known types and are "explicit" in the sense that network A is always one of the two parties to such a peering: transit (dedicated PNI), private peering (via dedicated PNI), public peering in the form of bilateral peering via an IXP, and public peering in the form of multilateral peering via an IXP's route server. However, this simple picture of network A's connectivity fabric gets significantly more complex when A is a large CP that operates, for example, deployments in a third-party network B that happens to have a number of eyeball networks as downstream customers. Note that content destined from any of A's deployments in network B to any of the end users in any of B's downstream customers necessarily contributes to interdomain traffic; that is, such content has to traverse existing explicit peerings between network B that houses A's deployments and B's downstream customers where the end users reside. In effect, as a result of operating deployments in B's network, the large CP A "inherits" B's explicit peerings with its downstream customers and can leverage them to serve its customers' content to those networks' end users. Since none of the two peers involved in such an inherited peering is the large CP A, we refer to them in this paper, as "implicit" peerings to distinguish them from the above-defined explicit peerings.

Note that in the above example, the large CP A also inherits from B any of B's explicit peerings with its upstream providers but the "terms-of-use" for these types of inherited or implicit peerings are typically more restrictive (e.g., use for cache fill is allowed; use of transit to serve other networks' end users is restricted or not allowed) than those associated with B's downstream-related implicit peerings. In this paper, we are mainly concerned with downstream-related

[1] Searching various online sources produces other relevant details about Facebook's serving infrastructure, including (i) the deployment of Facebook-owned and supplied servers known as Facebook Network Appliances (FNA) in third-party networks [6] and (ii) the operation of a newly-deployed private inter-data center backbone network called Express Backbone [20].

implicit peerings. However, to provide a complete picture of the full connectivity fabrics of large CPs such as Akamai, we will include the upstream-related implicit peerings and state their number separately. Irrespective of the type of implicit peering, from a practical perspective, the main difference between explicit and implicit peerings is that the latter give the large CP A no say in either establishing or operating them. In fact, while A is by and large in charge of the traffic that traverses any of its explicit peerings, A is just one of the many contributors to the traffic carried by implicit peerings. Nevertheless, since implicit peerings are used by A to carry its customers' content, we view them as being as critical a part of A's connectivity fabric as A's explicit peerings, especially because of the mutual benefits that the large CP A and network B derive from them. For example, in the case of B's downstream-related implicit peerings, their use by A reduces transit costs for B and the cost for serving content for A. At the same time, by getting content closest to eyeballs, the use of these implicit peerings improves the performance of content delivery as experienced by the end users (i.e., a metric by which B and its downstreams get evaluated by end users and A gets evaluated by its customers).

2 OVERVIEW OF AKAMAI'S SERVING INFRASTRUCTURE

In this section, we describe Akamai's serving infrastructure in terms of its footprint, focusing in particular on the diverse nature of the deployed servers and how Akamai's servers are organized into clusters and ultimately into deployments, and present some typical types of deployments.

2.1 Basic Components: Footprint

As a general rule, Akamai's servers can be grouped into end-user facing (EUF) delivery servers, non-end-user facing (non-EUF) delivery servers, and non-delivery servers. The first group consists of HTTP and/or HTTPS (referred to as HTTP/S throughout this paper) servers that are directly involved in serving content to end users and other delivery servers. In contrast, their non-EUF counterparts are HTTP/S servers that participate only indirectly in the delivery of content to end users and other servers (e.g., serving content to other servers such as storage servers or performing functions such as transcoding media content). While these non-EUF delivery servers may not be required for some scenarios they are essential for the delivery of the type of content that relies on them. Finally, Akamai also operates non-delivery servers, and as the name indicates, the servers belonging to this group do not play any role in the delivery of content but are used for other purposes. An important example of such servers are Akamai's BGP collectors (see below). Our interest in this paper is in the combination of Akamai's EUF delivery servers and Akamai's BGP collectors.

Akamai's EUF delivery servers run Akamai's own software stack on custom-built hardware. These servers can be flexibly configured to serve many different purposes and have various capabilities. Noteworthy and distinctive server capabilities include aspects related to delivery, performance, and caching. The same software is used for servers that serve different workloads (e.g., cacheable, non-cacheable content), different customer needs (e.g., origin offload), or different traffic types (e.g., latency-sensitive traffic). What differs is which servers and what capabilities of those servers are used with respect to delivery, performance, or caching and how they are organized (e.g., flat, hierarchically) in each use case. To communicate, delivery servers talk HTTP/S to other Akamai servers and customers' origin servers as well as to clients running on end user devices (e.g., browsers, customers' download clients or players, set-top boxes). Akamai's EUF delivery servers are organized into clusters that are located in a total of about 3.3k deployments within more than 1,600 networks around the world.

2.2 Typical Deployment Types

Although the 3.3k deployments that contain EUF delivery servers vary in size, design, role, and capabilities, Table 1 summarizes the characteristics of four generic deployment types and provides the main differences and similarities in how they are architected and used. The listed Type 1, Type 2, Type 3, and Type 4 deployments are representative of Akamai's present-day serving infrastructure and account for more than 85% of all deployments. We will describe aspects of their external connectivity in Section 2.3.1 and mention further relevant details about these four deployment types when describing and explaining some of the main results of our empirical analysis in Section 3 and Section 4.

For the purpose of this paper, it is important to note that by virtue of operating an Akamai-owned border router (on Akamai's peering AS) that participates in BGP, Type 3 and Type 4 deployments contribute *only explicit but no implicit peerings* to Akamai's connectivity fabric. In contrast, the absence of an Akamai-owned router and reliance on a hosting network-owned border router for Type 1 and Type 2 deployments implies that they contribute *only implicit but no explicit peerings* to Akamai's connectivity fabric.

2.3 An Infrastructure for Measurement

Irrespective of whether or not they contain EUF delivery servers, Akamai's deployments also play an important role as a set of vantage points of a global-scale measurement platform that this CDN has developed over time and keeps changing and improving in the face of ever-changing needs

Table 1: Characteristics of Akamai's typical deployments with least one EUF delivery server group.

	Deployments without an Akamai router		Deployments with an Akamai-owned router	
	Type 1	Type 2	Type 3	Type 4
Typical size	Small/Medium	Medium/Large	Large	Large
IP address space used	Host network	Host network	Akamai	Akamai
AS used	Host network	Host network	Akamai	Akamai
Akamai transit link	No	Yes	Yes	Yes
Use of transit link	Cache fill	Cache fill/Serve	Mainly cache fill	Mainly cache fill
Can serve all end users	No	Yes	Yes	Yes
Target end users/networks	Host/Downstreams	All	Small/Medium	Medium/Large
Proximity to end users	High/Very High	Low/Medium	Medium/High	High
Example of typical setting	Eyeball network	Transit network	IXP	PNIs w. eyeball networks
Example of atypical setting	Wholesale network	N/A	N/A	PNIs w. backbone providers

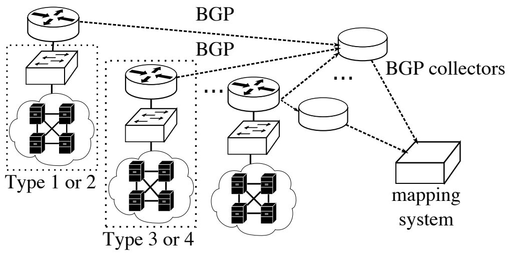


Figure 1: Akamai and non-Akamai routers send BGP information over the Internet to the BGP collectors, which in turn send it to Akamai's mapping system.

and requirements. As resources-rich vantage points, Akamai's server clusters also act as "measurers" that produce a range of measurement data. While most of the server clusters specialize in producing data that provides important information about the overall state of Akamai's serving infrastructure, others are exclusively focused on collecting BGP information that is pertinent for maintaining an up-to-date view of Akamai's connectivity fabric and central to its role of delivering content from where it is ingested or resides to where it is consumed in a performance-optimal manner. Since the information collected by the different server clusters about network health, performance, and routing can (and does) change over time as a result of the dynamic nature of the Internet in general and its routing system in particular, the measurements are typically obtained at regular time intervals (e.g., quarter-hourly, hourly, daily).

To illustrate, Akamai's server clusters maintain an overall view of the state of Akamai's infrastructure by keeping track of and reporting, among other quantities, their cluster-specific bandwidth usage and system load. They also actively measure network conditions (e.g., loss, latency) and connectivity (e.g., route) between them across different deployments – criss-cross measurements leveraging server clusters as vantage points. In addition to measuring network conditions and connectivity to other server clusters, these measurers also actively measure network conditions and connectivity to popular name server/resolvers and other targets, in part for evaluating their own capabilities to communicate with those targets. In effect, the measurements to these targets let the individual server clusters assess their capabilities to communicate with end user clients or other servers that are in the same network segments as those targets.

2.3.1 Non-Delivery Servers as BGP Collectors. Akamai operates 80 BGP collectors across the globe. Akamai's BGP collectors are non-delivery servers. They receive routing information from Akamai and non-Akamai routers inside the various deployments over the Internet and provide that information as input to Akamai's mapping system (see Figure 1 for a high-level overview of the BGP information flow) in a form that the mapping system can process and consume. For instance, an important capability of Akamai's BGP collectors is to translate third-party network-specific BGP communities into a common set of Akamai-specific BGP communities. Because of the critical role they play as components of Akamai's measurement platform and as producers of input data for Akamai's mapping system (see Section 2.3.2), deployments that host BGP collectors have to be always available and conform to specific rules.

The routing information received by these BGP collectors from Akamai's various deployments is deployment dependent. In particular, it is the deployment type that determines what BGP information is obtained via what BGP sessions with which networks. For example, for Type 1 and Type 2 deployments that do not operate an Akamai-owned router and instead rely on a router in the hosting network, that hosting network's router has BGP sessions to nearby (in terms of network distance) BGP collectors and sends them the BGP information that the hosting network provides, including routes for end user and name server/resolver prefixes and BGP communities. Here, the prefixes can be the hosting network's own prefixes, the prefixes of its downstream customers and even prefixes of networks for which there is a special agreement for Akamai’s traffic. BGP communities are particularly informative because by setting BGP communities, the hosting network can signal to Akamai how to serve the network’s own prefixes and the prefixes of its downstream (or other) networks.

For a Type 3 deployment, the BGP information originates in part from the BGP sessions that the Akamai-owned router has with each IXP member that peers bilaterally with Akamai and in part from the BGP sessions that that router may have with the IXP’s route server(s). The deployment’s Akamai-owned router also has BGP sessions to more than one BGP collector for redundancy. However, in the case of a Type 3 deployment, it is the routing table and not the router’s BGP table (i.e., all the BGP information it obtains over time) that is sent to the BGP collectors. The routing table contains only the “best” (also referred to as “active”) routes from all the routes received from the various direct (i.e., bilateral) or indirect (i.e., route server) sessions with the IXP’s peers. Moreover, information that is not sent includes routes that are filtered at the router-level and routes that are received from the deployment’s transit link. In view of what information Akamai’s BGP collectors receive, Type 3 deployments are similar to Type 4 deployments. In the latter case, the Akamai-owned router has a BGP session for each PNI and sends BGP information about the best paths it receives from the various PNIs to Akamai’s BGP collectors. Thus, in case the Akamai-owned router has a PNI with an eyeball network that also hosts one or more Type 1 deployments, the BGP information that the BGP collectors receive from the PNI may not be as fine-grained as the information it obtains from the Type 1 deployment(s).

2.3.2 Measurement Platform vs. Mapping System. A key element of Akamai’s service delivery platform is its mapping system. At its core, Akamai’s mapping system relies on DNS to route each end user client to a deployment with at least one EUF delivery server that ultimately serves the requested content. As a DNS-based system, this mapping system has evolved over time, and we refer the interested reader to [14] for more details. However, what matters for the purpose of this paper is that Akamai’s mapping system is a consumer of a myriad of data (including the data from Akamai’s BGP collectors) that originates from Akamai’s global-scale measurement platform.

Akamai’s mapping system ingests a large number of different measurement data and combines them to ultimately return for each end user request a rank-ordered list of deployments with EUF delivery servers (and corresponding IP addresses). In particular, even though the mapping system takes BGP collector data as one of its many inputs, it is not a system that makes decisions solely based on BGP. At the same time, the mapping system is also a consumer on non-measurement data (i.e., data that is neither produced nor collected by Akamai’s measurement platform). Examples of such non-measurement data include cost-related information about peering links and detailed topological information about a hosting eyeball network. In short, to achieve Akamai’s main objective of optimizing the performance of service delivery, its mapping system relies on inputs of various kinds (e.g., measurement and non-measurement data), is highly flexible as there are numerous ways to tune it to affect traffic flow to overcome issues or meet special needs, and is constantly evolving as new functionalities get added to satisfy an ever more diverse customer base, an increasing selection of service offerings, and rapid innovations in key Internet technologies.

3 AKAMAI’S CONNECTIVITY FABRIC

After describing the available datasets, we provide a detailed assessment of the reach and structure of Akamai’s connectivity fabric and pay particular attention to what type of BGP information sheds light on what aspects of this fabric.

3.1 Available Datasets

3.1.1 Proprietary BGP information – ViewA. For our study, we obtain the portion of the Internet control-plane information that is used by Akamai as one of the inputs to its mapping system. To this end, we rely on Akamai’s BGP collectors that dump on an hourly basis the information in their BGP tables in MRT format. Our dataset consists of the BGP table dumps from all the BGP collectors and will be referred to in the following as ViewA. The data contains both IPv4 and IPv6 information. We analyzed six hourly snapshots of ViewA, and Table 2 lists a few key metrics for the six snapshots and shows how these metrics vary over an eight-month period.

Note that the type of hourly snapshots of ViewA shown in Table 2 represent exactly the BGP information that Akamai used as routing data input for its mapping system at those particular times. Since our work in this paper concerns the hourly inputs as obtained by Akamai’s mapping system and not their evolution over time, and given the rather modest variations in the metrics shown in Table 2, due to limited space, in the rest of this section, we report on results that concern our analysis of the 2017-09-17 snapshot of ViewA. This snapshot was collected from some 4.5k BGP sessions between Akamai’s BGP collectors and non-Akamai and Akamai routers and consists of more than 3.65M AS paths and about 1.85M IPv4 and IPv6 prefixes. However, as an illustration that our reported results are consistent with and representative of those obtained for the other snapshots listed in Table 2, we discuss in Section 4 additional results for our most recent 2018-05-17 snapshot.

Table 2: Overview of Akamai's BGP data – ViewA.

	Sep 17 2017	Oct 10 2017	Nov 10 2017	Dec 5 2017	Jan 10 2018	May 17 2018
ASes	61.3k	60.2k	59.9k	60.4k	63.2k	61.5k
Prefixes	1.85M	1.80M	1.83M	1.85M	1.92M	1.88M
Paths	3.65M	3.53M	3.53M	3.49M	3.56M	3.36M

3.1.2 Publicly available BGP information – ViewP. To put ViewA in perspective, we use and combine data from three well-known public data sources; i.e., Route-Views (RV) [32], RIPE NCC RIS (RIPE) [31], and the daily routing snapshots collected by Packet Clearing House (PCH) [24]. We obtain the BGP table data from all the available collectors from these data sources for 2017-09-17, the same day as the data for our 2017-09-17 snapshot of ViewA data was obtained. The RV and PCH data contains IPv4 and IPv6 information whereas the RIPE data provides only IPv4 information. The RV and RIPE data is stored in MRT format whereas the PCH data is available as compressed text files that contain the output of running the *show ip bgp* command on the PCH route collectors. We combined the three public datasets into one dataset and refer to it below as ViewP. Table 3 provides details about each of the three public datasets for 2017-09-17 and also about their combined ViewP which consists of more than 21M different AS paths.

3.1.3 ViewA *vs.* ViewP*.* Note that for the purpose of our study, there is no need for examining multiple snapshots of ViewP. In fact, in contrast to prior studies that typically required multiple snapshots of BGP data, such as ViewP, to examine questions about the observed AS-level Internet such as its structure, completeness, or its evolution over time (e.g., see [12, 17, 34] and references therein), our use of ViewP is rather restrictive; that is, we use ViewP mainly for comparing inherent properties of ViewP-like datasets against their counterparts in data such as ViewA. For example, one such property is that ViewP consists of BGP table data while ViewA is based on routing table data that Akamai's BGP collectors receive from their BGP sessions with non-Akamai and Akamai routers (see Section 2.3.1). This means that for a given prefix, ViewP typically has information about many different AS paths to that prefix while ViewA (i.e., BGP tables from Akamai's BGP collectors) only provides one path per deployment – the best path. Another distinguishing qualitative feature concerns what routes are filtered and thus are not part of ViewP and ViewA, respectively. ViewA does not include routes that are received from the links of Akamai with its transit providers, but ViewP will in general include those transit provider's BGP data. We discuss implications of these and other such features in this section.

Table 3: Overview of the public BGP data – ViewP.

Dataset	Collectors	ASes	Prefixes	Paths
RV	19	58.4k	872k	12.5M
RIPE	18	57.9k	737k	11.4M
PCH	140	58.0k	733k	0.4M
ViewP	177	58.6k	900k	21.1M

3.2 On the Reach of Akamai

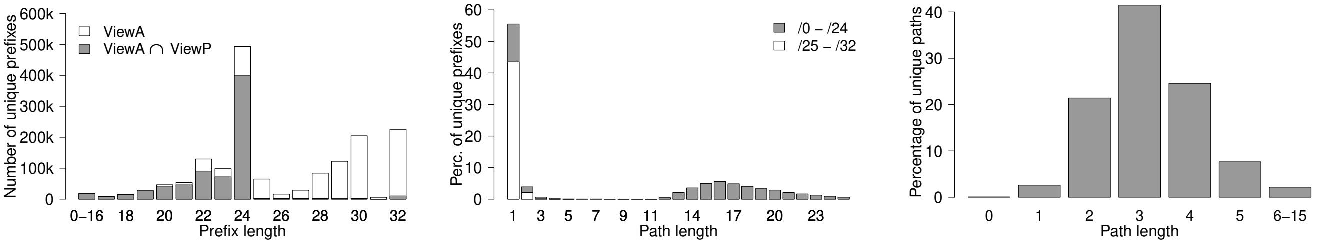
3.2.1 Serving the World: ASes and Prefixes. We first perform an AS-level analysis of ViewA and find a total of 61.3k unique routeable ASes for Akamai; that is, Akamai sees at least one originating prefix from 61.3k ASes. This compares to 58.6k unique routeable ASes seen in ViewP. For comparison, Hurricane Electric, a large backbone provider, reports seeing some 60.7k routeable ASes [18], an indication that global-scale providers such as Akamai and Hurricane Electric tend to see more routeable ASes than ViewP.

Next, a closer look at the originating prefixes reveals a total of 1.75M unique IPv4 prefixes in ViewA and only 0.85M in ViewP. Hurricane Electric reports a very similar result for the number of prefixes it observes (see [18]). Table 4 (left half) provides details about the unique IPv4 and IPv6 prefixes that are only seen in ViewA, only in ViewP, and in both ViewA and ViewP, respectively. We observe, for example, that the number of prefixes that are only present in ViewP (i.e., ViewP \ ViewA) is comparatively small – only 99.5k IPv4 prefixes are exclusively seen in ViewP. To explain why Akamai receives almost twice as many prefixes as ViewP, Figure 2a shows a breakdown by prefix length of the number of unique IPv4 prefixes in ViewA and their overlap with those in ViewP. The plot shows that of the 1M unique prefixes that only Akamai receives (i.e., ViewA \ ViewP), around 75% of them are of length /25 or longer.

When analyzing this data further, we notice that almost all the IPv4 prefixes that are present in ViewA but absent from ViewP originate from ASes that are seen in ViewP. Thus, although at the AS-level, ViewA and ViewP are similar, at the level of originating IPv4 prefixes, Akamai receives information in a much more fine-grained manner compared to what can be discerned from ViewP.[2] The fact that of the almost 1M unique IPv4 prefixes that are only seen in ViewA, 75% are of length /25 or longer suggests that they play a key role for Akamai's content delivery service (see below for details), and this observation will be qualitatively the same for different instantiations of ViewA and ViewP.

3.2.2 Serving the World: AS Paths. To demonstrate the above-mentioned qualitative differences between ViewA and

[2] A well known practice by network operators is to filter prefixes longer than /24 to limit the growth of their Internet routing tables. Note however that ViewP still sees some 60k IPv4 prefixes of length /25 or longer.



(a) Unique prefixes by pref. length. **(b) Unique prefixes by path length.** **(c) Unique paths by path length.**

Figure 2: IPv4 prefixes and paths in ViewA.

Table 4: Num. of unique prefixes and paths in ViewA.

	Prefixes			Paths		
Dataset	IPv4	IPv6	Both	IPv4	IPv6	Both
ViewA	1.75M	97k	1.85M	3.0M	863k	3.7M
ViewP	0.85M	48k	0.90M	20.8M	526k	21.1M
ViewA ∩ ViewP	0.75M	47k	0.80M	1.1M	165k	1.2M
ViewA \ ViewP	1.00M	50k	1.05M	1.9M	698k	2.5M
ViewP \ ViewA	0.10M	1k	0.10M	19.7M	428k	19.9M

ViewP, we find 21.1M unique AS paths in ViewP to reach the 0.9M observed prefixes seen in that dataset, but for the 1.85M encountered unique originating prefixes in ViewA, we only observe 3.7M unique AS paths in ViewA.

Zooming in on these observed 3.7M unique AS paths seen in ViewA, we find that 2.5M or 68% of them are not present in ViewP. Moreover, as summarized in Table 4 (right half), the overlap in observed AS paths between ViewA and ViewP is small and uneven – the 1.1M AS paths seen in both datasets make up 30% of the AS paths seen in ViewA, but only 5% of the AS paths seen in ViewP. Together, these results show that not only does Akamai’s mapping system ingest less than 15% of the unique AS paths encountered in ViewP but that given the relative stability of the number of AS paths in ViewA over time, a significant number of the AS paths in ViewP are largely irrelevant for the operation of Akamai.

Next, Figure 2b shows a barplot of the number of unique IPv4 prefixes for which there exist exactly k AS paths ($k \leq 30$) from which Akamai can choose from. Here, each bar shows the number of prefixes separate for short (i.e., /0 -/24) and long (/25 - /32) IPv4 prefixes. We observe that about 55% of the IPv4 prefixes seen in ViewA have only a single AS path (not shown here: this number jumps to 93% if we consider instead the IPv4 prefixes seen exclusively in ViewA) and that prefixes of length /25 or longer are typically only reachable via a single AS path. To contrast, for ViewP, we find 10 or more different paths for 81% of its IPv4 prefixes.

Figure 2c depicts the distribution of AS path lengths for ViewA and shows a median path length of three AS hops and a maximum path length of 15 AS hops (not shown here: ViewP results in a similar plot, though with a maximum path length of 22 AS hops).[3] Note that the bar at path length zero corresponds to the cases where Akamai’s deployment is in the same AS as the prefix associated with the path. As expected, in most such cases, Akamai’s deployments know about only a single path to those prefixes.

3.3 Akamai’s Connectivity Fabric

Our focus in this section is on identifying the set of Internet interconnections or peerings that Akamai uses to deliver content from its EUF delivery clusters to the prefixes that requested that content and were observed in ViewA.

3.3.1 Explicit Peerings. As defined in Section 1, the “explicit” peerings of a network A can be identified by parsing the AS path information that is maintained, collected and transmitted by BGP. Also recall that explicit peerings involving Akamai can only be established in its Type 3 and Type 4 deployments where Akamai operates its own border routers that participate in BGP and represent the Akamai-side of any of its explicit peerings.

We search ViewA for the next hop ASNs of each Type 3 and Type 4 deployment and count the number of unique next hop ASNs of Akamai. Note that this count represents the ground truth for the set of Akamai’s explicit peerings because the Akamai-owned routers supply Akamai’s BGP collectors with the best routes they receive from the BGP sessions they have with all their peers. In total, we find 6,111 such unique neighbor ASes or explicit peerings for Akamai (some 10% of all routeable ASes seen in ViewA). More than 6,000 of the neighbor ASes are learned from IXPs (i.e., from Type 3 deployments) and the remaining 200 or so from Type 4 deployments; see Table 5 (left half) for a breakdown by deployment type and IP version.

Knowing the total number of explicit peerings does not provide the full picture of this set of interconnections. To provide more details, Figure 3 (right half) uses a box plot to show the number of explicit peerings per deployment for Type 3 and Type 4 deployments, respectively. Analyzing those explicit peerings even further, we find that the number of peering locations per explicit peering is highly skewed:

[3] We observe that the longest AS paths are inflated by routing loops.

Table 5: Number of peerings in ViewA.

Deployment	Explicit peerings			Implicit peerings		
Type	IPv4	IPv6	Both	IPv4	IPv6	Both
Type 1	-	-	-	26,216	3,965	26,429
Type 2	-	-	-	7,275	2,127	7,322
Type 3	6,013	2,746	6,075	-	-	-
Type 4	204	185	227	-	-	-
Total	6,050	2,794	6,111	28,152	5,309	28,353

out of all the 6,111 explicit peerings, Akamai sees 50 of them at 25 or more deployments, 290 at 10 or more deployments, and 859 of them at 5 or more deployments.

To illustrate how the established ground truth for Akamai’s set of 6,111 explicit peerings stacks up against what is typically visible in publicly available BGP data, we examine ViewP for explicit peerings that involve Akamai. We search ViewP’s AS paths, count the number of unique ASNs preceding or following Akamai’s peering ASN, and find a total of 450 such ASNs or explicit peerings. On the one hand, seeing Akamai’s peering AS in ViewP is fully expected since as an active participant in BGP, Akamai must advertise some of its own prefixes to receive traffic, for example in the form of content from its customers. At the same time, encountering only such a small number of explicit peerings is a reminder that based on control-plane information alone, it is impossible for a third-party observer to see another network’s bilateral (explicit) peerings in a given location [40]. Even if two networks peer multilaterally at the same location, they may not receive the same BGP information.

3.3.2 Implicit Peerings. As defined in Section 1, Akamai’s “implicit” peerings can neither be identified nor associated with Akamai by parsing the AS path information available in collected BGP data. In particular, since the presence of Akamai inside a hosting network is hidden from BGP, its implicit peerings cannot be studied using ViewP. Also recall (see Table 5) that while Akamai’s Type 1 and Type 2 deployments do not contribute to Akamai’s explicit peerings, as integrated parts of a given hosting network, they are pertinent for determining the implicit peerings that Akamai “inherits” from this hosting network. Specifically, Type 1 deployments can serve the host AS as well as the host’s downstream customers (assuming the downstream customers permit it) and Type 2 deployments can serve the hosting (transit) network and its downstreams. Finally, for reasons also mentioned in Section 1, we further divide Akamai’s implicit peerings into downstream- and upstream-related implicit peerings and quantify in the following their contributions to Akamai’s connectivity fabric separately.

First, checking ViewA for downstream-related implicit peerings, we find that Akamai utilizes a total of 28,353 unique such interconnections; that is, almost half of all routeable ASes seen in ViewA are within one AS-hop from Akamai. Table 5 (right half) gives a breakdown of the observed implicit peerings by deployment type and IP version and shows an uneven distribution, with almost four times as many implicit peerings for Type 1 deployments (26,429) compared to Type 2 deployments (7,322). Note, however, that Type 1 deployments are hosted in orders of magnitude more different host ASes compared to Type 2 deployments. Figure 3 (left half) shows that the median number of implicit peerings that Akamai inherits from Type 1 deployments is 10; for Type 2 deployments, the median is close to 1,000, which is consistent with the fact that transit providers (Internet core, Type 2 deployments) are typically well-connected while eyeball ISPs (Internet edge, Type 1 deployments) only forward (a subset of) their downstreams to Akamai.

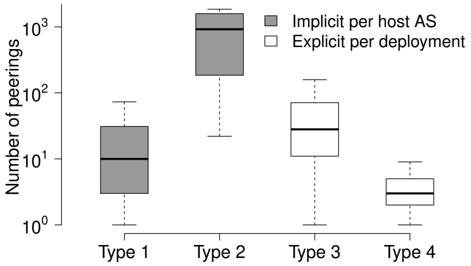


Figure 3: Implicit peerings by hosting AS (Type 1 and 2) and explicit peerings by deployment (Type 3 and 4).

Next, to show that the number of Akamai’s upstream-related implicit peerings pales in comparison to the observed 28,353 downstream-related implicit peerings, we note that in general, ViewA does not provide the information needed to obtain the precise upstream connectivity of those networks that host Type 1 deployments.[4] Instead, we leverage a combination of ViewA information (e.g., the Type 1 deployment’s hosting ASes) and ViewP AS path information (e.g., hosting ASes’ upstream providers) and infer a total of 1,506 unique upstream-related implicit peerings that Akamai can utilize to get traffic in or out of its Type 1 and Type 2 deployments (i.e., incoming traffic resulting from cache fill requests for Type 1 and Type 2 deployments and outgoing traffic for serving content for Type 2 deployments).

3.4 Illustrative Examples

3.4.1 Routerless deployments – Types 1 and 2. A Type 1 deployment provides Akamai with control-plane information about the hosting network. In fact, it is generally in the interest of the operators of that network to share with Akamai detailed information about the prefixes of its end

[4] For Type 2 deployments, ViewA provides ground truth with respect to Akamai’s upstream-related implicit peerings; similar arguments as in the case of Akamai’s explicit peerings for Type 3 or Type 4 deployments apply.

users and the corresponding name servers/resolvers and to set and share BGP communities to tell Akamai how to serve those prefixes. Consider an actual example of a large eyeball network that has no downstream customers and hosts 18 different Type 1 deployments. We observe that more than 99% of the prefixes that this network shares with Akamai are /32 prefixes tagged with either of two different BGP community attributes. The few remaining prefixes – including aggregations of /32 prefixes – can be found in ViewP. The rationale for this eyeball network to share such fine-grained information with Akamai is twofold. For one, this network leverages end-user mapping. Moreover, by means of the BGP communities, it signals Akamai a preference over which deployment should serve which end users. While the lack of downstream customers results in no downstream-related implicit peerings from these 18 deployments for Akamai, examining this network's upstream connectivity, we find more than 45 upstream-related implicit peerings (that are leveraged for cache fill but not for serving content to end users on other eyeball networks).

The operators of the networks that host Type 2 deployments typically do not provide any private information but tend to share with Akamai information that they also provide to other networks/customers. However, Type 2 deployments located in selected networks can contribute a large number of unique downstream-related implicit peerings. For example, when examining an actual Type 2 deployment on the network of a large global backbone provider, we find that it sends Akamai more than 668k different IPv4 prefixes and more than 43k different IPv6 prefixes (i.e., the complete routing table [1]). Those prefixes are served through more than 1,500 different ASes, resulting in more than 1,500 downstream-related implicit peerings for Akamai. At the same time, as a Tier 1 network, this hosting AS contributes no upstream-related implicit peerings.

3.4.2 Deployments with a router – Types 3 and 4. On the one hand, Type 3 deployments are the main contributors to the number of Akamai's explicit peerings. For example, a single Type 3 deployment at one of the large European IXPs contributes more than 600 explicit peerings.

On the other hand, since Type 4 deployments tend to be used to connect Akamai with bigger networks in terms of bandwidth (not necessarily footprint) than the majority of networks with which Akamai peers at IXPs, they typically contribute fewer explicit peerings than Type 3 deployments. For example, in the case of an actual Type 4 deployment that is located in the same metro area as the Type 3 deployment we just considered, we find that it connects to only seven different networks that include two big cloud providers, two large eyeball and transit providers, two smaller eyeball providers, and one global provider from which Akamai buys transit. As a result, this deployment only contributes seven explicit peerings to Akamai's connectivity fabric.

Summary: *The footprint of Akamai's serving infrastructure consists of EUF delivery clusters in some 3.3k deployments across the globe that are used to serve a total of 1.75M unique IPv4 originating prefixes (plus 97k unique IPv6 originating prefixes) in 61.3k ASes. These observed prefixes can be served via a total of 3M unique AS paths, where prefixes of length /25 or longer are typically only reachable via a single path. The connectivity fabric of Akamai's serving infrastructure is made up of 6,111 explicit and 28,655 implicit peerings where the latter consist of 28,353 downstream-related and 1,506 upstream-related implicit peerings. Importantly, while some of the observed explicit peerings can be recognized in* ViewP*, none of the implicit peerings are visible in* ViewP.

4 THE SERVING INFRASTRUCTURE OF AKAMAI: PERFORMANCE

We show in this section how Akamai utilizes its connectivity fabric to serve content to end users worldwide and examine the performance (e.g., RTT, throughput) that this content experiences as it traverses Akamai's edge.

4.1 Available Datasets

To study performance-related aspects, we rely on server logs of the HTTP/S sessions between all EUF delivery servers and request-generating clients. These logs contain transport-layer information for a sample of all the HTTP/S sessions. Each server uses a sampling rate of 5% (i.e., 1-in-20 HTTP/S sessions) and for each sampled HTTP/S session, the server logs a record. Among the fields in each record are the IP address of the client, the IP address of the server, the total number of bytes sent to the client, the corresponding transfer time, and a smoothed round-trip time (RTT) value. This value is an estimate of the RTT between the server and the client. Transferring larger objects allows for better estimations of the RTT (more round trip samples). For the purpose of this study, we only consider HTTP/S sessions for objects larger than 300KB. Using the total bytes sent and the corresponding transfer time, we compute the session's mean throughput and use that value and the smoothed RTT as our metrics-of-choice for quantifying the performance of a session. We obtain one day's worth of logs for 2017-09-17 and 2018-05-17, the days corresponding to our first and last snapshot of ViewA, respectively. Each of these two logs contains a total of more than 11 billion records. We rule out possible sampling bias in this data by leveraging its substantial size; that is, when examining various randomly chosen subsets, we find that all of Akamai's deployments with at least one

EUF delivery server cluster and more than 90% of the full dataset's prefixes are present in all the subsets.

Since our analysis of the two log datasets corresponding to the 2017-09-17 and 2018-05-17 snapshots of ViewA produced very similar results, our focus below is on describing the observed findings for 2017-09-17. However, we will also include sample plots (see Figure 7) that explicitly compare the results for the two eight months-apart snapshots and quantify the observed similarity.

4.2 Akamai's Peering Edge "in Use"

Our analysis in Section 3 of the 2017-09-17 snapshot of ViewA revealed a vast and complex connectivity fabric that Akamai can leverage for serving content. This analysis was strictly control plane-oriented and relied exclusively on BGP information. In the process, we showed that out of the approximately 21M paths that could be easily discerned from ViewP-like datasets around the time of our analysis, Akamai ignores some 85% of them and only presents the 3.7M or so best paths to its mapping system. In the following, we leverage Akamai's 2017-09-17 server log measurements to provide an Akamai-focused data plane perspective of these 3.7M paths. That is, we are interested in understanding how the large number of identified implicit peerings and smaller number of explicit peerings are used to enable and facilitate Akamai's content delivery service.

To quantify Akamai's use of its implicit and explicit peerings, we proceed as follows. For each of the log records, we use the IP address of the server to associate servers with their corresponding deployment. Likewise, we use the IP address of the client to group together records by client AS. Next, we rely on information about the deployments (i.e., deployment type, link information where applicable) and the ViewA data to determine for each record the AS path from the deployment to the client AS. Subsequently, we group all the log records into the following four categories: (i) Onnet (all HTTP/S sessions served from Type 1 deployments), (ii) Transit (all HTTP/S sessions served from Type 2 or via the transit links of Type 3 or Type 4 deployments), (iii) IXP (all HTTP/S sessions served via the IXP links of Type 3 deployments), and (iv) PNI (all HTTP/S sessions served via the PNI links of Type 4 deployments).

This way, we end up with records that are all annotated with attributes indicating deployment, client AS, AS path, and link type or category (i.e., onnet, transit, IXP, and PNI). By examining the AS path of each such annotated record, we can thus identify AS paths that we only see from Type 1 and/or Type 2 deployments (i.e., using an implicit peering), or only from Type 3 and/or Type 4 deployments (i.e., using an explicit peering), or from a combination of Type 1/Type 2 and Type 3/Type 4 deployments (i.e., using both an implicit and explicit peering). In fact, using this information provides an opportunity to infer the likely type of client AS via the expected demand that this client AS generates for Akamai, at least for client ASes that operate in marketplaces with a well-developed Internet infrastructure. For example, seeing an AS path from both implicit and explicit peerings usually indicates that the traffic demand is medium to high in which case the client AS typically represents either a medium eyeball network that hosts Type 1 deployments and participates in public peering (i.e., peers with Akamai at one or more IXPs) or a large network with many eyeballs that hosts Type 1 deployments and peers privately with Akamai (Type 4 deployments). In a similar fashion, inferences can be drawn when seeing an AS path from implicit peerings only or from explicit peerings only.

Figure 4 summarizes our findings and shows three bars whose height (y-axis) represent the percentage of unique paths seen exclusively from implicit peerings only, from explicit peerings only, and from both implicit and explicit peerings, respectively. By using the width of the bars to encode traffic volume, we observe that the paths that are seen from both implicit and explicit peerings are the fewest in numbers but are responsible for about half of all the traffic served. This result confirms our intuition that the big networks that generate strong demand which translates into large amounts of traffic being served by Akamai are well connected to Akamai. At the same time, the largest number of paths is seen by explicit-only peerings, but these paths generate the least amount of traffic. In this case, the explanation is that from peerings at IXPs, Akamai can reach many destinations but the aggregate demand/traffic is typically low. The paths seen exclusively by implicit peerings occupy a middle ground, both in terms of number of paths and volume of traffic generated. Note that this middle ground appeals to small to medium-sized (eyeball) networks, hosting Type 1 deployments while receiving some types of content from their upstream provider(s) or being served by Type 2 deployments exclusively, that generate limited demand/traffic volume.

Finally, Figure 5 shows an extreme case of skewness with respect to the demand generated by the various client ASes or, equivalently, the traffic Akamai is serving to those client ASes.[5] For one, we observe that when considering all paths, some 90% of the overall demand is coming from about 1% of the paths. Similar findings apply when considering all paths seen from implicit-only or explicit-only peerings. Moreover, when looking at only those paths that are seen from both implicit and explicit peerings, we see a slightly less pronounced skewed demand distribution but recall that the percentage of such paths is small compared to implicit-only or explicit-only (see Figure 4). The skewness of Internet path usage is not new and has been observed in the past [37].

[5] Filtering out all requests for objects smaller than 300KB and working with sampled data may impact our findings quantitatively but not qualitatively.

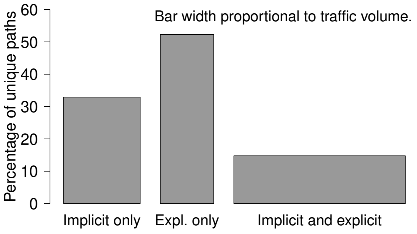


Figure 4: Paths by peering type.

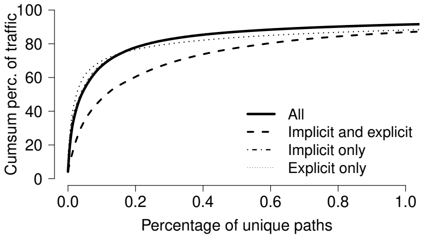


Figure 5: Traffic by unique paths.

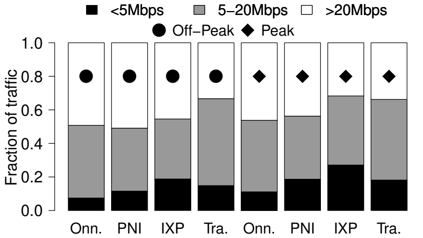


Figure 6: Throughput at (off-)peak.

4.3 Illustrative Examples

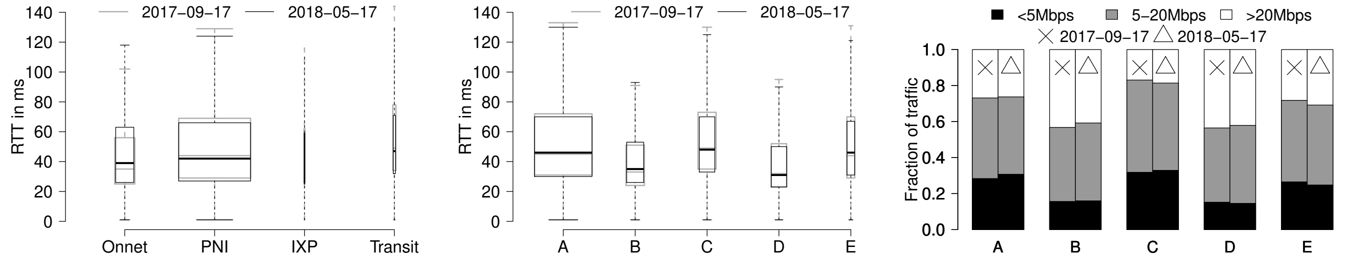
We next illustrate with three real-world scenarios how Akamai's content delivery service performs "in the wild". To this end, we examine the performance (e.g., RTT, throughput) that content experiences as it traverses Akamai's edge in different regions around the world; that is, between the EUF delivery server that Akamai's mapping system identified as being best suited to serve a given content and the end user client that requested that content.

4.3.1 Serving a large ISP in a single country. We consider a large eyeball ISP in a European country. Both the country's population and Akamai's deployments are concentrated in one large metro area and a few medium-sized cities. Akamai's deployments serve more than 1Tbps at peak on a normal day to that ISP, and the vast majority of requests from this large ISP's end users is served from Akamai's deployments within the country, specifically from Type 1 deployments and the PNI links of Type 4 deployments. A very small fraction is served from Akamai's transit links (Type 2/Type 3/Type 4 deployments) and an even smaller piece from Type 3 deployments, even though this country is served by a large IXP in the metro area where the large eyeball ISP is present and Akamai has deployments.

Performance-wise, analyzing the requests that originate from the clients on this ISP's network, we find that Akamai serves more than 99% of all the requests from deployments that are either zero (i.e., Type 1 deployments) or one AS-hop away. Moreover, we find that the median RTT values are all around 25ms, irrespective of the four link types. However, as expected and shown in Figure 6, when examining throughput stratified into "less than 5Mbps", "between 5-20 Mbps", and "more than 20Mbps" and comparing between off-peak and peak hours, we observe a decrease in performance for all four link types at peak vs. off-peak, with IXP links showing the biggest performance hit, mainly because they are more likely to experience congestion during peak hours. Note, however, that in this real-world example, Akamai's mapping system directs only a negligible number of request-generating clients to Type 3 deployments.

4.3.2 Serving a country with multiple ISPs. This scenario is concerned with a different country where the five largest ISPs combined serve more than 80% of the country's end users. In this case, both the end user population and Akamai's deployments are more uniformly distributed across the country. The country has a large IXP and a few smaller geo-dispersed IXPs, and Akamai has deployments at all of them. In contrast, the five big ISPs have by and large no presence at those IXPs and severely limit their use of public peering. In total, Akamai's deployments serve more than 4Tbps at peak on a typical day to the five ISPs. We use this scenario to illustrate the similarities between the results of our analysis of the 2017-09-17 and 2018-05-17 snapshots of ViewA and their corresponding log datasets.

Figure 7a shows for the two different snapshots performance in terms of RTT by link type. The plots use the width of the boxes to encode the traffic volume served by Akamai, and where discernible, gray and black boxplots correspond to the 2017-09-17 and 2018-05-17 snapshots of ViewA, respectively. While, as expected, the IXP option is hardly used, these large providers and Akamai have good reasons to prefer onnet and PNI over transit. For one, to avoid transit cost and exert better control over the large volume of traffic that Akamai sends to those providers, PNI is preferable over transit. At the same time, to achieve best performance, onnet is a practical choice since it gets Akamai closest to the end users. Next, Figure 7b shows the same plot as in Figure 7a but differentiates by ISPs. To understand the patent differences in performance among these large providers, we first note that while all five ISPs operate a fixed-line network, ISPs A, C, and E also operate a cellular network, and it is well-known that mobile users experience in general higher RTTs. Another relevant factor that explains the differences in performance (not only with respect to RTT but also for throughput as shown in Figure 7c where for each ISP, the left and right stacked bar plots are for the 2017-09-17 and 2018-05-17 snapshots, respectively) is the different arrangements Akamai has with these large ISPs. In short, different providers serving end users in one and the same country make their own (different) decision about choosing from among the available link- and deployment options, and Figure 7 shows that those decisions matter when it comes to performance.



(a) RTT by link type.

(b) RTT by ISP.

(c) Throughput by ISP.

Figure 7: Serving a country with multiple ISPs.

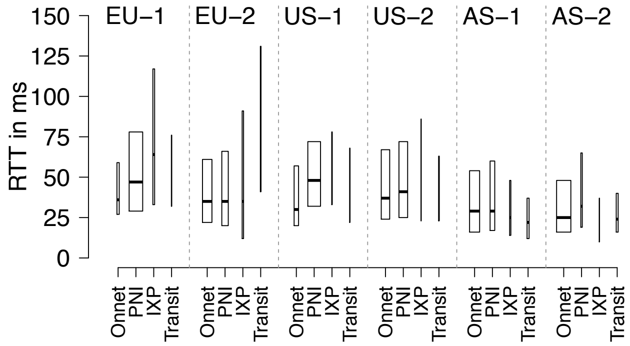


Figure 8: RTT by link type and metro.

4.3.3 Serving the world from different metros. For this example, we consider all of Akamai's deployments in six metro areas around the world and show again only results for the 2017-09-17 snapshot of ViewA. EU-1 and EU-2 are two metro areas in Europe that are considered to be major Internet hubs and have large IXPs; US-1 and US-2 are major metros on the east- and west-coast of the US, respectively, with substantial Internet infrastructure; and AS-1 and AS-2 are major commercial centers and Internet hubs in Asia. In all six metro areas, all four link- and deployment options are used, but as Figure 8 shows, to a varying degree. As in the previous figure, the different widths of the boxes encode the traffic volume and show that in all metro areas, onnet and PNI combined serve most of the traffic. Except for US-2, there are noticeable differences in traffic volumes between onnet and PNI, and they can be explained partly by differences in population size, partly by differences in availability, access, and cost of PNIs, and partly by how large ISPs in the different metros view Akamai or other large CPs as peering partners. For example, even though EU-1 and US-1 are Internet hubs with a high density of data centers, their population compared either to EU-2 and US-2, respectively, or to the total population in the two respective countries is not that large to justify a large number of Type 1 deployments.

Overall, onnet and PNI achieve the best performance (in terms of RTT), except for AS-1 where performance-wise, transit has a slight edge. Although transit is across the board a very small fraction of the traffic, its relative performance differs markedly for the three continents. In the US with its remaining Tier-1 ASes, transit performs similar to PNI, and in Asia, it performs good due to a strong reliance in transit providers in that region. To explain the wide ranges in RTT for IXP in EU-1 and EU-2 note that Akamai's deployments at European IXPs serve both local end users and end users in remote locations i.e., networks from other countries that connect to the IXPs (e.g., see [13]).

Summary: *When examining how traffic associated with actual user requests traverses Akamai's dense connectivity fabric, we find that some 90% of the overall traffic is coming from just 1% of the paths. This extreme skewness also holds for all those paths seen from explicit-only, implicit-only, and combined explicit-implicit peerings. Considering different scenarios around the world, we observe that different providers make different decisions about how to connect with Akamai and that these decision matter for performance.*

5 RELATED WORK

Two recent papers [42, 47] have contributed to a renewed interest in the actual structure and operations of the serving infrastructures of large CPs such as Facebook and Google and have demonstrated that the principles of SDN are applicable to public-facing networks. Complementing these systems-focused studies that offer only a few details about the actual connectivity fabric component of Facebook's or Google's serving infrastructures, our work provides a first-of-its kind in-depth account of this very connectivity fabric of Akamai, a large, global-scale CDN.

Our work is also related to prior research efforts on (i) mapping the footprints of different large CP infrastructures (e.g., see [3, 5, 25, 43, 45] and also [2, 7, 9]); (ii) providing new insights into the structure and evolution of the AS-level Internet (e.g., see [17, 33, 35, 41] and references therein), including intricate interconnectivity fabrics at the large IXPs across the world (e.g., see [4, 8, 11, 13, 40]); (iii) studying the flattening of the Internet (e.g., see [15, 16, 22, 30, 48]); (iv) exploring

CDN-ISP collaborations (e.g., see [21, 26, 38, 39]); and (v) optimizing content delivery to end users in a rapidly changing Internet (e.g., see [10, 29, 36, 44, 46, 49] and references therein). Our study is complementary to these and similar efforts; it provides a detailed account of the connectivity fabric of Akamai's serving infrastructure and illustrates how this large CP leverages this fabric to optimize the performance of content delivery as experienced by the end users.

6 DISCUSSION

Realizing that the serving infrastructures of today's large CPs come in different shapes and sizes and change in response to emerging technologies, new business models, and a constantly evolving Internet edge, we consider our detailed account of Akamai's current serving infrastructure and the breakdown of its connectivity fabric into its various components as a valuable reference point for examining its own evolution in time. For example, as part of our future work, we plan to study the evolution of this large CDN's serving infrastructure as a whole and of its connectivity fabric in particular as it leverages and expands its own multi-service backbone to transport its traffic between its own server clusters in a performance-aware and cost-effective manner [27, 28] and at the same time expands its business model to include more service offerings.

We also view our work to be an important step towards future efforts on understanding the serving infrastructures of other large CPs in general and on quantifying the advantages or disadvantages of one serving infrastructure design over another in particular. For example, there is some resemblance between the designs of Akamai's, Google's and Facebook's current serving infrastructures in the sense that they all utilize, in one form or another, highly-distributed collections of deployments (including deployments in third-party networks) that are organized in some hierarchical fashion on top of some specialized private backbone network. This observation begs the question about the optimality properties of this particular design choice over alternative designs (no deployments in third-party networks, no private backbone) when the all-important underlying objective of these large CPs is the delivery of content to end users in a cost-effective, performance-optimal, reliable and scalable manner. What makes studying this problem especially challenging is that it requires examining largely opaque and at times fast moving targets. That is, the large CPs view details about their serving infrastructures (including factors such as types of customers, services, and workload) as proprietary information, and even if publicly available, these details change over time.

Finally, by piecing together some of the interconnection options that are available in today's Internet to the large CPs and are utilized by them in practice (e.g., see Section 3.4), the following scenario describes an all-to-realistic use case. Take a large content owner/producer that utilizes the services of a large cloud provider to store/process its content. In turn, this cloud-based content is accessed by a large CDN that transports it across its private backbone for delivery to end users serviced by a large ISP. This content will typically traverse different PNIs all the way from the where it is produced to the large ISP's network and thus none of the associated voluminous traffic will be visible in the public Internet. This shift of traffic from the "public" Internet as we know it to the "private" Internet is real and massive and well-known among network operators (e.g., see [19]). However, by their very nature, the publicly available datasets that network researchers commonly use to study the evolution of the Internet's interconnection fabric and its traffic patterns say little if anything about the portion of the private Internet that can be expected to see (or already sees) most of the "action". The development of new methodologies that allow third parties to study different facets of the Internet's evolution looms as an important open problem for closing the gap between what network operators know based on empirical evidence and what network researchers can study and quantify based on relevant measurements.

7 CONCLUSIONS

Complementing recent studies that focus largely on the design of new SDN-based Internet peering edge architectures that enable today's large CPs to route their traffic at scale and in a performance-aware manner, our work provides the first account of the actual scale of the peering edge of such a large CP. By examining the actual connectivity fabric of the serving infrastructure of a large global-scale CDN, we show that it consists of about 6,100 explicit peerings and some 28,500 implicit peerings. The latter refer to existing interconnections between a third-party network and its downstreams that this CDN has access to and can utilize for its content delivery service simply by virtue of operating deployments in such third-party networks. We further illustrate how this CDN leverages this dense connectivity fabric for serving its content "to the ISPs of the world."

ACKNOWLEDGMENTS

We want to thank Christian Kaufmann for his detailed comments and valuable suggestions on earlier versions of this paper, Steve McManus for sharing his insights and knowhow about Akamai's control-plane data and BGP collectors, and Rick Weber for providing us access to his group's data collection and computational resources. We also thank Carl Princi, Jim Wyllie and Pat Larkin for their support of this project and our shepherd Stefano Vissicchio and the anonymous reviewers for their helpful and constructive feedback.

REFERENCES

[1] IPv4 & IPv6 CIDR Report. http://www.cidr-report.org/as2.0. Accessed: Jan. 2017.

[2] Vijay Kumar Adhikari, Yang Guo, Fang Hao, Volker Hilt, Zhi-Li Zhang, Matteo Varvello, and Moritz Steiner. Measurement Study of Netflix, Hulu, and a Tale of Three CDNs. *IEEE/ACM TON*, 23(6), 2015.

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

[4] Bernhard Ager, Nikolaos Chatzis, Anja Feldmann, Nadi Sarrar, Steve Uhlig, and Walter Willinger. Anatomy of a Large European IXP. In *ACM SIGCOMM*, 2012.

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

[6] Seth Bennet. Facebook Scalable Interconnection. https://www.peering-forum.eu/system/documents/124/original/09.30_-_Facebook_-_Seth_Bennet.pdf.

[7] Timm Böttger, Félix Cuadrado, Gareth Tyson, Ignacio Castro, and Steve Uhlig. A Hypergiant's View of the Internet. *ACM CCR*, 2017.

[8] Samuel Henrique Bucke Brito, Mateus A. S. Santos, Ramon dos Reis Fontes, Danny Alex Lachos Perez, and Christian Esteve Rothenberg. Dissecting the Largest National Ecosystem of Public Internet eXchange Points in Brazil. In *PAM*, 2016.

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

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

[11] Ignacio Castro, Juan Camilo Cardona, Sergey Gorinsky, and Pierre François. Remote Peering: More Peering without Internet Flattening. In *ACM CoNEXT*, 2014.

[12] H. Chang, S. Jamin, and W. Willinger. Internet Connectivity at the AS-Level: An Optimization-Driven Modeling Approach. In *ACM SIGCOMM Workshop on Models, Methods, and Tools for Reproducible Network Research*, 2003.

[13] Nikolaos Chatzis, Georgios Smaragdakis, Jan Böttger, Thomas Krenc, and Anja Feldmann. On the Benefits of Using a Large IXP As an Internet Vantage Point. In *ACM IMC*, 2013.

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

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

[16] Amogh Dhamdhere and Constantine Dovrolis. The Internet is Flat: Modeling the Transition from a Transit Hierarchy to a Peering Mesh. In *ACM CoNEXT*, 2010.

[17] Amogh Dhamdhere and Constantine Dovrolis. Twelve Years in the Evolution of the Internet Ecosystem. *IEEE/ACM TON*, 19(5), 2011.

[18] Hurricane Electric. Internet Statistics. https://bgp.he.net/report/netstats. Accessed: Jan. 2017.

[19] Equinix. Private Data Exchange Between Businesses Forecasted to Outpace the Public Internet by Nearly 2x in Growth and 6x in Volume by 2020. https://www.equinix.com/newsroom/press-releases/pr/123570/private-data-exchange-between-businesses-forecasted-to-outpace-the-public-internet-by-nearly-2x-in-growth-and-6x-in-volume-by-2020.

[20] Facebook. Building Express Backbone: Facebook's new long-haul network. https://code.facebook.com/posts/1782709872057497/building-express-backbone-facebook-s-new-long-haul-network.

[21] Benjamin Frank, Ingmar Poese, Yin Lin, Georgios Smaragdakis, Anja Feldmann, Bruce M. Maggs, Jannis Rake, Steve Uhlig, and Rick Weber. Pushing CDN-ISP Collaboration to the Limit. *ACM CCR*, 2013.

[22] Phillipa Gill, Martin F. Arlitt, Zongpeng Li, and Anirban Mahanti. The Flattening Internet Topology: Natural Evolution, Unsightly Barnacles or Contrived Collapse? In *PAM*, 2008.

[23] Google. Google Edge Network. https://peering.google.com.

[24] Packet Clearing House. Daily Routing Snapshots. https://www.pch.net/resources/Routing_Data.

[25] Cheng Huang, Angela Wang, Jin Li, and Keith W. Ross. Measuring and Evaluating Large-Scale CDNs. In *ACM IMC*, 2008.

[26] Wenjie Jiang, Rui Zhang-Shen, Jennifer Rexford, and Mung Chiang. Cooperative Content Distribution and Traffic Engineering in an ISP network. In *ACM SIGMETRICS/Performance*, 2009.

[27] Christian Kaufmann. Akamai ICN. https://pc.nanog.org/static/published/meetings/NANOG71/1532/20171003_Kaufmann_Lightning_Talk_Akamai_v1.pdf. NANOG 71 Light. Talk, 2017.

[28] Christian Kaufmann. ICN - Akamai's Backbone. https://www.linx.net/wp-content/uploads/LINX101-Akamai-ICN-ChristianKaufmann.pdf. LINX Meeting 101, 2018.

[29] Rupa Krishnan, Harsha V. Madhyastha, Sridhar Srinivasan, Sushant Jain, Arvind Krishnamurthy, Thomas E. Anderson, and Jie Gao. Moving Beyond End-to-End Path Information to Optimize CDN Performance. In *ACM IMC*, 2009.

[30] Craig Labovitz, Scott Iekel-Johnson, Danny McPherson, Jon Oberheide, and Farnam Jahanian. Internet Inter-Domain Traffic. In *ACM SIGCOMM*, 2010.

[31] RIPE NCC. RIS Raw Data. https://www.ripe.net/analyse/internet-measurements/routing-information-service-ris/ris-raw-data.

[32] University of Oregon. Route Views Archive Project. http://archive.routeviews.org.

[33] Ricardo V. Oliveira, Dan Pei, Walter Willinger, Beichuan Zhang, and Lixia Zhang. In Search of the Elusive Ground Truth: the Internet's AS-level Connectivity Structure. In *ACM SIGMETRICS*, 2008.

[34] Ricardo V. Oliveira, Dan Pei, Walter Willinger, Beichuan Zhang, and Lixia Zhang. The (in)Completeness of the Observed Internet AS-level Structure. *IEEE/ACM TON*, 18(1), 2010.

[35] Ricardo V. Oliveira, Beichuan Zhang, and Lixia Zhang. Observing the Evolution of Internet as Topology. In *ACM SIGCOMM*, 2007.

[36] John S. Otto, Mario A. Sánchez, John P. Rula, and Fabián E. Bustamante. Content Delivery and the Natural Evolution of DNS: Remote DNS Trends, Performance Issues and Alternative Solutions. In *ACM IMC*, 2012.

[37] Vern Paxson. End-to-end Routing Behavior in the Internet. In *ACM SIGCOMM*, 1996.

[38] Ingmar Poese, Benjamin Frank, Bernhard Ager, Georgios Smaragdakis, and Anja Feldmann. Improving Content Delivery Using Provider-aided Distance Information. In *ACM IMC*, 2010.

[39] Ingmar Poese, Benjamin Frank, Bernhard Ager, Georgios Smaragdakis, Steve Uhlig, and Anja Feldmann. Improving Content Delivery with PaDIS. *IEEE Internet Computing*, 16(3), 2012.

[40] Philipp Richter, Georgios Smaragdakis, Anja Feldmann, Nikolaos Chatzis, Jan Böttger, and Walter Willinger. Peering at Peerings: On the Role of IXP Route Servers. In *ACM IMC*, 2014.

[41] Matthew Roughan, Walter Willinger, Olaf Maennel, Debbie Perouli, and Randy Bush. 10 Lessons from 10 Years of Measuring and Modeling the Internet's Autonomous Systems. *IEEE Journal on Selected Areas in Communications*, 29(9), 2011.

[42] Brandon Schlinker, Hyojeong Kim, Timothy Cui, Ethan Katz-Bassett, Harsha V. Madhyastha, Ítalo Cunha, James Quinn, Saif Hasan, Petr Lapukhov, and Hongyi Zeng. Engineering Egress with Edge Fabric: Steering Oceans of Content to the World. In *ACM SIGCOMM*, 2017.

[43] Florian Streibelt, Jan Böttger, Nikolaos Chatzis, Georgios Smaragdakis, and Anja Feldmann. Exploring EDNS-Client-Subnet Adopters in Your Free Time. In *ACM IMC*, 2013.

[44] Ao-Jan Su, David R. Choffnes, Aleksandar Kuzmanovic, and Fabián E. Bustamante. Drafting Behind Akamai (Travelocity-based Detouring). In *ACM SIGCOMM*, 2006.

[45] Ruben Torres, Alessandro Finamore, Jin Ryong Kim, Marco Mellia, Maurizio M. Munafò, and Sanjay G. Rao. Dissecting Video Server Selection Strategies in the YouTube CDN. In *IEEE ICDCS*, 2011.

[46] Marc Anthony Warrior, Uri Klarman, Marcel Flores, and Aleksandar Kuzmanovic. Drongo: Speeding Up CDNs with Subnet Assimilation from the Client. In *ACM CoNEXT*, 2017.

[47] Kok-Kiong Yap, Murtaza Motiwala, Jeremy Rahe, Steve Padgett, Matthew J. 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, Muhammad Mukarram Bin Tariq, Matt Tierney, Dzevad Trumic, Vytautas Valancius, Calvin Ying, Mahesh Kallahalla, Bikash Koley, and Amin Vahdat. Taking the Edge off with Espresso: Scale, Reliability and Programmability for Global Internet Peering. In *ACM SIGCOMM*, 2017.

[48] Bahador Yeganeh, Reza Rejaie, and Walter Willinger. A View From The Edge: A Stub-AS Perspective of Traffic Localization and Its Implications. In *IEEE TMA*, 2017.

[49] Hyunho Yeo, Sunghyun Do, and Dongsu Han. How will Deep Learning Change Internet Video Delivery? In *ACM HotNets*, 2017.