Satellite: Joint Analysis of CDNs and Network-Level Interference

Will Scott, Thomas Anderson, Tadayoshi Kohno, and Arvind Krishnamurthy,
University of Washington

https://www.usenix.org/conference/atc16/technical-sessions/presentation/scott

This paper is included in the Proceedings of the 2016 USENIX Annual Technical Conference (USENIX ATC '16).

June 22–24, 2016 • Denver, CO, USA

978-1-931971-30-0

Open access to the Proceedings of the 2016 USENIX Annual Technical Conference (USENIX ATC '16) is sponsored by USENIX.

Satellite: Joint Analysis of CDNs and Network-Level Interference

Will Scott, Thomas Anderson, Tadayoshi Kohno, and Arvind Krishnamurthy
{wrs, tom, yoshi, arvind}@cs.washington.edu

Abstract

Satellite is a methodology, tool chain, and data-set for understanding global trends in website deployment and accessibility using only a single or small number of standard measurement nodes. Satellite collects information on DNS resolution and resource availability around the Internet by probing the IPv4 address space. These measurements are valuable in their breadth and sustainability - they do not require the use of a distributed measurement infrastructure, and therefore can be run at low cost and by multiple organizations. We demonstrate a clustering procedure which accurately captures the IP footprints of CDN deployments, and then show how this technique allows for more accurate determination of correct and incorrect IP resolutions. Satellite has multiple applications. It reveals the prevalence of CDNs by showing that 20% of the top 10,000 Alexa domains are hosted on shared infrastructure, and that CloudFlare alone accounts for nearly 10% of these sites. The same data-set detects 4,819 instances of ISP level DNS hijacking in 117 countries.

1 Introduction

After several generations of elaborate measurement platforms, it remains difficult to characterize how web content is distributed and the extent to which it is open and unfettered. This lack of understanding is reflected in the questions we cannot easily answer: Which countries have servers operated by Google or Microsoft, and what is the footprint of various content distribution networks (CDNs)? Which websites have degraded availability due to network interference? Which sites are powered by various CDNs such as CloudFlare or Fastly? Which ISPs run caching proxies or other stateful middle-boxes? And so on.

Measurements that characterize web access is of value for publishers and end users alike. For publishers, it allows for informed choice about how to locate their content – which CDNs to use, and appropriate trade-offs between security and connectivity. Website optimization is dependent on the expected latency of user connections, which is difficult for publishers to predict in advance of choosing hosting or providers. For users, these measurements provide insight into whether operators comply with local regulations, and which sites will be able to warehouse data within their jurisdiction. Transparency in network censorship (what sites are being blocked, and how) is also critical to regulatory oversight and informing public debate. These lists are almost always kept secret, and even when available, it is difficult for watchdogs to verify that they reflect reality. This opacity makes it difficult for users to advocate for changes in policy or trust existing systems.

While we have some understanding of what measurements can address these questions, there is no existing data set or measurement platform that holds the answers. In fact, there are many challenges both in collecting the measurement data and analyzing it to characterize the current state of web content distribution.

First, we need measurements from globally distributed vantage points in order to characterize global website accessibility. Understanding both the Internet-wide expansions of CDNs and the interference practices around the world requires data from a diverse set of ISPs. Such a measurement platform does not exist yet in a public and transparent manner that supports reproducibility of results. Distributed collection platforms also face concerns of retribution towards those hosting the measurements.

Second, since deployment and accessibility characteristics can change rapidly, collection must be fine-grained and timely. Many documents of interference choose a single fixed point in time, or provide yearly updates. To be effective, we need to provide alerts that policies have changed quickly, which raises questions regarding how many domains can be monitored and what granularity is sufficient.

Third, the analysis of how websites employ CDNs and the identification of network interference must be tackled jointly. For example, when ISPs block websites by redirecting them to a block page, those servers are easily misconstrued as a CDN node for that geographical region. Consider the example of twitter.com. As shown in Table 1, the domain resolves to different IPs in the US, Russia, and China. A naive CDN mapping would conclude that there are likely points of presence in all three countries, while a naive interference measurement might conclude interference in both China or Russia, or might give up due to the diversity of IPs returned. In reality, the Russian IP maps to a Twitter CDN node, while

Resolver	Response	Behavior
USA (8.8.8.8)	199.59.149.198	Twitter
Russia (77.88.8.8)	199.16.156.102	Twitter
China (180.76.76.76)	159.106.121.75	Failure

Table 1: Resolutions of Twitter.com by different resolvers

the Chinese resolution is due to interference.

In the remainder of this paper, we demonstrate Satellite, an automatic framework that is able to identify CDN infrastructure in tandem with anomalies. Through the use of Internet scanning and reflecting queries on public infrastructure we avoid the pitfalls that come with distributed infrastructure. By clustering sites based on DNS resolution, and by finding responses which do not fit those clusters, we are able to identify interference without misclassifying CDNs. This analysis is able to both monitor the growth of shared hosting platforms, and which sites are blocked by network-level interference.

We address the need for a distributed measurement platform by using a single end-host to collect DNS resolutions from a large number of globally-distributed and open DNS resolvers. Instead of pursuing crowd-sourced deployments or analyzing limited snapshots of data obtained from operators in privileged positions, we instead focus on what is possible from active measurements conducted by a single end-host. Doing so both reduces the barrier to entry for organizations to run their own independent measurements, and removes the complex work of coordinating a distributed testbed and verifying the untrusted dataset collected from it. While results from a single machine may be biased, the validation steps are the same as those of distributed infrastructure; it continues to be the case that you can't definitively claim interference from a single instance of connection failure, and instead extract evidence from aggregate trends.

We choose DNS as our main platform of measurement because it has developed as a narrow waist that is used both by CDNs for routing traffic and for the interposition of block pages by ISPs and nations. CDNs use the DNS resolution process for load balancing and routing because it is the first step in a web page access; making a good decision at the DNS level ensures fast connections for the rest of the loading process. Network interference also often occurs at the DNS layer, because while a single IP may host content for many sites, DNS requests have an easily parseable format and allow restriction of specific domains. The existence of shared infrastructure that hosts many sites (CDNs) is exactly why interference continues to be commonly implemented at a resolution level.

We address the need for timely global measurements by designing a system that can measure the global connectivity of tens of thousands of domains with weekly precision. By focusing on coverage rather than a specific event or geographic region, Satellite acts as a database supporting higher level analysis of policy changes as they occur. By measuring the Alexa top 10,000 global domains, we are able to detect evidence of interference in many countries and automatically detect most popular shared infrastructure without manual targeting of measurement.

We address the need for joint understanding of infrastructure and interference through our algorithmic interpretation of Satellite data. We correlate the addresses of domains across ISPs and learn the customer pools of CDNs. Looking at the pools of IPs, we can learn the points of presence of CDNs and which CDNs have business relationships with which ISPs. By looking at which locations resolve to which points of presence we can understand the geographic areas served by different points of presence. By tracing the patterns of divergence from clusters, we are able to separate the effects of network interference from confounding site distribution factors.

Satellite has limitations in the view of Internet infrastructure it reveals. Some shared services explicitly partition incoming requests across disjoint sets of servers. Dedicated IP addresses are used to support SSL for some old browsers, to reduce dynamic generation of certificates, and as part of fault isolation strategies. Akamai is an example of such a shared infrastructure. Satellite does not report these platforms as single entities, but rather as multiple smaller shards, defining the more specific subsets of IPs assigned to each customer (i.e., domain).

Satellite is a fully open project consisting of the code for data collection and analysis, a growing year-long repository of collected data, and derived views of site structure and interference. Satellite is built for transparency, minimizing the trust that needs to be placed in the system or its operators. We are working with several independent organizations to independently collect and corroborate data. We hope this structure enables future researchers to trust collected data without the need to replicate collection work. Building Satellite to run on a single machine is aimed to maximize the sustainability of the project, and our ability to amass a longitudinal data set of changing Internet behavior.

The major contributions of Satellite are:

- A single-node measurement system for monitoring global trends in network interference and CDN deployment.
- An algorithm for the joint analysis of network anomalies and determination of shared infrastructure from point measurements of domain resolution.
- Data on the distribution and accessibility of 10,000 popular domains over the last two years.

In the remainder of this paper we will elaborate on the design and operation of Satellite, and present some of the site behavior we have discovered through these measurements.

2 System Design

Satellite is motivated by a number of explicit design goals differentiating it from existing platforms and systems.

- **External Data Collection**: We want the system to function without requiring in-situ resources. This avoids the need to recruit volunteers, and focuses on safety and coverage across networks.
- **Continuous Measurement**: We want the system to be able to quickly notice changes in CDN deployments and network access policies.
- **Transparent and Ethical Measurements**: We want the system to be transparent, so that others can easily trust and make use of collected data. We aim for high ethical standards to minimize harm to DNS server operators from collected data.
- **Joint analysis of CDN deployments and Network Interference**: We want a system which simultaneously measures shared infrastructure and interference of web access, since the two are tightly intertwined.

2.1 System Overview

The Satellite system is arranged as a pipeline which collects and analyzes data. It is run as a weekly cron job, which schedules data collection, and performs initial aggregation, analysis and archiving of each data set. The implementation details of the pipeline are described in more detail in Section 3. At a high level, Satellite is structured into the following discrete tasks:

Identifying DNS resolvers by scanning the Internet. We detect active, open, long-lived DNS resolvers through active probing.

Assembling a target domain list by expanding a list of popular domains to ensure CDN coverage.

Performing active DNS measurements where candidate domains are measured against discovered resolvers.

Collection of supplemental data where organization metadata and geolocation hints are gathered.

Aggregation of DNS resolutions by combining records at the AS level to allow for efficient processing in subsequent analysis.

Joint analysis of CDNs and network interference through the calculation of fixed-points in clusters of domains believed to use shared infrastructure.

Export of measurement results by publishing visualizations and data sets with footprints of CDNs and significant observed anomalies.

2.2 Identifying DNS resolvers

Our measurements are based on gathering data on how domains behave for different clients around the world. There are several options available for this type of collection. Traditionally, researchers have used cooperating hosts in a variety of networks [24, 30]. More recently, the EDNS extension has allowed clients to indicate that they are asking for a response that will be used by someone in a different geographic area to approximate multiple vantage points [35, 6]. Very few domain name servers support EDNS, but we can take advantage of the same behavior the mechanism is designed to fix. By making requests to many resolvers, we can learn the different points of presence for target domains. For instance, the `8.8.8.8` resolver is operated by Google and provides a US-centric view of the world, while `180.76.76.76`, "BaiduDNS", provides a Chinese centric view.

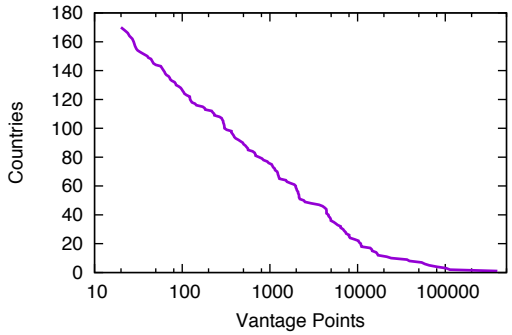

Figure 1: DNS servers discovered in each Country. We find 169 countries hosting DNS resolvers in more than 20 class-c networks.

We enumerate the IPs acting as DNS resolvers by probing the IPv4 Address space with zmap [12]. Compared to 32 million open DNS servers monitored by the Open Resolver Project [28] (a service measuring the potential for reflected denial of service attacks through DNS, it does not share the IPs of discovered servers), we independently discover 12 million servers which respond to requests with a well-formed response. Of these, 7 million servers across 1.5 million class-c (/24) networks offer recursive resolution and give a correct IP address when asked to resolve our measurement server. These servers provide coverage of 20,000 ASes (Autonomous Systems, typically representing an ISP), and cover 169 countries with at least 20 class-c networks, as shown in Figure 1.

2.3 Ethics of Collection

Our measurements prompt machines in remote networks to resolve domains on our behalf. This traffic to remote networks may result in unintended consequences to these relays, and as such we do our best to minimize harm in keeping with best practices [11].

Open DNS resolvers are a well known phenomenon, and lists of active resolvers can be downloaded without the overhead we incur in scanning. We find that the act of scanning the IPv4 address space to find active resolvers does generate abuse complaints from network operators. By maintaining a blacklist of networks which have requested de-listing (less than 0.5% of the address space), we have not received any complaints related to our scan-

ning or subsequent resolutions in the last quarter. Some operators have asked us to keep their network spaces private, which prevents us from releasing this list publicly. Others running the system should expect to recreate a similar list. We have never received a complaint from overloading a DNS resolver with queries for our tracked domains.

We abide by the 7 harm mitigation principles for conducing Internet-wide scanning outlined by the zmap project [12]. In particular, we (a) coordinated with the network administrators at our university in handling complaints, (b) ensured we do not overload the outbound network, (c) host a web page explaining the measurements with an opt-out procedure, and have clear reverse DNS entries assigned to the measurement machine, (d) clearly communicate the purpose of measurements in all communications, (e) honor any opt-out requests we receive, (f) make queries no more than once per minute, and spread network activity out to accomplish needed data collection over a full one-week period, and (g) spread the traffic over both time and source addresses allocated to our measurement machine.

To get a better sense of the impact our queries have on resolvers, we operated an open DNS resolver. In a 1 week period after running for 1 month, the resolver answered over one million queries, including 800,000 queries for domains in the Alexa top 10,000 list. Satellite made only 1,000 of these requests.

We have additionally adopted a policy of only probing DNS servers seen running for more than one month to reduce the potential of sending queries to transient resolvers. This reduces our resolver list by 16%[1]. Measurements in IP churn indicate that the bulk of dynamic IPs turn over to subsequent users on the order of hours to days, making it unlikely that our measurements target residential users [38].

2.4 Mapping CDNs and Network Interference

We know that for many CDNs it is common to resolve domains to different IP addresses based on where the client is. While the diversity of IPs makes it more difficult to understand what an 'unexpected' deviation is, the primary insight we can use is that in many cases these CDN infrastructures are shared by many sites. The set of sites on a shared infrastructure is often independent of the set of sites which are targeted by network interference.

Consider the case of `thepiratebay.se`, a domain hosted with `strawpoll.me` on Cloudflare. In a US location, like the DNS resolver operated within UC Berkeley (AS25), both domains resolve to IPs in the `141.101.118/24` subnet. However, across many networks in Iran (for instance AS50810), the first resolves instead to `10.10.34.36`, an internal LAN address, while the second continues to resolve to Cloudflare owned IPs.

[1] Specifically comparing the live resolvers discovered between March 20th and April 20th, 2015.

$domains \leftarrow$ the set of all domains
$ips \leftarrow$ the set of resolved IPs
function EDGE($domain, ip$)
  return |ASes where $domain$ resolved to ip|
end function
function IPTRUST($domain, ip$)
  $\triangleright\ 0-1$ value representing how likely an IP is a server for a domain.
  return $\frac{\sum_{d \in domains} \text{EDGE}(d,ip) * \text{DOMAINSIMILARITY}(domain,d)}{\sum_{d \in domains} \text{EDGE}(d,ip)}$
end function
function WEIGHT($domain, ip$)
  $\triangleright$ EDGE weighted by IPTRUST.
  return $\text{EDGE}(domain, ip) * \text{IPTRUST}(domain, ip)$
end function
function DOMAINSIMILARITY(dom_a, dom_b)
  $\triangleright\ 0-1$ value representing how likely two domains are hosted on the same servers.
  return

$$\frac{\sum_{ip \in ips} \text{WEIGHT}(dom_a, ip) * \text{WEIGHT}(dom_b, ip)}{\sqrt{\sum_{ip \in ips} \text{EDGE}(dom_a, ip)^2} * \sqrt{\sum_{ip \in ips} \text{EDGE}(dom_b, ip)^2}}$$

end function

Figure 2: Pseudocode of CDN and interference detection joint analysis algorithm. The two functions `DomainSimilarity` and `IPTrust` depend on each other, and are iteratively computed until a fixed point is approximated. The result of these two functions allows direct determination of both the IPs hosting clusters of domains, and the resolutions which are anomalous.

To automate this form of detection, we automatically find cliques of domains hosted on the same infrastructure, and use the combined resolutions of those domains to map the IPs of the underlying infrastructure. Using multiple domains will help us to overcome the randomness present in individual domain resolutions, and to notice when one domain behaves strangely in a specific geographic region. We are not using IP metadata to map provider infrastructure, but rather the sets of IPs (potentially across providers) that form the footprints of popular domains.

To process the data, we perform a joint analysis using the algorithm in Figure 2 (also described in text below). Then, we use the stable values from that computation to extract cliques and deviations, which represent shared infrastructure and interference respectively.

2.4.1 Joint Analysis Algorithm

Given a bipartite graph linking IP addresses and domains, our goal is to separate the graph into two components: 'real infrastructure', and 'interference'. An intuition of how to think of this separation is shown in Figure 3. To find this separation, we compute two quantities: A similarity metric `DomainSimilarity`, for how close

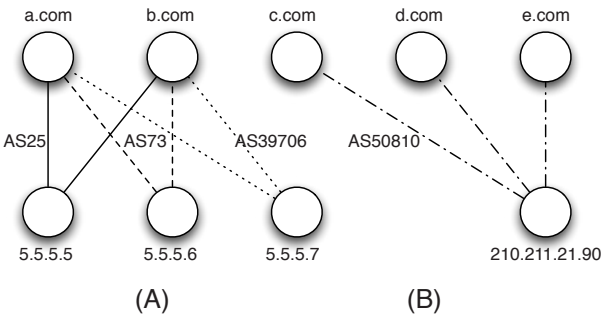

Figure 3: The relationship between Domains and IPs. Each edge corresponds to a resolution, labeled by the autonomous system of the resolver. In this example, we see *a.com* resolve to `5.5.5.5` in UC Berkeley, AS 25. In (A) we see a clique of domains supported by the same infrastructure, while (B) shows otherwise unrelated domains resolving to the same IP within AS 50810.

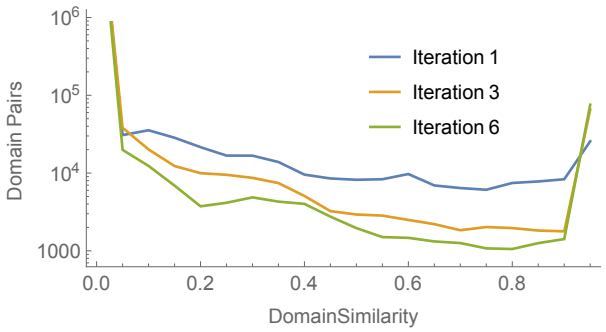

Figure 4: `DomainSimilarity` distribution after iterative calculation. After the first iteration, 25,000 edges with similarity above 95% are found. After 5 iterations there are 75,000 strong similarities, fewer 'uncertain' similarities, and less than 1000 similarities that changed by more than 5%.

two domains are, and a trust metric `IPTrust`, for how likely an IP is to be an authentic resolution for a given domain. In Figure 3a, we would hope that *a.com* and *b.com* have a high `DomainSimilarity`, since they resolve to the same IPs. In Figure 3b, we would hope the IP `210.211.21.90` has a low `IPTrust` score, since many otherwise unrelated domains resolve to it. This process is similar to the HITS algorithm for finding "authoritative" sources for pages [23].

The `DomainSimilarity` metric specifically represents the fraction of the time that two domains resolve to the same IPs. We use the different IPs as independent dimensions in which the resolutions of each domain can be represented as a vector. The distance between Domains is then the cosine distance between the two resolution vectors.

The `IPTrust` metric calculates the confidence for whether any given IP address resolution of a domain is correct. To calculate our confidence in a resolution, we say the probability a domain resolves to an IP is equal to the average similarity between that domain and the other domains which have resolved to that IP. To score whether we believe that `thepiratebay.se` resolves to `10.10.34.36`, we would look at other domains which have resolved to `10.10.34.36` and consider their `DomainSimilarity` with `thepiratebay.se`.

We now discuss cases where a provider allocates non-disjoint but partially overlapping sets of IPs to different domains. For example, if a domain `a.com` resolves to IPs A,B, and C, while `b.com` resolves to C, D, and E. If the different IPs are in the same class-c network, then our analysis will see both `a.com` and `b.com` as resolving to the class-c network that corresponds to A, B, C, D, and E, thus attributing a high `IPTrust` value to the class-c network for the two domains. Class-c is chosen as the most specific public announcement of IP ownership, limiting accidental grouping of different providers. If the IPs are in different class-c networks, the `IPTrust` can still be high if the `DomainSimilarity` is high. In cases where there is only a small fraction of IP space overlap, metadata is not present, and `DomainSimilarity` is low, Satellite will consider the two domains to be in separate clusters. This will attribute a low `IPTrust` to C.

For intuition behind these metrics, consider the representative case of the Fastly CDN. Taking one IP range, `23.235.47.0/24`, we find that across the 72 domains Satellite clusters as Fastly the `IPTrust` metric ranges between 0.75 and 0.98. Across all other domains, this IP range was seen as a resolution for 22 other domains, across which its average `IPTrust` was 0.20 and maximum was 0.30. The range of `IPTrust` in the Fastly cluster shows that the strongly connected cluster boosted scores for IPs that were infrequently resolved for some domains.

To derive an initial estimate of `DomainSimilarity`, we set `IPTrust` to 1.0. We then iteratively calculate these two quantities until a fixed point is approximated, generally in 5-6 iterations. Figure 4 shows the effect of iteration on the distribution of domain similarities. Without iterating to the fixed point, many domain pairs have a similarity coefficient close to 0.5. Subsequent iterations concentrate the emergent clusters to more clearly define shared infrastructure (close to 1.0) as well as block-pages (close to 0.0).

2.4.2 Cliques and Deviations

`DomainSimilarity` and `IPTrust` form the core metrics we need to determine both CDN footprints (the cliques of similar domains and associated set of IP addresses they're served from), and network anomalies (sets of domains sent to IPs with low trust in isolated ASNs).

CDN cliques: To find clusters of domains with similar resolutions in the matrix of calculated `DomainSimilarity` values, we use a greedy

Domain	Alexa Rank
www.ebay.com	18
cntv.cn	79
indiatimes.com	110
dailymail.co.uk	114
etsy.com	149
cnet.com	151
deviantart.com	168
forbes.com	175

Table 2: The highest ranked domains identified in the largest 'Akamai' cluster.

CDN	Size	Representative Domain
CloudFlare	726	reddit.com
Amazon AWS	647	amazon.com
Akamai	410	ebay.com
Google	141	google.com
Dyn	112	webmd.com
Rackspace	77	wikihow.com
Fastly	72	imgur.com
Edgecast	68	soundcloud.com
Incapsula	55	wix.com
AliCloud	54	163.com

Table 3: Largest CDN clusters. The top 10 CDNs account for 20% of monitored domains.

algorithm of first making arbitrary clusters, and then finding the best 'swaps' possible until a local maxima is found [13]. This clustering technique has been found to perform close to human labeling.

Table 2 shows an example of the highest popularity sites that were clustered into the clique representing the Akamai infrastructure. The largest clusters are shown in Table 3. We count the 10 largest shared hosting platforms hosting 1967 domains, making up almost 20% of those measured.

At a global level, strongly connected components represent domains hosted by the same servers. This may be domains resolving to one IP everywhere, or domains with the same CDN configuration which consistently resolve to the same IPs from different vantage points. If we narrow our consideration to the ASes based in a single country, blocking can also appear as a cluster with the block page IP clustered with all of the blocked domains. These clusters are only found in the ASes of individual countries, and the difference between detected clusters globally and nationally is a strong signal for this behavior. On the other hand, this is only one of many ways to interfere with DNS. Some forms, like the response of random IPs used by some Chinese ISPs [5], will reduce `IPTrust` without creating these obvious clusters.

It should be at first surprising that Akamai, one of the largest CDN providers, is represented by a low number of domains. We find that while Akamai transfers a large amount of traffic, we count many of their domains as independent entities for two reasons. First, Akamai often delivers home pages as a relatively small set of IP addresses that are dedicated to HTTPS for the specific customer. Second, Akamai is located in over 1000 different ISPs, with most IPs assigned to servers advertised and using those IPS's AS numbers as their origins. These two factors cause many Akamai customers to be treated as independent entities by Satellite, and not seen as part of their shared serving infrastructure.

We can compare the relationship Akamai has with customers to that of Cloudflare, which also provides 'white-label' services for large customers to customize their presence through custom DNS name servers and SSL deployed for older clients unable to perform server name identification. Cloudflare partitions its customers across several distinct IP spaces. Some of these IPs have reverse PTR and whois information identifying them as Cloudflare, while others do not. The use of IP addresses within Cloudflare ASes and Cloudflare associated WHOIS information allow satellite to cluster these services as one entity with more certainty than the less obviously related Akamai customers.

Network interference: The question of "who is blocking what?" can be answered by finding ASes where a majority of resolutions have low `IPTrust` for a given domain. There are actually several ways in which an AS can deviate, which correspond to different forms of interference. For example, Iran regularly sends `thepiratebay.se` to `10.10.34.36`, and we see `IPTrust` of 6.6×10^{-9} for those resolutions, since the IP is also seen for a number of other blocked domains which don't otherwise overlap.

To extract instances of interference that are reflected in the `IPTrust` metric, we look at the distribution of values for resolutions at the AS level. When the distribution for an AS is depressed in a statistically significant manner (we currently look for a mean 4 standard deviations below the overall distribution) we consider the AS-domain pair to be 'suspicious'.

There are several types of interference which can all be easily distinguished from normal behavior, but which require special identification. We handle these through a decision tree, which provides a conservative estimate of known forms of interference. Crucially, this approach benefits from the fact that we are able to point to the mechanism which triggers each flagging. The categories we classify as interference are:

1. Too few resolutions or too many unparsable responses are received.

2. A domain which is otherwise 'single-homed' (meaning a single IP address is found regardless of client location) resolves to non-standard locations.

3. A domain with an otherwise 'dominant' AS resolves to many ASes.

4. Resolution deviates from an expected CDN cluster.

Instances of interference are accounted to occur because of the first of these classes which is applicable. All of these classes can be inferred from the already computed resolution data. Our initial AS-level aggregation allows us to directly find invalid or suppressed resolutions. We showed in Figure 5 that the majority of the most popular domains are single-homed, which we use for the third and fourth decisions. Finally, for domains which appear to be hosted on shared infrastructure, we use the `IPTrust` score computed above. When resolutions deviate from the expected CDN footprint, we are able to include automatic analysis of the availability of the most high-profile instances of interference.

3 Implementation

3.1 Assembly of domain list

To understand how sites behave, we must first know the sites we are interested in monitoring. It is unrealistic to monitor all domains on the Internet, since there are technically an infinite number of registered domains due to the dynamic nature of sub-domain resolution. Without a priori knowledge of CDNs and their expected IPs around the world, we need to monitor a representative set of domains to organically learn that knowledge.

We accomplish this goal by targeting the top 10,000 worldwide domains as measured by Alexa[2]. All of these domains receive high amounts of traffic. The least popular, `qualcomm.com`, is estimated to receive over 10,000 visitors per day. While not a perfect list, 10,000 domains contains the diversity needed to organically discover important CDNs. Looking at the smaller Alexa top 1k domain subset, we find that under a quarter of the domains we cluster into CDNs are listed. For services like CloudFlare, which partition their IP space across different domains, our clustering algorithm would be overly cautious without access to an appropriately diverse sample set.

We make HTTP requests to each domain, since we find that many bare domains (e.g. `expedia.com`) redirect to a prefixed domain (e.g. `www.expedia.com`), which are served on different infrastructure. When we detect such redirections, we include both the bare and prefixed domains in subsequent steps. We observe these redirects in roughly one fourth of monitored domains.

3.2 Active DNS measurement

Our goal in Satellite is to provide a tool for longitudinal mapping of the accessibility and distribution of web entities. To quickly detect updates and policy changes, we must constrain the amount of time we are willing to allow probing to run. Given the goal of weekly measurements of 10,000 domains from a single host, we request each domain from 1/10th (or roughly 150,000) of discovered DNS vantage points, maintaining geographic diversity while spreading network load across available hosts. This results in a measurement period of roughly 48 hours at a probe rate of 50,000 packets per second. We find our measurement machine to be CPU limited at about 100,000 packets per second. Unlike a typical zmap scan, our resolution probes have a high response rate, which results in significant CPU processing work.

Our probing is accomplished by extending zmap with a custom 'udp_multi' mode, where hosts are sent one of several packets. The packet sent is chosen based on the destination IP address only, resulting in a stable set of requests across measurement sessions — the same resolvers will receive the same queries each week. This approach was chosen for efficiency, multiple scanning processes and accompanying pcap filters increased CPU load and resulted in dropped packets. Instead, we found this extension to be a conceptually simple and efficient extension to the existing zmap tool.

The result of a 48 hour collection process is a 350GB directory containing tuples of resolver IPs, queried domain, time-stamp, and received UDP response. We record the full packet responses we receive, under the assumption that in the future we may find other fields of the DNS responses to be of interest. The raw format of base-64 encoded packets is extremely verbose, but since the response packets for each domain are largely the same, a full run can be compressed to 20GB. By taking this relatively easy step of compression, our measurement machine has had no trouble storing our year of collection results.

3.3 Supplemental data collection

There are several pieces of supplemental data that are valuable to Satellite in understanding the measurements we conduct. For IP addresses of interest, we collect information to improve our ability to map IPs back to their controlling organizations. For these organizations and the IPs they control we also use supplemental information to understand which geographic points of presence are used. While our measurements do not rely on our ability to understand these associations, downstream analysis can benefit from them.

3.3.1 IP Metadata

We retrieve meta-data on resolved addresses to better understand what organizations they belong to, and whether two addresses are likely to be equivalent. The two signals we have found valuable to include in this process are the reverse PTR records for the addresses, and the WHOIS organization entry controlling the address. Reverse PTR records are contained in the 'in-addr.arpa.' pseudo-tld in the DNS hierarchy. They are maintained by the organizations controlling the IP address, and often provide

a canonical name when the IP belongs to a known service. The WHOIS database is a database of IP ownership maintained by IANA and its delegates that contains organizational responsibility, in the form of technical and abuse contacts, for addresses.

We perform direct lookups for both the PTR and WHOIS organizational contacts for all distinct IP addresses resolved. We then perform a clustering of each data set: All IPs with the same WHOIS organization are clustered into a WHOIS cluster, and all IPs with consistent PTR records are clustered together. To cluster PTR records, we use a simple heuristic: if all but the final dot-separated section of the returned records are equal, we put the IPs in the same cluster. For instance, a west coast resolution of `apple.com` has the PTR record of `a23-200-221-15.deploy.static.akamaitechnologies.com`, while an east coast resolver sees `a23-193-190-30.deploy.static.akamaitechnologies.com`. Since both cases end with `deploy.static.akamaitechnologies.com`, they are clustered together as part of the same entity.

3.3.2 Geolocation

During our collection and aggregation process we maintain a network, rather than geographical, view of the data. We prefer aggregation at a Class-C address level, which reduces calculations without losing precision or mixing IPs owned by different entities. Our other form of aggregation is on the AS level, to represent the aggregations of IPs which will see a similar view of the rest of the Internet. ASes represent business relationship which exist, and the AS which 'owns' an IP range is responsible for managing abuse and routing of packets for those IPs. As such, even when a sub-range is delegated, we assume the full AS experiences a consistent routing policy. There are always exceptions to such assumption: the Comcast AS contains clients on both of the east and west coast of the US, who will reach different data centers of many cloud services. Our use of AS aggregation will consider these results as a single combined data point. Likewise, the Google and Edgecast systems operate servers in many countries. When addresses in these ASes are used as resolvers, we consider them to be in the closest location to our measurement machine, the US.

For the visualization of infrastructure locations in the evaluation of this paper, we have to associate IPs with geographical locations. For this, we use three data sources: the country of registration for the *whois* point of contact (Used for AS location), the MaxMind [26] country-level database (Used for IP location), and the list of anycast prefixes from Cicalese et. al. [8]. When MaxMind geolocates different IPs within an AS to multiple countries, we use that list. Otherwise, we use the country of registration. Since MaxMind cannot handle geographic diversity

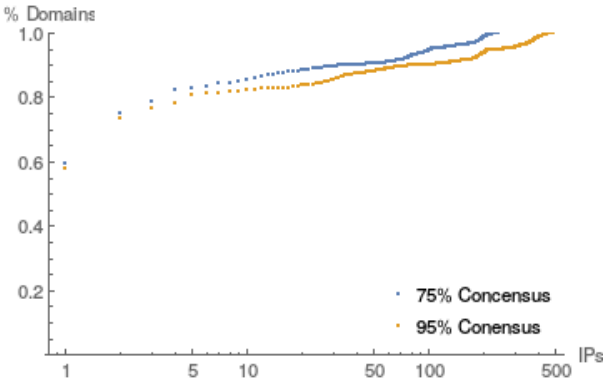

Figure 5: Number of IPs hosting different domains at two thresholds for dominant addresses. For 60% of domains, 1 IP accounts for 75% of all resolutions, and for 80% of domains, 10 IPs account for 95% of resolutions.

hidden by anycasting, we explicitly geolocate the points of presence of anycasting IPs, and use the closest point to a given resolver.

We find that 1% of distinct IPs resolved in a typical Satellite data match the anycast prefix list. To estimate the points of presence of these IPs, we measure latency from a range of vantage points, as in [22], resolving topology with [24]. In the future, we hope to learn these latencies through the DNS requests we already make using the technique in [17]. We find that since the CDNs we are identifying are highly distributed, we end up with observations which are either very small latencies indicating a point of presence near the vantage point, or are large enough to not impose additional constraints.

While these geographic heuristics are not infallible, they are largely accurate at the country level [32, 31]. As such, they provide a grounding for initial data exploration. When considering specific interference or deployment situations, it remains important to identify the relevant subsets of data. For instance, when we consider Iran in Figure 12, we manually limit our analysis to ASes of known ISPs in the country. We hope that a verifiable geographic database of infrastructure sourced from open measurements becomes available.

3.4 Aggregation

To support interactive exploration and analysis of collected data, satellite automatically aggregates the observed responses of each weekly collection. This automatic processing also materializes several views of the aggregated data which are used in subsequent analysis.

This automatic process attempts to parse each received packet as a DNS response, validates that it is a well-formed DNS response, and records the IP addresses returned. We tabulate these values for each resolver AS and domain. The resulting mapping is roughly 3 GB, and is used as the basis of subsequent processing. The 100-fold reduction comes from stripping the formatting and other

fields of DNS responses, and from aggregating responses by resolver AS. Scanning this file to calculate basic statistics takes under 5 minutes on a single 2.5GHZ core of our lab machines, and the format lends itself to parallel execution when more complex tasks are needed.

In addition to initial aggregation, we automatically build lookup tables for the set of IPs which have been resolved for each domain, and the total set of IPs seen as resolution answers. We also calculate the set of domains associated with each IP to facilitate reverse lookups of other domains potentially co-hosted on an IP. On a recent execution of Satellite, we saw a total of 5,337,315 distinct IPs resolved, located within 6,742 distinct ASes.

The domain resolutions we have collected already provide insight into the inner workings of popular websites. In Figure 5, we show how much diversity we found in the responses for each domain globally. If almost all responses return a single IP address, we can make the inference that the dominant IP is the canonical server for the domain. In other words, the domain is 'single homed'. In our monitored domains, we see this behavior in 60% of domain, the far left data points in the graph. Slightly further right in Figure 5 are domains which use simple load balancing schemes. We see that roughly 80% of domains have four or less 'dominant' IPs. This figure doesn't capture the use of anycast IP addresses, but does indicate that even for top domains, the majority have a single or small set of 'correct' addresses. The tail to the far right on the figure indicate domains which use geographically distributed infrastructure, and which require more complex analysis to determine whether individual resolutions are correct. For example, we record over 500 IP ranges for the `google.com` cluster, and over two hundred for many akamai hosted domains like `www.latimes.com`.

4 Evaluation

4.1 Address Validation

To validate our ranking and clustering algorithms, and our data collection process more generally, we make web requests to each resolved IP address as a potential location of each sampled domain. More specifically, we connect to each IP which has been seen as a candidate, and request the '/favicon.ico' file, using the domain as the 'Host' header. Slightly under half of the monitored domains have this file and can be validated in this way. We record hashes of all returned content, and compare these hashes against copies of the favicons fetched using local DNS resolution to determine whether an IP is correctly acting as a host for a given site.

Over a total of 965,522 completed resolutions, 82% of resolved IPs are deemed 'correct'. 5,479 domains are skipped in this validation, because no authoritative favicon is present, and validation is performed on the other 4,521. These domains are not used when we evaluate clustering performance.

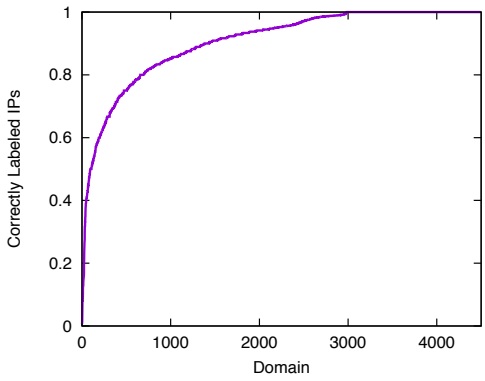

Figure 6: For each of the 4,500 domains with favicons, The fraction of distinct IPs resolved with a 'correct' `IPTrust` score over 0.5. Our automated classification matches favicon presence for over 90% of IP-domain pairs.

In Figure 6, we show the agreement between this validation process and the confidence scores for IPs used in our clustering algorithm. We treat an `IPTrust` score of 0.5 as trusted, but find similar results for other thresholds due to the polarization of trust scores by the iterative calculation. While there is noticeable divergence between the IPs in our scorings and the favicon results, over 95% of those failures are false-negatives (our algorithm was overly conservative in creation of clusters, and gives low scores to IPs the favicon process showed to be correct). The vast majority of these occur in situations where a single partition of IPs is normally resolved for a domain, but other IPs are also able to respond correctly when queried. Both Akamai and CloudFlare exhibit this behavior. Partial aggregation of these clusters has a minor effect on this view, since when domains are fully partitioned onto separate IPs we only consider our trust of those IPs we've actually seen resolved.

This validation technique is susceptible to manipulation by an adversary which returns the correct favicon image on an otherwise malicious server. We are not aware of any block pages behaving in this way.

In principle, validations like the use of favicons or signals like reverse DNS lookups can also be used in the clustering process to further refine which IPs are believed 'correct' for domains. To us though, this result shows that the DNS resolutions themselves are able to produce largely reliable mappings of CDN IP addresses.

We can also validate our clustering algorithms against the ground-truth of IP prefixes advertised by some CDN providers. For this validation, we consider the Fastly CDN, which uses a compact set of prefixes maintained at `https://api.fastly.com/public-ip-list`. We find that all 12 IP prefixes found by Satellite as the Fastly CDN cluster are included in the officially advertised list. The Satellite cluster con-

CDN	IP Space	Clustered ASes
CloudFlare	107008	75
Akamai	264960	489
Google	476416	1036
Cloudfront	128512	21
Incapsula	12288	17
Fastly	8192	17
Dyn	2304	9
Edgecast	24832	65
Automattic	3584	5
AliCloud	41728	42

Table 4: IPs in each of the ten largest shared infrastructure platforms. Variance in size between Dyn, Fastly, Automattic and the others is due to use of Anycast. Some ASes are significantly undercounted by clustering, Akamai has points of presence in over 1,000 ASes.

tains $72/80$ domains found using this ground truth list of IP prefixes. For geolocation, the MaxMind database reports multiple locations, accounting for 5 of the 10 Fastly countries, including the US, Australia, and three of four locations in Europe (mistaking Germany for France). The Australian class-c network prefix is identified as anycasting, which we resolve to 4 of the 5 additional locations – New Zealand, Japan, Hong Kong, and Singapore – agreeing with the results of [8]. These two techniques lead us to correctly find 8 of the 10 locations, missing Brazil and mistaking Germany for France.

4.2 Website Points of Presence

While we have shown in this paper that the Satellite technique is able to accurately map the IPs which are operated by targeted websites, we have not yet shown the implications of that data. Here, we attempt to characterize the dominant content distribution entities in the Internet today, and provide some insight into where they operate and the international nature of the Internet today.

In Table 4, we show the IP space we estimate for the largest CDN clusters. These platform each have unique network structures, and use a range of technologies including rotating IPs and anycast, which make it difficult to directly compare scale from these numbers. For instance, most Google IPs resolve to IPs within Google's own AS, while IPs from Akamai are largely resolved to IPs located in the ASes of consumer ISPs.

In Figure 7, we use the geolocation of ASes to count which countries these providers are located within. One striking feature of this geolocation exercise is to note that the 10 largest content distribution networks use IP addresses allocated to ASes registered in at least 145 countries. We trust MaxMind for these locations, but attempt to be conservative, including neither anycast resolution nor clustering the true extent of partitioned providers like Akamai. This undercounting is reflected in Table 4, which indicates the primary cluster we use for Akamai accounts for under half of the over 1,000 ASes they report [1].

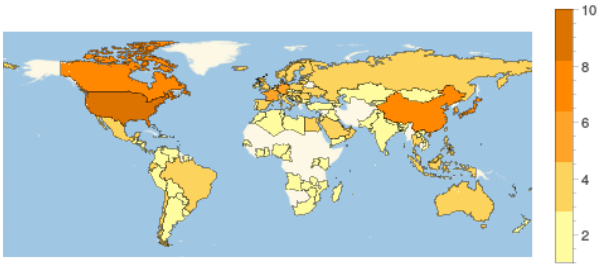

Figure 7: Points of presence of the CDNs from Table 3. Anycast is not included, indicating conservative counts.

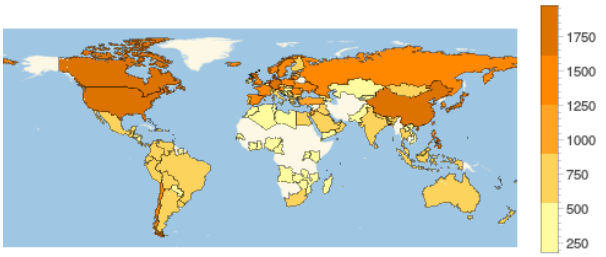

Figure 8: Number of sites resolved locally in each country.

In Figure 8, we plot how many domains are resolved within each country. We see at least 18% of all domains (2325) resolving to an in-country IP address for resolvers in China, while other countries like Mexico resolves only 5% (559) of domains locally. This view of domain locality can be used to understand which publishers have complied with local regulations, and to track how much Internet traffic will transit international links.

4.3 Interference

Our confidence scoring of how well IPs represent domains helps us address an ongoing pain point in interference measurement: how to know if a returned IP address is 'correct'. The primary issue in this determination traditionally has been whether an IP that is not the same as the canonical resolution is a CDN mirror or an incorrect response. Using CDN footprints along with more simple heuristics for single-homed domains allow us to identify instances of inaccessibility with higher confidence.

We measure interference through positive identification of the four categories in 2.4.2. These categories are conservative, but remain valid for not fully clustered CDNs.

Figure 9 shows the number of largely inaccessible domains found in a single snapshot of collected data. We find at least 5 of the monitored domains to be inaccessible in at least one Autonomous System in over 78 countries.

We then divide the instances of observed interference across other factors. Figure 10 shows a comparison of interference for sites on CDN infrastructure versus those which are single-homed. While roughly 80% of sites are single homed, we see as much interference is directed

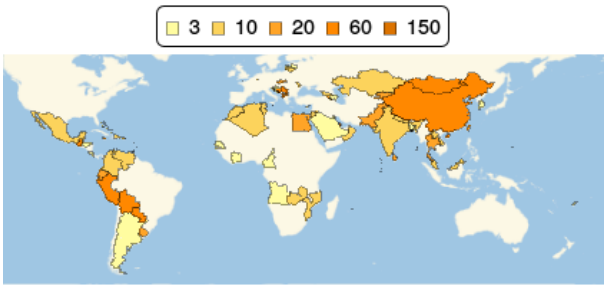

Figure 9: Number of domains inaccessible in each country.

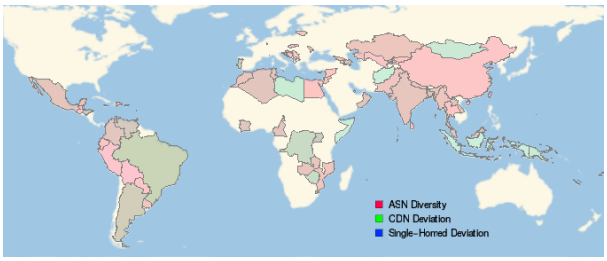

Figure 10: Types of interference by country. Anomalies are geographic, with some regions like China providing a diversity of false IP addresses, while others like Libya using a single block page. There are no occurrences of only 'CDN Deviation', or 'Single-Homed Deviation.' The relative shades indicate the mixture of the different categories present in each country.

at distributed sites, perhaps due to their popularity. This indicates that naive approaches have been missing a significant fraction of total interference instances.

It is possible for a censor to mask their interference from Satellite. Injecting DNS responses using a system of the type known to be in use by China could be targeted to miss an external observer, by only responding to requests originating within the Country or responding correctly to external queries. While much less visible to Satellite, these forms of interference would themselves be visible, and could even be less effective internally. The switch to other techniques like IP or keyword-based blocking would also not be visible in the current DNS data set.

4.4 Broader Implications

Our stated purpose in building Satellite and collecting data on the presence and accessibility of popular sites was to allow for new insights into the changing structure of the internet. What are those insights? Many of the implications are inextricably tied to real world events and politics, and reflect on the censorship practices and business environments of nation states. While we aren't comfortable claiming to understand these sociopolitical structures without accompanying real-world evidence, we can show value in the data in light of the larger trends occurring in Internet Governance.

In Figure 11 we show the delta of how many more domains are resolved within each country compared to six months prior, based on location of IPs with trust above 0.5

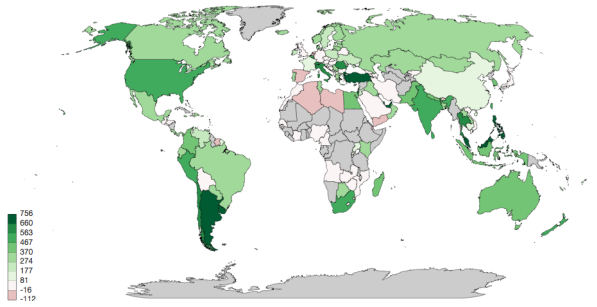

Figure 11: How many more sites resolve locally (to IPs within the country) in September 2015 compared to 6 months prior. This figure is based on a dataset of 8,800 domains which remained in the top 10,000 list at both sample points.

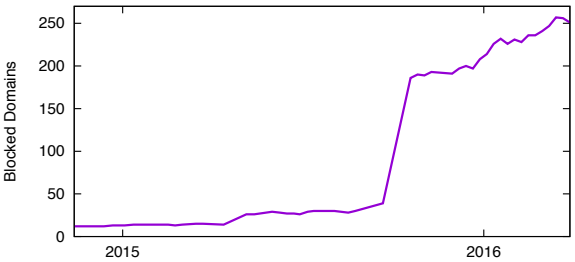

Figure 12: Number of domains detected to have anomalous resolutions in Iran since late 2014. An interactive version is at `http://satellite.cs.washington.edu/iran/`.

on a per-domain basis. What this shows for each country is how many new domains are now resolved internally where previously they would have been resolved to international servers. This shows the expansion of CDN infrastructure, but also an increasing ability of governments to regulate access within their national territories [10].

In Figure 12 we show the number of domains which are detected to have anomalous resolution across Iranian ISPs. We see a spike in the second half of 2015, which correlates with statements from the authorities there that they were beginning a second phase of filtering. More recently, Satellite has recorded additional inaccessible domains in the lead up to February 2016 elections.

5 Related Work

The active probing techniques used by Satellite build upon a long history of Internet measurement [24, 7]. The subsequent analysis of connectivity data has been tackled by previous generations of censorship measurement systems, though Satellite differs in the breadth of the measurements it aggregates and the way it handles noisy data.

Active scanning of the Internet has been used to measure important properties of ISPs, and has been shown to reasonably map individual CDNs [18, 6]. In particular, the rate of churn of DHCP reservations within consumer ISPs [27] has been estimated and the presence of Bluecoat DPI boxes [25] has been detected using active measure-

ment techniques. Active probing was used for the Internet census characterization of scale [3] and more generally in the web security space to measure the uptake of software updates and vulnerabilities [33, 12]. It has not yet to the best of our knowledge been used to independently measure the footprints of CDNs or longitudinal ISP-level interposition on traffic.

What to Measure: Determining domains of interest is by itself a tough problem. There are many billions of DNS records in use on the Internet [4, 14], and there are obvious deficiencies with the coverage or representativeness of lists of top sites. Previous measurement studies have used either top domains as reported by a neutral providers like Alexa [2], or more targeted lists they hand curate [15]. One of the most popular lists for censorship work is the list of sensitive domains maintained by Citizen Lab [30]. Satellite needs to measure a set of sites which reveal the shared infrastructure of CDNs, and we choose the Alexa top 10,000 domains as our base measurement set in order to achieve that coverage.

How to Measure: Researchers have invested considerable effort in the measurement of network interference, both by using participants within target networks [15, 20, 29] and through purely external mechanisms [9, 34]. DNS has been a measurement focus, largely because it is a commonly manipulated and unsecured protocol. DNS reflection against remote open resolvers has also been proposed for censorship measurement [36] as early as 2006. What we continue to lack is a system which is able to sustainably measure and act as a data repository for these measurements across both countries and time. Ripe Atlas [29] offers shared access to its distributed deployments of probes, but limits the types of measurements and rate-limits measurements such that regular probing of diverse domains by the deployment would require ownership of a significant fraction of the network.

Determining Site Presence: While determining which sites are of interest is hard, determining whether a given IP is a valid host for a site can be even harder. In their investigation of CDNs in 2008, Huang et. al [18] arrive at a similarly sized list of open resolvers as Satellite (280,000), and use them to map the Akamai CDN. They create their list of resolvers starting from DNS servers observed by Microsoft video clients, rather than direct probing. Specific CDNs like Google have also been crawled through the use of EDNS queries to simulate the presence of geographically diverse clients [6], but this is only possible for a small subset of deployments which support EDNS for redirection. Research focusing on censorship, like the analysis of ONI data [16], have used AS diversity to determine if IPs are valid for a domain, but do not explicitly consider CDN behavior.

There are also many commercial sites which offer traffic information for web sites. We know that some of this data is crowd-sourced through browser plugins, while other portions come from automatic robot crawling. For instance, the Alexa rankings are based off of a browser plugin which monitors the browsing habits of a small number of participating users. Some sites also show which sites run on identical IP addresses [19]. In practice we find that these systems appear to do direct lookups of IPs, since geographical distribution is not surfaced. They also do not appear to do significant identification of CDN IP spaces, since CDN'ed sites are not fully aggregated.

Determining Abnormal Behavior: Categorizing responses as normal or abnormal have typically been performed through the use of heuristics in how the response may deviate from expected behavior. This is true for both determining trust in a DNS response, and determining if a given connection is working as expected. These heuristics include metadata like the AS and reverse PTR record of the IP [16], behavior of HTTP queries to the server [21], and considering the aggregate prevalence of a given response [15]. More recent work has explored the use of aggregate statistical behavior to determine when network level behavior has changed [37]. These techniques provide valuable direction for Satellite, though there is not yet a comprehensive set of best practices for determining self-consistency and anomalies in our data set.

6 Conclusion

Satellite is already a valuable system for measurement of both CDNs and prevalence of interference. Our continued development efforts are focused on: (1) Improved reproducibility of geographic determination. (2) Developing an interactive visualization for interacting with data. (3) Integration of additional probing mechanisms for measurement of transport and IP level connectivity.

In this paper we have presented Satellite, a system for measuring web infrastructure deployments and availability from a single external vantage point. By lowering the bar for collecting, aggregating, and understanding we make this data much more accessible. Satellite the growing predominance of CDNs in the top Alexa domains. The same data shows evidence of growing interference of domain resolutions around the world. Satellite is a fully open platform, and both the data and code are available online at `satellite.cs.washington.edu`.

7 Acknowledgments

Special thanks to Sidney Berg, Adam Lerner, and the collaborators who have greatly improved Satellite. Thanks to Noa Zilberman for valuable insights in shepherding this paper to publication. This work is sponsored in part by the Open Technology Fund's Information Controls Fellowship Program, Google Research Award, and National Science Foundation (CNS-1318396 and CNS-1420703).

References

[1] Akamai. The akamai intelligent platform, 2016. akamai.com.

[2] Alexa, 1996. www.alexa.com.

[3] Anonymous. Internet census 2012, 2012. internetcensus2012.bitbucket.org.

[4] Anonymous. DNS census 2013, 2013. dnscensus2013.neocities.org.

[5] Anonymous. Towards a comprehensive picture of the great firewall's dns censorship. In *FOCI*. USENIX, 2014.

[6] M. Calder, X. Fan, Z. Hu, E. Katz-Bassett, J. Heidemann, and R. Govindan. Mapping the Expansion of Google's Serving Infrastructure. In *IMC*. ACM, 2013.

[7] B. Cheswick, H. Burch, and S. Branigan. Mapping and visualizing the internet. In *USENIX ATC*, 2000.

[8] D. Cicalese, D. Joumblatt, D. Rossi, M.-O. Buob, J. Augé, and T. Friedman. A fistful of pings: Accurate and lightweight anycast enumeration and geolocation. In *Computer Communications (INFOCOM)*, 2015.

[9] J. R. Crandall, D. Zinn, M. Byrd, E. Barr, and R. East. Conceptdoppler: A weather tracker for internet censorship. In *CCS*. ACM, 2007.

[10] R. Deibert and R. Rohozinski. Liberation vs. control: The future of cyberspace. *Journal of Democracy*, 2010.

[11] D. Dittrich, E. Kenneally, et al. The Menlo Report: Ethical principles guiding information and communication technology research. *US Department of Homeland Security*, 2011.

[12] Z. Durumeric, E. Wustrow, and J. A. Halderman. Zmap: Fast internet-wide scanning and its security applications. In *USENIX Security*, 2013.

[13] M. Elsner and W. Schudy. Bounding and comparing methods for correlation clustering beyond ILP. In *ILP-NLP*, pages 19–27, 2009.

[14] Farsight Security, Inc. Farsight DNSDB, 2010. dnsdb.info.

[15] A. Filastò and J. Appelbaum. OONI: Open observatory of network interference. In *FOCI*. USENIX, 2012.

[16] P. Gill, M. Crete-Nishihata, J. Dalek, S. Goldberg, A. Senft, and G. Wiseman. Characterizing web censorship worldwide: Another look at the opennet initiative data. *ACM Transactions on the Web*, 2015.

[17] K. P. Gummadi, S. Saroiu, and S. D. Gribble. King: Estimating latency between arbitrary internet end hosts. In *IMC*. ACM, 2002.

[18] C. Huang, A. Wang, J. Li, and K. W. Ross. Measuring and evaluating large-scale cdns. In *IMC*. ACM, 2008.

[19] HypeStat, 2011. hypestat.com.

[20] ICLab, 2013. iclab.org.

[21] B. Jones, T.-W. Lee, N. Feamster, and P. Gill. Automated detection and fingerprinting of censorship block pages. In *IMC*. ACM, 2014.

[22] E. Katz-Bassett, J. P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe. Towards ip geolocation using delay and topology measurements. In *SIGCOMM*. ACM, 2006.

[23] J. M. Kleinberg, R. Kumar, P. Raghavan, S. Rajagopalan, and A. S. Tomkins. The web as a graph: measurements, models, and methods. In *Computing and combinatorics*. Springer, 1999.

[24] H. V. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, and A. Venkataramani. iPlane: An information plane for distributed services. In *OSDI*. USENIX, 2006.

[25] M. Marquis-Boire, J. Dalek, S. McKune, M. Carrieri, M. Crete-Nishihata, R. Deibert, S. O. Khan, H. Noman, J. Scott-Railton, and G. Wiseman. Planet Blue Coat: Mapping global censorship and surveillance tools. 2013. citizenlab.org.

[26] MaxMind. Geoip, 2006. maxmind.com.

[27] G. Moreira Moura, C. Ganan, Q. Lone, P. Poursaied, H. Asghari, and M. Van Eeten. How dynamic is the ISPs address space? towards Internet-wide DHCP churn estimation. In *IFIP Networking*. IEEE, 2015.

[28] J. Muach. Open resolver project, 2013. openresolverproject.org.

[29] R. NCC. Ripe atlas, 2010. atlas.ripe.net.

[30] OpenNet initiative, 2011. opennet.net.

[31] I. Poese, S. Uhlig, M. A. Kaafar, B. Donnet, and B. Gueye. Ip geolocation databases: Unreliable? In *SIGCOMM CCR*. ACM, 2011.

[32] Y. Shavitt and N. Zilberman. A geolocation databases study. *Selected Areas in Communications, IEEE Journal on*, 2011.

[33] Shodan. shodan, 2013. `shodan.io`.

[34] J.-P. Verkamp and M. Gupta. Inferring mechanics of web censorship around the world. In *FOCI*. USENIX, 2012.

[35] P. Vixie. Extension mechanisms for DNS (EDNS0). RFC 2671, 1999.

[36] S. Wolfgarten. Investigating large-scale Internet content filtering. Master's thesis, Dublin City University, Ireland, 2006.

[37] J. Wright, A. Darer, and O. Farnan. Detecting internet filtering from geographic time series. *arXiv preprint arXiv:1507.05819*, 2015.

[38] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldszmidt, and T. Wobber. How dynamic are ip addresses? In *SIGCOMM CCR*. ACM, 2007.