Satellite: Observations of the Internet's Stars

Will Scott, Sidney Berg, and Arvind Krishnamurthy
{wrs, sidberg, arvind}@cs.washington.edu
University of Washington. Tech Report UW-CSE-2015-06-02

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 end hosts. 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 more organizations. We demonstrate a clustering procedure which accurately captures the IP footprints of CDN deployments, and then show how this stepping-stone allows for more accurate determination of correct and incorrect IP resolutions. Satellite 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 4819 instances of ISP level DNS hijacking in 117 countries.

1 Introduction

Even after several generations of elaborate measurement platforms, it remains difficult to perform an in-depth analysis of how the web content is distributed or even to understand the extent to which web access is open and unfettered. This lack of understanding emerges in the questions we cannot easily answer: Which countries have servers operated by Google or Microsoft? Which sites are powered by various content distribution networks (CDNs) such as Akamai or CloudFlare? Which ISPs run caching proxies or other stateful middleboxes? Which websites have degraded availability due to network interference? And so on.

While we may have some understanding of what to measure to answer these questions, there is no existing data set or measurement platform that can aid us in answering these questions. In fact, there are many challenges to both assembling the measurement data and analyzing it to characterize the current state of web content distribution. First, we would need measurements from globally distributed vantage points in order to characterize global website accessibility. While there has been some limited success at crowd-sourcing measurements and/or taking advantage of a globally distributed measurement platform such as PlanetLab, the research community has not yet been able to collect a data-set that is representative of a large fraction of countries and ASes. Second, since the deployment and accessibility characteristics vary significantly across websites and time, the data-sets should be collected at a fine-grained and timely manner. Third, the analysis of how websites employ CDNs and the identification of network interference are interrelated and have to be tackled jointly in order to obtain an accurate characterization. For example, when ISPs block websites by redirecting them to a block page, that server is easily misconstrued as a CDN node for that geographical region. Conversely, websites served through globally distributed CDNs can be confused with willful redirection of traffic by a local ISP. We need to determine the expected IPs of CDN deployments in order to characterize the abnormalities that are interference.

In this paper, we present a measurement tool chain and analysis system called *Satellite* that is designed to address the above challenges. First, we address the need for global measurements by developing a measurement system that uses 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.

Second, we build a measurement tool chain and infrastructure that performs these measurements in a timely manner and validates them to eliminate noisy data. Internet scanning from a single end-host is less complex or expensive than the deployment of a measurement platform, but it remains time-consuming and bandwidth intensive. The validation challenges for our data are similar to those faced by distributed measurement systems. Probing from a single vantage point is susceptible to the manipulation of results - a domain or ISP can respond differently to known measurement machines than to other traffic. We address these issues in the implementation of Satellite.

Third, we analyze the serving infrastructure for top Internet sites, and show that this understanding is inherently interlinked with an understanding of network interference. We present an automatic clustering process that is able to identify CDN infrastructure from the collected data. The responses which do not fit expected clusters are also interesting, because they allow us to identify interference without worrying about mis-classifying CDNs. Through this analysis we are able to both monitor the growth of shared hosting platforms, and trends in network-level interference which cause sites to become inaccessible.

Satellite is a public project consisting of the code for data collection and analysis, a growing 9 month repository of collected data, and derived presentations of site structure and interference. The focus on public release reduces the need for others to trust us in the interpretation of data. Instead, Satellite is designed to be operated by multiple interested organizations - allowing for independent auditing and confirmation of collected data. This also improves our confidence in the sustainability of the project, and our ability to amass a longitudinal data set of changing Internet behavior.

From interpretation of Satellite data, we are able to correlate the addresses of domains across ISPs to 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.

We now summarize the contributions of Satellite:

- A software platform for mapping CDNs, middleboxes, and DNS consistency from a single end host.
- Data on the reachability and routes to 10,000 popular domains over the last year.
- A method for inferring ISP level behavior by aggregating responses of individual hosts.

2 Background and Related Work

The active probing techniques used by Satellite build upon a long history of Internet measurements. The subsequent analysis of connectivity data has been tackled by previous generations of censorship measurement systems, though it differs in the breadth of the measurements and the ability to handle noisy data.

Active scanning of the Internet has been used to measure important properties of ISPs already, and has been shown to reasonably map individual CDNs [13, 5]. In particular, the rate of churn of DHCP reservations within consumer ISPs [18] has been estimated and the presence of Bluecoat DPI boxes [17] has been detected with this technique. 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 [21, 8]. 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, 10], forcing the choice of some sample. Previous measurement studies have used either top domains as reported by a neutral provider like Alexa [2], or used a more targeted list [11]. One of the most popular lists for censorship work is the list of sensitive domains maintained by Citizen Lab [20].

How to Measure

Researchers have invested considerable effort in the measurement of network interference, both by using participants within target networks [11, 15] and through purely external mechanisms [6, 22]. 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 [24], but we still don't publicly collected data for analysis across both countries and time.

Determining Site Presence

While determining which sites are of interest is hard, determining whether a given IP is a valid host for a site is even harder. In their investigation of CDNs in 2008, Huang et. al [13] arrive at a similarly sized list of open resolvers as Satellite (280,000), and use them to map a specific 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 [5]. Research focusing on censorship, like the analysis of Open Network Observatory data [12], have used ASN diversity to determine if IPs are valid for a domain but have not used CDN footprints.

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 [14]. 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.

3 Design

Satellite uses a single scheduler to measure and analyze data. This process manages both the data collection and subsequent aggregation and analysis and is designed as a weekly cron

job. In broad strokes, the measurement framework performs the following steps:

- **Domain Determination** Selects candidate domains to measure, and resolves their true names.
- **Probing** Scans the Internet for active DNS resolvers and web servers, and calculate which to target for primary measurement.
- **Resolution** Measures candidate domains against discovered resolvers and hosts.
- **Aggregation** Aggregates measured data for efficient processing.
- **Clustering** Calculates fixed-points of CDN clusters through a repeated scoring of resolution likelihood.
- **Extraction** Recovers the IPs belonging to CDNs, along with ASN level divergence from expected behavior.

In the remainder of this section we explain the procedure of each step, and provide insight into the collection and processing of data.

3.1 Domain Determination

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`[1], is estimated to receive over 10,000 visitors per day.

In our selection process, we make HTTP requests to each of these domains, since many of the bare domains (e.g. `expedia.com`) statically redirect to a primary domain (e.g. `www.expedia.com`), which can be served differently. When we detect these redirections, we include both the bare and prefixed domains in subsequent steps. This occurs for roughly one forth of monitored domains.

3.2 Probing

Our measurements are based on gathering data on how domains behave for different clients around the world. There are several options available for such collection. Traditionally, researchers have used cooperating hosts in a variety of networks[16, 20]. More recently, the EDNS extension has allowed clients to indicate that they are recursively resolving on behalf of another to better resolve responses.[23, 5] While very few domains support EDNS, we can take advantage of the same behavior that it is designed to fix. By making requests to many different resolvers, we can learn the different points of presence for target domains. For instance,

[1]In April, 2015

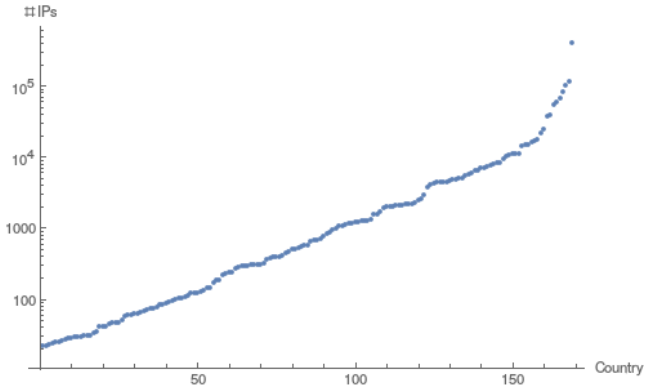

Figure 1: DNS servers discovered in each Country. We find over 5000 ASNs and 169 countries host more than 20 distinct DNS resolvers.

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.

We enumerate IPs serving as DNS resolvers, by probing the IPv4 Address space with zmap. Of the 32 million open DNS servers recorded by the Open Resolver Project [19], we find roughly 12 million respond to requests reliably. Of these, we find that 7 million servers across 1.5 million class-c (/24) networks respond correctly and offer recursive resolution. These servers provide coverage of 20,000 ASNs, as shown in Figure 1. There are statistically valid numbers of servers active in 169 countries.

3.2.1 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 harm to these servers, and as such we do our best to minimize the impact we cause in our collection.

Open DNS resolvers are a well known phenomenon, and lists of active resolvers can already 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, we have not received any complaints related to our scanning or subsequent resolutions in the last quarter. We have never received a complaint due to overloading a DNS resolver, or the follow-up probes rather than the initial scan.

The Alexa top 10k, while not a perfect list, provides the diversity needed to organically discover important CDNs without the need for hard-coding. Looking at the smaller global 1,000 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 appropriate diversity.

We attempt to abide by the 7 harm mitigation principles for conducing Internet-wide scanning outlined by the zmap

project[8]. In particular, we (a) coordinated with the network administrators at our university to route complaints back to us, (b) ensured we would not overload the network, (c) host a web page explaining the measurements with 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 a month to reduce the chance that we will send queries to non-official recursive resolvers. This reduces our server set by 16%[2]. Measurements in IP churn indicate that the bulk of dynamic IPs turn over to subsequent users on the order of hours to days.[25]

3.3 Resolution

Our goal in Satellite is to provide a tool for longitudinal mapping of the accessibility and distribution of web entities. To regularly detect updates and modifications, 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 find we must sample each domain across 1/10th (or roughly 200,000) of discovered DNS resolvers. This results in a resolution time of roughly 48 hours at a probe rate of 50,000 packets per second.

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. Since we only query known reliable DNS servers, we have a high response rate on scans, shifting the bottleneck in scanning to the processing of incoming packets.

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.

3.4 Aggregation

Our goal in aggregation is twofold. First, we hope to reduce the full amount of collected data to a format that is manageable for interactive interaction and analysis. Second, we hope to draw out patterns both between domains which share common resolution characteristics, and behavior between IPs within individual administrative domains.

[2]Specifically comparing the live resolvers discovered between March 20th and April 20th.

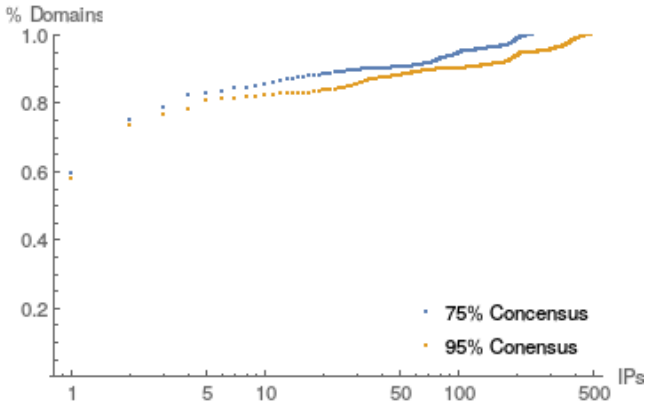

Figure 2: Number of IPs needed for consensus. For 60% of domains, 1 IP accounts for 75% of all resolutions, and for 80% of domains, 10 IPs capture 95% of resolutions.

To accomplish this, our first step is to interpret response data by attempting to parse each received packet as a DNS response, validating that it is a response to the query we expected to send, and saving the response A records. Through this process we build up a mapping from resolver ASN (Autonomous System, the level of an ISP) to resolved IPs for each domain. The resulting mapping is only 3 GB, and is used as the basis of subsequent processing. The 100-fold reduction comes from stripping the formatting and other fields of the DNS responses, and from aggregating responses by ASN. Scanning this file to map a basic function on the parsed JSON of each domain takes under 5 minutes on a single 2.5GHZ core of our lab machine, and the format is embarrassingly parallel if more complex task need to be distributed.

After this initial aggregation, we can quickly process and calculate additional views of data to aid our understanding. Each week, we automatically build lookup tables for all resolved IPs for each domain, and all domains resolving to each IP. We record 5,337,315 distinct IP resolutions for our sample of domains, located within 6,742 distinct ASNs.

The domain resolutions we have collected already provide insight into the inner workings of these popular websites. In Figure 2, we consider how much diversity is found in the responses for each domain. If almost all responses contain a single IP address, we can be pretty sure that the domain is 'single homed', and has a single address. In our monitored domains, we see this behavior in 60%, of resolutions, the far left data points in the graph. As we move right, we capture more domains which use a simple load balancing scheme, and find that roughly 80% of domains have four or less 'dominant' IPs. This view of domains doesn't capture the use of anycast IP addresses, but does indicate that even for top domains, the majority has a single or small set of 'correct' addresses. The remainder use geographically distributed infrastructure, requiring the use of more complex analysis.

3.5 Clustering

To find the points of presence of CDNs, we cluster domains in a three step process. First, we aggregate IPs to their first 3 octets for efficiency. Second, we compute a similarity metric between domains, based on how much of the time they resolve to the same destinations. Finally, we extract cliques within the similarity matrix, which represent the resolution patterns of shared Infrastructure.

3.5.1 Prefix Aggregation

We find many instances of CDNs which will load balance sites across one or many class-C prefixes. With 200,000 resolutions per domain, we notice significant performance improvements from first aggregating IPs before further comparison. For example, a typical Google domain is resolved to 6,200 distinct IPs, but only 1,300 distinct prefixes. In practice, this speeds up generation of the subsequent generation of a similarity matrix by a factor of 4 - to 1 hour instead of 4.

In addition, we find that for interactive manipulation of the data it is often useful to ignore the 'tail' IPs for faster speed. filtering the data to only IPs which account for at least 1% of resolutions for domains, we are able to reduce processing in the following steps to 5 minutes instead of an hour. This speed is efficient enough that we are able to tune subsequent process and see how algorithmic changes effect our final clusters much more rapidly. We do not use this stripping in our final calculations, since the additional time is needed to classify the less-common IP addresses as well.

While we gain performance from these aggregations, we must also justify their safety, and decide if they reduce the meaning of our analysis. In particular, consider the assumption that any IP within a Class-C prefix is valid for a site. We validate this assumption by looking at the BGP announcements for the prefixes observed, and find that 0% (a total of 37 IPs) are in a more precise than /24 resolution. This figure indicates that for just about all of these prefixes, the choice of IP is within the control of the site or CDN operator.

3.5.2 Domain Similarity

We compute domain similarity by measuring how many total resolutions of two domains are to the same prefixes. This can be thought of as the fraction of times domains resolve to an IP to which the other domain has also been resolve to. In fact, the resolutions of a domain are a vector of resolutions to different IPs, and our desired metric is the cosine distance between two such vectors. For intuition of why this is a reasonable approach, consider a single-homed domain, which resolves to one IP most of the time. We would like that IP to matter much more in comparisons with other domains, than IPs which the domain has resolved to only rarely. The "angle" of the vector in this case is in the direction of this primary IP, making a comparison of angles an appropriate metric.

We next use this similarity matrix to calculate a 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, weighted again by frequency of resolution. To score whether we believe that `google.com` resolves to `192.168.0.1`, we would look at other domains which have resolved to `192.168.0.1` and weight their similarities to `google.com` by how much of the time they resolve to `192.168.0.1`.

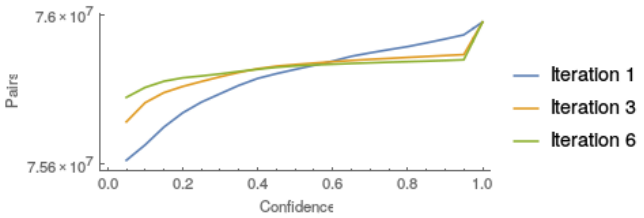

Figure 3: Domain similarities with and without repetition. An initial similarity calculation finds 25,000 edges with similarity above 95%. After 5 iterations of scoring, our process settles on 75,000 "strong" edges, and has stabilized with less than 1000 such edges added after the 6th iteration.

We then repeat the generation of our similarity matrix using the additional weighting of our IP confidence scores. We repeat this loop generation until we reach a fixed point, generally 5-6 iterations. Figure 3 shows the effect of recalculation on the distribution of domain similarity. Without iteration to the fixed point, we see many domains which have partial similarity. Through iterative weighting of IPs and domain similarities, we are able to focus on the emergent clusters to more clearly define sites which are co-hosted on shared infrastructure.

We can also perform a similar process at a country, or even AS level, to cluster sites which share infrastructure at more fine grained levels. This allows us to identify sites which use multiple infrastructure providers, for instance we note that `firefox.com.cn` is a generic CNAME to ChinaCache in most countries, but in China it resolves to IPs within the local ASNs.

This co-location of sites can interestingly also reflect somewhat on interference practices within ISPs. While the use of random or blocked IP resolution simply removes our confidence in the answer, the presence of a specific block site results in a clear cluster of the 'bad' domains within an administrative region. While this is not the focus of this study, we expect that the data can also prove useful for censorship analysis.

3.5.3 Clique Extraction

Finally, we calculate the individual domain clusters by finding strongly connected cliques within our similarity matrix. We perform this task by using the Vote/BOEM process recommended by [9], which considers the task of partitioning a data-set with full pairwise affinities in the context of document clustering. The process first performs an allocation of elements to create initial clusters, and then uses 'best one element moves' to search for better clusterings. This process

Domain	Rank
ebay.de	77
cntv.cn	79
indiatimes.com	110
dailymail.co.uk	114
etsy.com	149
cnet.com	151
deviantart.com	168
forbes.com	175

Figure 4: The highest ranked domains identified in the 'Akamai' cluster.

CDN	Size	Representative Domain
Cloudflare	803	life.com.tw
Akamai	391	ebay.de
Google	180	google.com
Cloudfront	131	kickstarter.com
Incapsula	122	www.juicyads.com
Fastly	90	www.adroll.com
Dyn	83	theverge.com
Edgecast	68	www.lemonde.fr
Automattic	50	time.com
AliCloud	44	www.163.com

Figure 5: Largest CDN clusters. The top 10 CDNs account for 20% of monitored domains.

created acceptable clusters, and is computationally efficient. Table 4 shows an example of the highest popularity sites which were clustered into a CDN which upon examination appears to represent the Akamai infrastructure. The largest clusters are shown in Table 5. We count the 10 largest shared hosting platforms hosting 1967 domains, making up almost 20% of those measured.

For most CDNs, our clique extraction process results in a single cluster representing the shared infrastructure. The notable exception is CloudFlare, which occupies 49 of the the roughly 200 clusters we create of four or more domains. This multiplicity is due to the use of 'ray's, where each domain hosted by the service is assigned to a partition with a specific allocation of IP addresses. This strict delineation results in a multiplicity of clusters. This is acceptable for our calculations of whether IP addresses are appropriate for a given domain, but sub-optimal for generating a single entry representing the entire Cloudflare footprint. For now, we hand-code the multiple clusters as Cloudflare, but we are actively working to integrate reverse-DNS lookups to perform this additional level of aggregation automatically. It is also noticable that Akamai, one of the largest CDN providers, is represented by a suspiciously low number of domains. We find that while Akamai transfers a large amount of traffic, their primary service choices account for this behavior. Akamai offers either bandwidth for large content items, which are served off of subdomains of akamai itself, so will not be given weighting in a count of domains. Their enterprise 'alta' solution works by routing customer IP blocks over the Akamai backbone, but those IPs will remain specific to individual sites and will not cluster with other Akamai customers.[1]

3.6 Interference Detection

Using resolution data in tandem with knowledge of CDN footprints and a confidence metric for IP resolutions, we are well prepared to detect instances of ISP level interference.

The main question, "who is blocking what?" can be answered by finding ASN outliers for domains. There are several ways in which an ASN could be an outlier, which correspond to different forms of interference. Some of these, like the lack of response to queries within an ASN, can be isolated as a unique event that Satellite data alone can characterize. Other methods, like redirection to a block page, are harder to differentiate from legitimate site behavior, like the use of a geographically-specific edge cache.

We first look for deviation from expected cluster behavior at an ASN level. This means looking at resolutions of a domain in an ASN, where the IPs for one domain have low scores, or where there are few resolutions while other domains are resolved at a normal level.

We then use a decision tree to classify deviations as 'suspicious'. We choose this approach because there are several approaches to interference which are known to be both in common use, and can be easily distinguished from normal behavior. By making decisions in this way we are able to provide a conservative estimate without worrying as much about the impact of false-positives, since there's a clear reason why each instance is classified as such. The categories we classify as interference are:

- Too few resolutions or too many invalid resolutions are received.
- A domain which is otherwise 'single-homed' (in the IP sense) resolves to non-standard locations.
- A domain with otherwise 'dominant' ASN resolves to many ASNs.
- Resolution is specific to the ASN, and deviates from expected CDN bounds.

All of these classes of interference can be inferred from the resolution data we have already computed. Our initial ASN-level aggregation allows us to directly find invalid or suppressed resolutions. We showed in Figure 2, that the majority even of the most popular domains are single-homed, which is used for the second two decisions. Finally, for detected CDNs, we have shown that we can determine the expected IP footprints of those clusters. When resolutions deviate from those expected footprints, we can make the final decision.

4 Evaluation

4.1 Address Validation

To validate our ranking and clustering algorithms, and our data collection process more generally, we make web re-

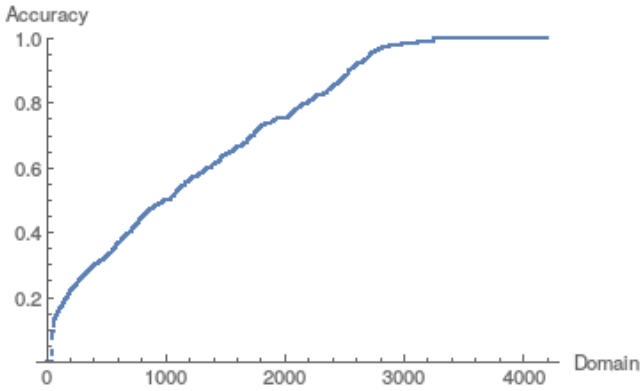

Figure 6: Validation Accuracy. For each of the 4,000 domains with favicons, we compute the score for how many of the IPs for that domain we correctly include as 'valid', compared to which IPs serve what we believe to be the correct favicon. 95% of the inaccuracy is false negative, indicative of our conservative clustering of CDNs like CloudFlare.

quests 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. 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 was found when we directly request their favicon. They are not used when we evaluate the performance of our clustering approach.

In figure 6, we show the agreement between this validation process and the confidence scores for IPs used in our clustering algorithm. 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 would reject IPs the favicon process showed to be correct. The vast majority of these false-positives occur due to situations like CloudFlare, where empirically only a single partition of IPs is resolved for domains, but in practice all of the IPs are able to answer requests for any of the hosted domains.

In principle, validations like the use of favicons we've conducted, 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.

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 dominate

CDN	IP Space	Distinct ASNs
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

Figure 7: The number of IPs we cluster into each of the ten largest shared infrastructure platforms. The magnitude variance in size between Dyn, Fastly, Automattic and the others is primarily an indication on their relative reliance on Anycast.

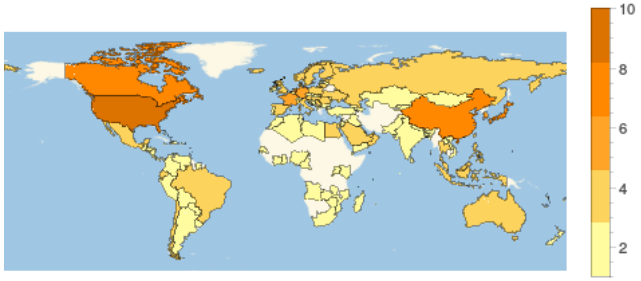

Figure 8: Points of presence of the top ten CDNs from Table 5. Note that Anycast is not accounted for here, so distribution is actually higher.

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 7, 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 ASN, while IPs from Akamai are largely resolved to IPs located in the ASNs of consumer ISPs.

In figure 8, we use the geolocation of ASNs 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 contain points of presence in at least 145 countries.

We can also see in this data the growing balkanization of the Internet [7]. In large countries which are pushing for more regulation of network sites, we see an increased amount of content resolved locally, rather than from external IP addresses. In 9, we plot how many of the domains are resolved to each country. We see at least 18% of all domains resolving to an in-country IP address for resolvers in China, while other

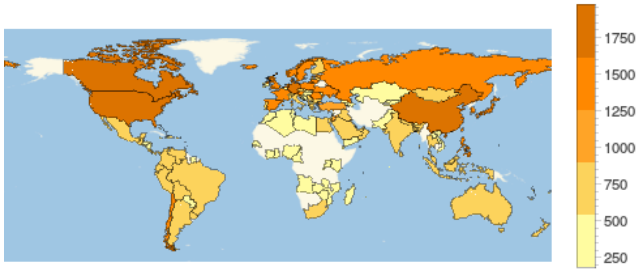

Figure 9: How many domains are resolved to servers in each country.

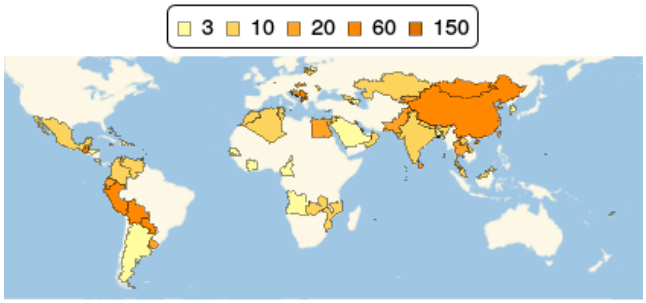

Figure 10: Number of domains inaccessible in each country.

countries like Mexico resolves only 6% of domains locally.

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 what the research resolves through 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.

Figure 10 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 ASN network in over 78 countries.

We then divide the instances of observed interference

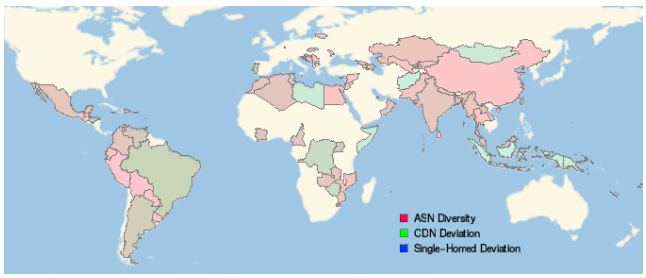

Figure 11: Types of interference by country. Domains in countries shaded red were inaccessible largely due to randomization of the IP space, those in green featured inaccessibility primarily in domains hosted on CDNs.

across other factors. Figure 11 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 at distributed sites, perhaps due to their popularity. This indicates that naive approaches have been missing a significant fraction of total interference instances.

5 Conclusion

Satellite is already a valuable system for understanding CDNs and measuring the prevalence of interference is a variety of networks. There are several areas for continued development that will extend the value of the system. In particular, our development efforts are focused on: (1) Integrating anycast IP address geographic resolution to get a better sense of where sites are located. (2) Developing a web site to interactively explore collected data. (3) Integration of additional probing mechanisms including reverse DNS, service detection, 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. We hope that by lowering the bar for collecting, aggregating, and understanding these measurements we can make the data much more available. We see Satellite as enabling consistent, long-term measurements of network conditions, providing increased transparency into understanding cloud connectivity, and monitoring interference around the world. We've shown the magnitude and growing predominance of the CDN industry amongst the top Alexa domains. Using the same data, we've also shown evidence of changing interference conditions around the world over the last year. Satellite is a fully open platform, and both the recorded data, and code allowing others to run their own measurements are available online at `http://satellite.cs.washington.edu`.

6 References

[1] Akamai. Alta: Product brief. `http://www.akamai.com/dl/product_briefs/product-brief-alta.pdf`.

[2] Alexa, 1996. `http://www.alexa.com`.

[3] Anonymous. Internet census 2012, 2012. `http://internetcensus2012.bitbucket.org/paper.html`.

[4] Anonymous. DNS census 2013, 2013. `https://dnscensus2013.neocities.org/`.

[5] M. Calder, X. Fan, Z. Hu, E. Katz-Bassett, J. Heidemann, and R. Govindan. Mapping the Expansion of Google's Serving Infrastructure. In *Proceedings of the ACM Internet Measurement Conference (IMC '13)*, 2013.

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

[7] R. Deibert and R. Rohozinski. Liberation vs. control:

The future of cyberspace. *Journal of Democracy*, 2010.

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

[9] M. Elsner and W. Schudy. Bounding and comparing methods for correlation clustering beyond ilp. In *Proceedings of the Workshop on Integer Linear Programming for Natural Langauge Processing*, pages 19–27. Association for Computational Linguistics, 2009.

[10] Farsight Security, Inc. Farsight DNSDB, 2010. `https://www.dnsdb.info`.

[11] A. Filast and J. Appelbaum. Ooni: Open observatory of network interference. In *USENIX Workshop on Free and Open Communications on the Internet*. USENIX, 2012.

[12] 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 (TWEB)*, 9(1):4, 2015.

[13] C. Huang, A. Wang, J. Li, and K. W. Ross. Measuring and evaluating large-scale cdns. In *ACM IMC*, volume 8, 2008.

[14] Hypestat, 2011. `http://www.hypestat.com`.

[15] Iclab, 2013. `https://iclab.org`.

[16] H. V. Madhyastha, T. Isdal, M. Piatek, C. Dixon, T. Anderson, A. Krishnamurthy, and A. Venkataramani. iplane: An information plane for distributed services. In *Proceedings of the 7th symposium on Operating systems design and implementation*. USENIX Association, 2006.

[17] 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. https://citizenlab.org.

[18] 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 *Networking*. IFIP, 2015.

[19] J. Muach. Open resolver project, 2013. `http://openresolverproject.org`.

[20] Opennet initiative, 2011. `https://opennet.net/`.

[21] Shodan. shodan, 2013. `https://www.shodan.io/`.

[22] J.-P. Verkamp and M. Gupta. Inferring mechanics of web censorship around the world. *Free and Open Communications on the Internet, Bellevue, WA, USA*, 2012.

[23] P. Vixie. Extension mechanisms for dns (edns0). RFC 2671, RFC Editor, August 1999. `http://www.rfc-editor.org/rfc/rfc2671.txt`.

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

[25] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldszmidt, and T. Wobber. How dynamic are ip addresses? In *SIGCOMM Computer Communication Review*, 2007.