Leveraging Internet Background Radiation for Opportunistic Network Analysis

Karyn Benson*†, Alberto Dainotti†, kc claffy†, Alex C. Snoeren*, Michael Kallitsis‡
*Computer Science and Engineering, UC San Diego †CAIDA, UC San Diego ‡Merit Network, Inc.
{karyn, alberto, kc}@caida.org, snoeren@cs.ucsd.edu, mgkallit@umich.edu

ABSTRACT

For more than a decade, unsolicited traffic sent to unused regions of the address space has provided valuable insight into malicious Internet activities. In this paper, we explore the utility of this traffic, known as Internet Background Radiation (IBR), for a different purpose: as a data source of Internet-wide measurements. We collect and analyze IBR from two large darknets, carefully deconstructing its various components and characterizing them along dimensions applicable to Internet-wide measurements. Intuitively, IBR can provide insight into network properties when traffic from that network contains relevant information and is of sufficient volume. We turn this intuition into a scientific investigation, examining which networks send IBR, identifying components of IBR that enable opportunistic network inferences, and characterizing the frequency and granularity of traffic sources. We also consider the influences of time of collection and position in the address space on our results. We leverage IBR properties in three case studies to show that IBR can supplement existing techniques by improving coverage and/or diversity of analyzable networks while reducing measurement overhead. Our main contribution is a new framework for understanding the circumstances and properties for which unsolicited traffic is an appropriate data source for inference of macroscopic Internet properties, which can help other researchers assess its utility for a given study.

Categories and Subject Descriptors

C.2.5 [**Local and Wide-Area Networks**]: Internet

Keywords

Internet background radiation; Network telescope; Opportunistic network analysis

1. INTRODUCTION

Obtaining data from a diverse set of hosts and networks is a major challenge in Internet measurement research. We explore the potential for an unconventional data source, unsolicited traffic sent to unused regions of the address space, known as Internet Background Radiation (IBR), to address this challenge.

Monitoring unused portions of the IPv4 address space reveals that IBR is of considerable volume, incessant, and originates from a variety of services [41, 48]. This unsolicited traffic is caused by scanning (e.g., searching for hosts running a vulnerable service), misconfigurations (e.g., a typo in the IP address for a mail server), backscatter (responses to packets with forged source IP addresses, including spoofed DoS attack), bugs, etc. Historically, researchers have used this traffic to study worms, DoS attacks, and scanning.

More recently, instead of studying malicious activities, researchers have leveraged IBR to learn about hosts and networks generating unsolicited traffic [12,15,20,22,24,32,44]. The pervasively sourced components of IBR make a darknet—a region of the address space exclusively dedicated to collecting IBR—the potential recipient of traffic from all networks connected to the global Internet: botnets employ machines worldwide to perform scans; misconfigurations can occur in any network; and many networks host services that are potential victims of DoS attacks (causing backscatter). Casado *et al.* [15] first proposed using IBR (and other types of "spurious" network traffic, such as SPAM emails) to illuminate regions of the address space where traditional techniques fail to provide visibility (e.g., in the presence of NAT). Recent studies of censorship events, IPv4 address space utilization, and filtering policies have verified this benefit [22, 24, 44].

However, these studies focused on isolated events or specific components of IBR. It is unclear if the same analysis techniques work on similar events or with different collections of IBR (e.g., using different times or IBR vantage points). More broadly, these studies do not provide insight into which properties are amenable to analysis using IBR and whether the networks themselves must have certain characteristics to allow IBR-based inferences.

To evaluate IBR's utility as an Internet-wide data source, we begin by evaluating properties that support opportunistically measuring many networks. In Section 4, we quantify the large number and diversity of sources, which facilitate insight into many networks. In Section 5, we analyze the components of IBR, through which we can glean considerable information from packet-level data (e.g., the operating system from TCP options). In Section 6, we evaluate IBR's persistent nature, which permits repeated observations and often predictable temporal behavior.

In the context of inferring global properties of Internet networks, IBR also has fundamental limitations and challenges. Although IBR originates from many sources, we lack control over who sends it and when. In particular, the mix of popular applications changes regularly [41], which reduces the predictability of IBR. Additionally, IBR is unidirectional; since a darknet does not respond to unsolicited traffic we cannot infer flow-level information. Moreover,

IMC'15, October 28–30, 2015, Tokyo, Japan.

ACM 978-1-4503-3848-6/15/10 ...$15.00.
DOI: http://dx.doi.org/10.1145/2815675.2815702 .

packets with spoofed source addresses will lead to inaccurate inferences; as a result, IBR needs to be sanitized. Section 3 provides a summary of our sanitization technique, which we previously verified [22].

The case studies we present in Section 8 highlight the strengths and weaknesses of using IBR as a data source for Internet-wide measurement. Our experience suggests that IBR is useful: (1) When the presence of a source in darknet traffic provides additional context. For example, although we find fewer open resolvers than active probing, we know that attackers are actively using the ones found in IBR. (2) To obtain a large sample. We could easily calculate uptime for over half-a-million sources, to determine that a common technique for inferring uptime is invalid for certain operating systems. (3) For hosts unreachable through active probing. We can determine the uptime for NATed clients, which are unlikely to respond to external probes. (4) To reduce measurement overhead. We can identify flapping and non-flapping routes without sending packets. Such an analysis could focus active probing on routes that have recently changed.

2. RELATED WORK

We are far from the first to analyize Internet Background Radiation. Pang *et al.* performed the first major characterization of IBR [41], with the aim of identifying and filtering out malicious traffic. Brownlee detected new activities in IBR based on inter-arrival time [14], while Wustrow *et al.* examined IBR from multiple darknets over several years to discover which destination addresses received a disproportionate amount of IBR, which is useful before assigning IP addresses [48]. We have different goals than these prior characterizations of IBR, however, leading us to consider different aspects: we ask "who sends IBR?" and "how often do we receive IBR?" instead of "why do we receive IBR?" or "where is IBR destined?" In particular, we focus on the *quantity and frequency of sources* sending IBR instead of the *volume of packets or bytes*. For this reason, another difference from previous IBR characterization studies is that we remove spoofed packets from IBR before performing our analyses (spoofed source IP addresses represent fake sources).

There have been a number papers that use IBR to provide proof-of-concept of a measurement technique [12, 20–22, 24], or examine the applicability of a certain type of traffic to Internet-wide measurement [32, 44]. The primary goal of our three case studies is not to lengthen this list, but to demonstrate the effects of IBR's nature on our ability to extract Internet-wide properties. Casado *et al.* considered (but did not quantify) the nature of unsolicited traffic in their proposal for its utilization in opportunistic measurement [15]. However, this paper provides a more comprehensive view of IBR as a measurement data source. We put other related work into this framework in Table 4.

3. DATASETS

Our primary datasets are collections of IBR. To assist in analyzing IBR, we also use a mapping of IP addresses to prefixes, ASes and geographic locations, and a classification of the types of Autonomous Systems (ASes).

IBR Traffic. A darknet or network telescope is a collection of routed but unused IP addresses, i.e., all traffic these addresses receive is unsolicited. Darknets capture—but do not respond to—IBR. Both UC San Diego and Merit Network operate large darknets, which we call UCSD-NT and MERIT-NT respectively. UCSD-NT observes traffic destined to more than 99% of IP addresses in a contiguous /8 block. MERIT-NT covers about 67% of a different /8 block.

We study packet traces captured from July 31 to September 2, 2012 and July 23, 2013 to August 25, 2013. We choose these time periods because they align with the ICMP-ping based census conduced by ISI [29]. We refer to these 34-day periods as the *2012 census* and *2013 census*, respectively. We label our datasets based on the collection site and the year: `UCSD-12`, `UCSD-13`, and `MERIT-13` (which are 5.1, 4.0 and 1.5 TB of compressed data, respectively). To perform a longitudinal analysis spanning several years, in Section 7.1 we also use flow-level datasets—a summary of pcap data (i.e., protocol, source IP, destination IP, source port, destination port, flags, TTL, and number of packets)—collected by UCSD-NT from April 2008 to January 2015.

We use two darknets to study how position within the address space influences our results. Comparing UCSD-NT and MERIT-NT is not straightforward since the darknets are different sizes. For a fair comparison, we construct the dataset `partial-UCSD-13`, which is the traffic to a subset of IP addresses in UCSD's darknet. Specifically, we include traffic to an IP address UCSD.B.C.D if the IP address MERIT.B.C.D is part of MERIT-NT. As a result, approximately the same number[1] of destination IP addresses contributes to `partial-UCSD-13` and `MERIT-13`. We explore differences between `UCSD-12` and `UCSD-13` to study how the time of collection influences our results.

Including spoofed IBR traffic, i.e., traffic with a forged source IP, in our analyses would likely lead to incorrect inferences. We apply previously published techniques [22] to obtain a list of unrouted networks and remove spoofed traffic from the pcap datasets. The primary technique identifies spikes in unrouted addresses observed per hour. From these spikes we develop heuristics to exclude spoofed traffic. The heuristics also remove almost all traffic with source IP addresses in known dark blocks, providing validation (the validation results of "de-spoofing" in `UCSD-12` and `UCSD-13` are available elsewhere [21, 22]). As a result, we reduce the number of /24 blocks that appear to send us traffic from ≈10M to ≈3M. As a final step, we exclude all traffic from unrouted IP addresses, which may exist due to failed egress filtering by remote networks or spoofed sources missed by our heuristics. We apply similar techniques to the flow-level data spanning several years.

Prefixes. To analyze networks at the prefix level, we map source IP addresses to BGP announced prefixes. We consider a prefix announced if, on the first day of the dataset, it is visible by 95% of the ASes peering—and providing a full routing table—with Routeviews and RIPE RIS collectors, based on RIB data.[2] For each IP address we use the most-specific prefix.

Autonomous Systems. We use CAIDA's Prefix-to-AS mapping dataset (pfx2as) to map IPv4 addresses to AS numbers [10]. CAIDA extracts this dataset from BGP announcements captured by Routeviews. Specifically, we use the mapping produced on the first day of the IBR datasets. To label ASes as transit/access providers, content providers, or enterprise networks, we use a dataset provided by CAIDA developed using a scheme similar to that proposed by Dhamdhere and Dovrolis [25].

Geolocation. We use historical MaxMind country-level databases to geolocate the .0 address of each /24 block in our IBR datasets. Since MaxMind updates the database regularly (to reflect changes in the address space), we use the databases produced on August 1, 2012 and August 16, 2013 for the *2012 census* and *2013 census* periods, respectively.

[1] A handful of IP addresses in UCSD-NT's /8 matching the MERIT.B.C.D criterion are not darknet addresses.

[2] We choose the same threshold as previous work [36].

	Announced 2012	Announced 2013	UCSD-12	UCSD-13	UCSD-13 Partial	MERIT-13
IP addresses	2.61B	2.66B	148M (5.7%)	133M (5.0%)	109M (4.1%)	111M (4.2%)
/24 blocks	10.2M	10.4M	3.13M (31%)	3.15M (30%)	2.65M (26%)	2.76M (27%)
Prefixes	410k	452k	198k (48%)	205k (45%)	170k (38%)	175k (39%)
ASes	44k	46k	24.3k (55%)	24.2k (54%)	19.3k (44%)	19.8k (45%)
Countries	245	236	234 (96%)	233 (99%)	231 (98%)	232 (98%)

Table 1: The number (and percentage of announced resources) of IP addresses, /24 blocks, prefixes, ASes, and countries observed in each dataset is consistent across sites (UCSD-NT vs. MERIT-NT) and years (2012 vs. 2013).

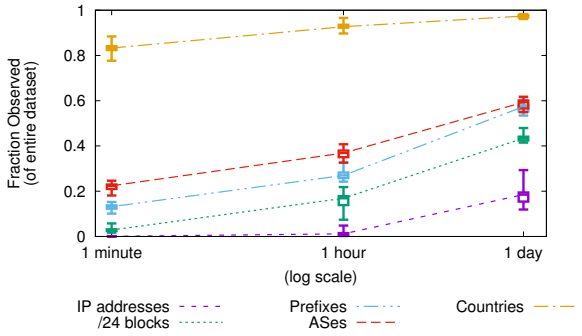


Figure 1: **Fraction of Sources Observed Per Minute, Hour and Day** (UCSD-13). The longer one observes, the more sources one can observe, especially at the IP address granularity.

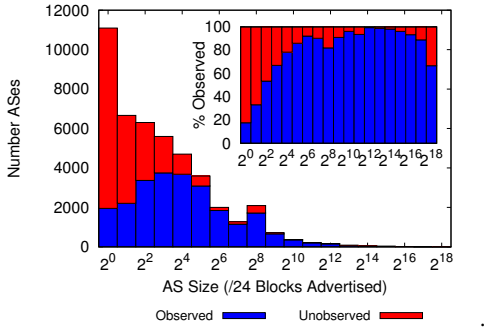


Figure 2: **Number and percentage of observed ASes by the number of /24 blocks announced.** Although we observe only half of announced ASes in UCSD-13, most missed ASes announce few /24 blocks.

4. WHO SENDS IBR?

We investigate how many and what type of networks send IBR. In all our datasets we observe traffic from a non-trivial number of IP addresses (> 100M), /24 blocks (> 2.6M) and prefixes (> 170k), and traffic from almost all countries and most large networks (including non-enterprise ASes). As a result, we can potentially use IBR to characterize many hosts and /24 blocks, and provide Internet-wide analysis at the AS or country-code level.

4.1 How many sources are observed?

Table 1 reports the absolute number of sources (IP addresses, /24 blocks, prefixes, ASes and countries) observed through our datasets. Compared to the total address space announced in BGP, we observe a few IP addresses, more than a quarter of /24 blocks, close to half of all prefixes and ASes, and almost all country codes. However, a large fraction of address space announced in BGP may not actually be "used", which we define as generating traffic on the global Internet [21, 50]. Based on previous literature, we observe about half of the inferred used /24 blocks: using seven different data sources, Dainotti *et al.* found 5.3M actually used /24 blocks in 2013 [21], while Zander *et al.* estimated that a total of 6.2M to 6.3M /24 blocks were used in June 2014 [50].

While the numbers in Table 1 are consistent across all four datasets, we find considerably fewer sources (except at country-level granularity) with shorter measurement intervals. Figure 1 shows statistics on the fraction of sources observed in a minute, hour, or day for UCSD-13 (the other datasets show similar values for all source and time granularities). As expected, by lengthening the observation period, we capture additional sources. However, due to repeated contact, the growth in number of sources observed is less than linear.

At the time granularities depicted in Figure 1, the number of observed sources is highly variable. Diurnal patterns in IBR [48] are one cause of variability, especially for small source granularity (i.e., IP addresses and /24 blocks). The changing composition of IBR (Section 5) contributes to the variance on longer time scales. Section 7.1 describes this aspect, based on variations observed over years.

4.2 What types of networks are observed?

We observe traffic from diverse locations. In the UCSD-13 dataset, we miss only three countries that announce /24 blocks. All three countries are small islands or collections of islands, each with a population of under 4,000 people [18].

Many ASes do not send IBR to our darknets: we observe about half of ASes announced in BGP. However, most missed ASes are small. Figure 2 shows, for UCSD-13, the distribution of observed ASes in terms of /24 blocks announced. Of the 20.6k unobserved ASes in UCSD-13, almost half announce a single /24 block, and 90% announce the equivalent of 8 or fewer /24 blocks. Conversely, we observe 86% of ASes that advertise the equivalent of at least a /16 block – we call these ASes large. ASes belonging to the US Department of Defense account for a fifth of unobserved large ASes, which appears to have many routed but "unused" /24 blocks [21]. In terms of AS type, we miss 26% of large ASes classified as enterprise, and about 4% of the large ASes classified as transit/access or content. The comprehensiveness of IBR's coverage of large ASes implies that it originates from diverse set of networks. In Section 6, we analyze how often the same ASes are observed over time.

4.3 Lessons learned

The number of sources captured by a network telescope is dependent on the duration of observation, the time of day, and the size of the network. Across our datasets, we consistently observe a significant fraction of the observably "used" IPv4 address space, and in particular nearly all large transit/access and content ASes. As a result, IBR has the potential to provide an Internet-wide view.

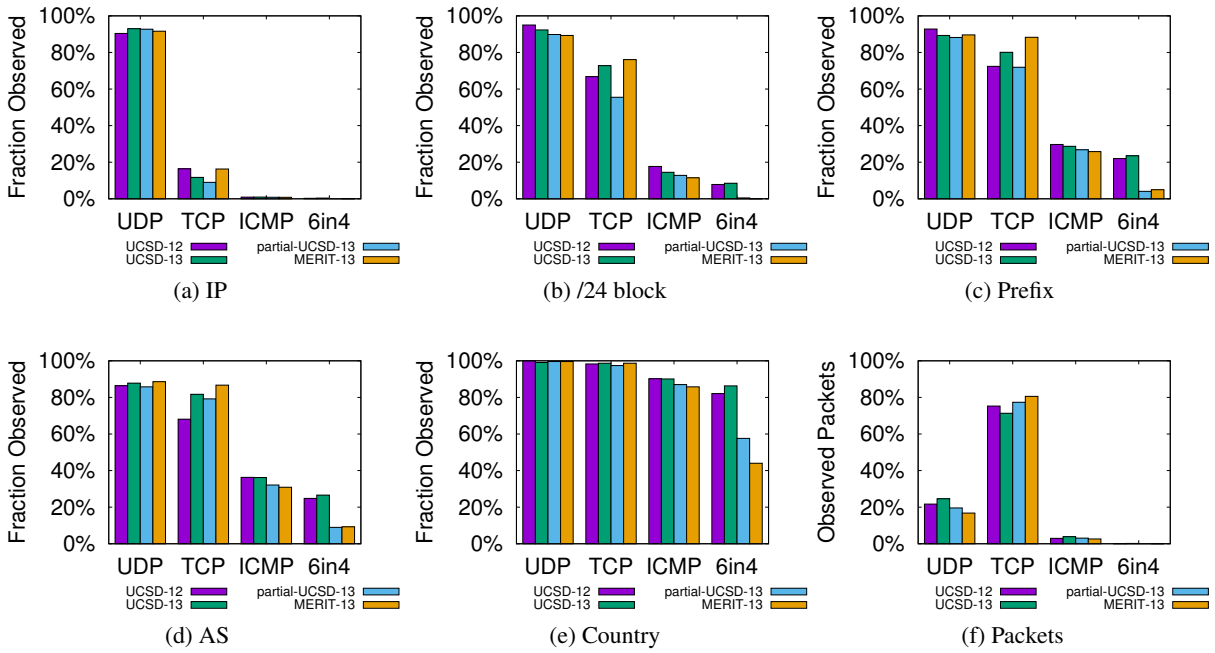


Figure 3: **Top protocols.** Most IP addresses send UDP traffic. At the /24 block, prefix, AS and country levels we observe a similar percentage of sources sending TCP and UDP. TCP accounts for most packets.

5. WHAT IS IBR MADE OF?

Most often, only a certain type of traffic is helpful in inferring a property of a network. For example, the authors of a previous study [12] use the retransmission behavior of TCP to infer packet-loss. It is thus important to understand the composition of IBR with respect to its potential information content. Enumerating all types of IBR-derivable information is a daunting, and probably impossible task. Instead, we characterize IBR along two basic dimensions: transport layer protocol and application, since the information encoded in IBR is a function of them.

5.1 How many sources use TCP vs. UDP?

Figure 3 reports the fraction (out of the total observed in the respective dataset) of IP addresses, /24 blocks, prefixes, ASes, and country codes observed through the most popular transport layer protocols. We observe most IP addresses via UDP traffic. Both TCP and UDP packets provide high visibility into /24 blocks and ASes, although neither provides complete coverage. All transport layer protocols provide excellent coverage of countries.

Wustrow *et al.* [48] characterize IBR based on the volume of *packets*, and not the number of *sources*. They find that from 2006–2010 TCP was the dominant protocol (above 75% of packets) for all years except 2008. Although our datasets are not directly comparable (they do not remove spoofed packets), we also find that TCP is the dominant protocol by number of packets (Figure 3f). Since UDP is the dominant protocol in terms of source IP addresses and TCP is the dominant protocol in terms of packets, the protocols may have different strengths when inferring network properties: UDP is more likely to provide wide coverage, while TCP is more likely to support analyses requiring repeated contact (Section 6).

5.2 Which applications contribute the most?

Here we classify IBR in terms of the process that generates the traffic. Port-based analysis, used in previous characterizations of IBR [48], is insufficient to analyze application-layer data [23]. For example, Qihoo 360 Safe traffic is the dominant application for all top-10 UDP destination ports.

Unlike Pang *et al.* who responded to unsolicited traffic [41], we passively collect IBR. With limited information, we perform a best-effort classification of IBR into components (that is, classes of phenomena responsible for different traffic) based on observations of initial communication attempts. For well-studied phenomena, we leverage known properties (e.g., the ranges of addresses of Conficker targets and the decryption algorithm for Sality's command-and-control packets [17,27]). The "Bro Scanner" category is based on Bro's definition of a scanner: contacting at least 25 unique destinations on the same port within 5 minutes [46]. We assign to the "Encrypted" category traffic with packets where $\text{entropy}(\text{payload}) \approx \log_2(\text{len}(\text{payload}))$. We then manually look for abnormalities in the number of observed source /24 blocks and derive a packet or flow-level filter matching the responsible traffic. We investigate and identify new phenomena based on TCP/UDP ports, UDP payloads, packet lengths, TCP flags, and number of packets. We complete the analysis in time (e.g., why a certain hour captures many /24 blocks) and space (e.g., why a darknet /16 block receives many /24 blocks). We perform this analysis iteratively: once we identify a component, we remove it from our data and find additional components causing abnormalities.

Table 2 reports the components that contribute a significant number of source /24 blocks. We aggregate some small components and all unclassified components into the "Other" category. We group the components based on the reason they appear in IBR: accidentally (i.e., due to bugs or misconfigurations), as part of a scan, as a by-product of spoofed traffic, such as DoS attacks, received by a network (which sends backscatter to the darknet), and for unknown reasons. Our classification process discovers some interesting large Internet phenomena. For example, most BitTorrent traffic appears to be the result of index-poisoning attacks that pollute the DHT

Component	UCSD-12		UCSD-13			MERIT-13		
	Total	Unique	Total	Unique	Partial Total	Total	Unique	∩ UCSD-13
Bugs & Misconfigurations								
File Sharing (BitTorrent, eMule, QQLive) [34, 35, 40]	2,640k	284k	2,490k	344k	1,910k	2,090k	377k	1,980k
Qihoo 360 Safe Bug [1]	1,450k	98.5k	1,340k	117k	1,110k	1,110k	138k	1,050k
Encapsulated IPv6 (6in4, Teredo) [5]	1,080k	9.48k	744k	11.5k	392k	368k	5.94k	312k
Gaming (Xbox, Steam) [3, 4]	503k	4.50k	490k	14.3k	258k	185k	11.9k	131k
Botnet C&C (ZeroAccess, Sality) [27, 38]	551k	17.3k	184k	4.97k	51.7k	51.6k	2.37k	25.7k
Scanning								
Conficker [17]	642k	24.4k	579k	58.1k	573k	568k	96.9k	563k
Bro Scanner [46]	597k	8.48k	197k	4.57k	104k	99.1k	4.06k	91.8k
Backscatter								
Backscatter [39]	394k	45.3k	392k	51.6k	247k	246k	21.3k	219k
Unclassified								
Encrypted [28]	1,450k	98.5k	1,340k	117k	819k	755k	29.8k	667k
Other	1,980k	73.8k	1,910k	127k	1,440k	1,70k	135k	1,410k
All Components	3,130k		3,150k		2,650k	2,760k		2,670k

Table 2: **/24 blocks observed by IBR component.** IBR is composed of many different types of traffic. File-sharing traffic contributes the highest number of /24 blocks in all datasets, but there are variations based on time (`UCSD-13` vs `UCSD-12`) and position (`partial-UCSD-13` vs `MERIT-13`). We observe most /24 blocks through multiple IBR components, implying that insight into a network is not dependent on a single type of traffic.

with bogus IP addresses. We determine a UDP payload was sent from Qihoo 360 Safe by investigating some live hosts at UC San Diego responsible for it. A byte-order bug, triggered when a host receives updates via a P2P network, causes this traffic. In Section 7, we link trends of the individual components to changes in IBR properties over time.

When studying 2010-era IBR reaching four /8 networks, Wustrow *et al.* find that scanning accounts for the majority of packets in all but 1.0.0.0/8 [48]. In our datasets, many well-studied, malicious IBR phenomena—scanning (including Conficker), backscatter—also account for most of the packets (collectively contributing about 83% of all packets in `UCSD-13`). But, surprisingly, malicious traffic is not the largest component of IBR in terms of sources. Packets with a P2P file-sharing payload contribute over 1.9M /24 blocks in all datasets, accounting for over two-thirds of all /24 blocks observed; Qihoo 360 Safe traffic alone contributes about 100M IP addresses.

We observe most /24 blocks through multiple IBR components, implying that many types of IBR can provide insight into the same networks. In particular, even without the top IBR components, the "Other" component alone, contributes with 1.4M /24 blocks. The "Unique" column of Table 2 reports the number of /24 blocks observed through a single IBR component. For each component, the number of unique /24 blocks is at least an order of magnitude smaller than the total number of /24 blocks observed through that component. As a result, if the composition of IBR changes slightly we would still observe many of the same networks.

5.3 Lessons learned

Some IBR-based inferences require a certain type of traffic; other network properties can be inferred regardless of the underlying application, but their success is dependent on the composition of IBR. Fortunately, IBR is made up of many components, each of which contributes relatively few unique /24 blocks (implying some analyses may be robust to fluctuations in IBR composition). While most packets are TCP (due to scanning and backscatter), we observe more IP addresses from UDP traffic (due to P2P and bugs). IBR is commonly known as malicious traffic. However, we find that the phenomena that contribute the highest number of sources (over 1M /24 blocks) appear to be of benign nature.

6. HOW OFTEN DO WE RECEIVE IBR?

In this section, we consider inferences that require multiple observations of a given host/network. For example, Benson *et al.* [12] determine that the path from hosts in an AS to a darknet changed by observing the behavior of the TTL field. In addition to looking for changes in given fields, we can leverage the timing between packets (e.g., to infer uptime [32]) and the predictability of repeated contacts (e.g., to infer outages [24]).

To study repeated contact from IBR sources, we report (1) how often a host/network is observed, (2) the length of time between the first and last observation of a source, and (3) the timing between contacts. Our approach is to partition our dataset into 1-minute, 1-hour, and 1-day time bins and record the sources sending IBR in each bin. In mathematical notation, let S and T be the set of all sources and time bins at given granularities, and $I_s(t)$ be an indicator function for a source s for a time bin t that is 1 if the source is observed, and 0 otherwise. For property (1) we compute, for each $s \in S$:

$$\sum_{\{t \in T\}} I_s(t);$$

for property (2) we determine, for each $s \in S$:

$$\max_t \{t \in T | I_s(t) = 1\} - \min_{t'} \{t' \in T | I_s(t') = 1\};$$

and property (3) can be expressed as a multiset, where we include for each $s \in S$ and $\{t \in T | I_s(t) = 1\}$ the value (if it exists)

$$t - \max_{t'} \{t' \in T | I_s(t') = 1 \wedge t' < t\}.$$

Communication attempts may span multiple time bins, which could lead to inadvertently skewing properties (1), (2) and (3). In Table 3, we report statistics on communication attempts (packets with the same {source ip, destination ip, protocol, source port, destination port} observed in one hour of data) by IBR component from `UCSD-13`. The number of communication attempts varies depending on the IBR component, as does the behavior of the hosts sending each type of traffic (as evidenced by the median number of attempts per source IP address). However, for all components, the average number of packets per communication attempt is small. Manual investigation reveals that the timing between packets is also

Component	Communication Attempts	Avg. Pkts per Attempt	Median Attempts per Source IP
File Sharing	1,120M	6.13	2
360 Safe	1,520M	1.62	11
Encap. IPv6	108M	4.49	2
Gaming	95.4M	1.04	1
Botnet C&C	13.3M	2.95	3
Conficker	13,800M	1.98	109
Bro Scanner	27,400M	1.10	684
Backscatter	20,700M	1.23	6
Encrypted	137M	2.33	1
Other	1,740M	3.33	3
Total	66,700M	1.50	11

Table 3: **Communication attempts by IBR component for** `UCSD-13`. IBR components vary in the number of communication attempts made, and the median attempts made per source IP addresses. But, all components have a low number of packets per attempt, which suggests binning the data will not result in significant double counting.

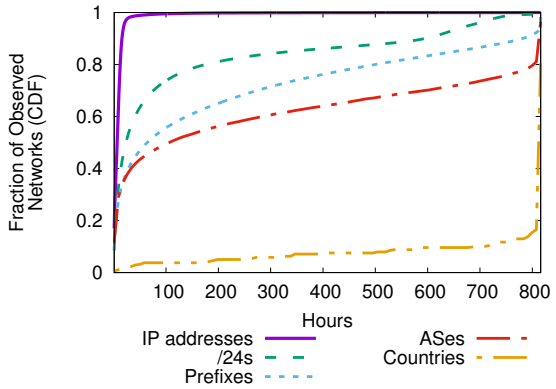


Figure 4: **CDF of fraction of sources observed using 1-hour time bins(`UCSD-13`).** We observe most countries and some ASes in nearly every time bin, which means we should be able to make repeated inferences at these source granularities.

small (e.g., 3 seconds between retransmission of Conficker packets). As a result, our partitioning approach confines most attempts to a single time bin, implying that binning does not significantly skew our calculations in the following sections.

6.1 How often do sources send IBR?

The frequency with which we can infer properties of a remote network depends on how often we receive traffic from that network. Figure 4 shows the cumulative distribution function of sources observed using 1-hour time bins in `UCSD-13`. The other datasets exhibit similar distributions. We observe frequent contact at coarse source granularities, i.e., countries and some ASes. The values on the far right of Figure 4 indicate the number of networks that we observed in every hour `UCSD-13`, which suggest that inferences requiring near-constant traffic samples are only possible for ≈80% of countries and ≈20% of ASes. We also explore (not shown here) the distribution of number of contacts with time bins of 1-minute and 1-day. As expected, the CDF curves shift towards more frequent contact as we move to larger time bins.

Approximately 12% of IP addresses are unsuited for repeated measurements because we observe them in only one 1-minute time bin, and we observe most IP addresses in less than 11 1-minute time bins. But as the size of the time bin increases to hours or days, the

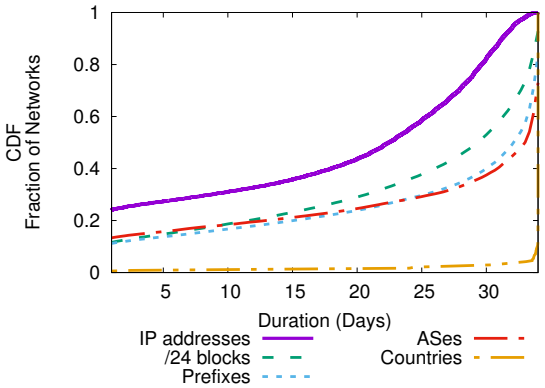


Figure 5: **CDF of contact duration (`UCSD-13`).** At all source granularities the contact duration is long, which is desirable for analysis throughout the datasets.

number of contacts per source increases. For example, we observe traffic from over 75% of IP addresses, /24 blocks, prefixes ASes and countries in multiple days.

6.2 What is the total duration of contact?

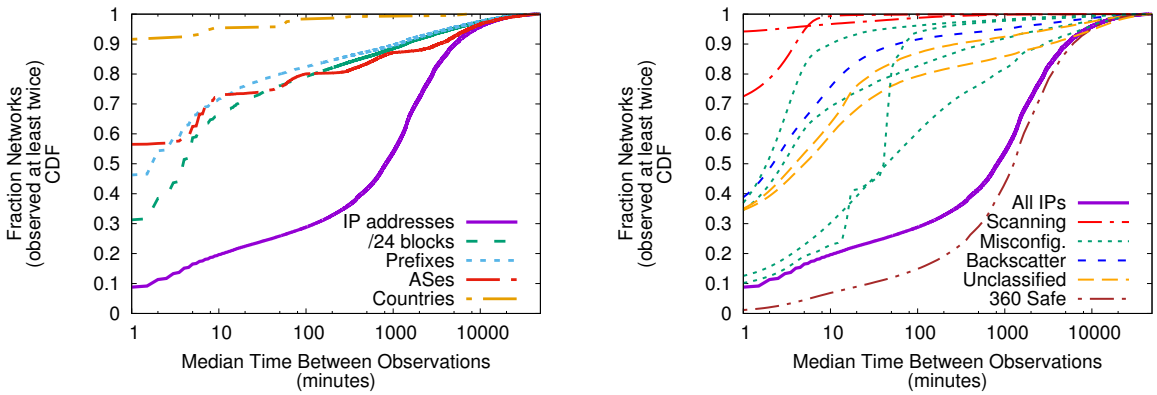
To conclude if our observations are the result of a single bursty event, or if sources are visible throughout the *2012* or *2013 census* periods, we investigate the range of times that we observe a source. We calculate each source's duration of contact (time of last contact minus time of first contact). Figure 5 shows the CDF of this distribution. The total duration of contact is long (over 29 days out of 34) for most /24 blocks, prefixes, ASes, and countries. Despite observing most IP addresses in only a few 1-minute or 1-hour time bins, the duration of contact is also long for IP addresses (50% IP addresses had a duration of contact longer than 22.5 days), implying that there is a long time between consecutive observations of a source (Section 6.3).

We attribute the long duration of contact at the IP level to Qihoo 360 Safe traffic, which has a diurnal cycle. Since about 70% of IP addresses send Qihoo 360 Safe traffic in `UCSD-13`, it strongly influences the overall duration at the IP address granularity. Without Qihoo 360 Safe traffic, 80% of IP addresses have a contact duration of less than one day. However, there is only a small influence on the duration of contact at the /24 block, prefix, AS, and country granularities. The signal for these aggregated granularities is comprised of a mix of traffic components and is not dependent on Qihoo 360 Safe.

This analysis shows the potential to make IBR-based inferences at the /24 blocks, prefixes, ASes and countries granularities for the duration of the datasets. At the IP-address granularity, we observe the sources throughout the datasets, but this is mostly due to Qihoo 360 Safe traffic.

6.3 Frequency of communication attempts?

To evaluate our ability to perform fine-grained analysis with IBR, we study the time between observations of traffic from a source. Figure 6a shows the median time between all sources that we observe in at least two 1-minute time bins. We observe most countries all the time: the median time between observations is 1 minute for 92% of countries. At the /24 block and AS levels, the time between observations is often longer, although the time between contacts at these granularities is often within 10 minutes. There is a longer period of time between observations of an IP address: half of IP



(a) All traffic.

(b) By IBR component (IP addresses).

Figure 6: **Median time between observations (`UCSD-13`).** Most /24 blocks, prefixes, ASes, and countries observed multiple times have a short time between observations (less than 10 minutes), which is desirable for fine-grained analysis. By component, scanning traffic has the shortest median time between observations.

addresses have a median inter-observation time of more than 13.7 hours. However, for some IP addresses the inter-observation time is still short (27% of IP addresses have a median inter-observation time of less then 1 hour).

Figure 6b shows the breakdown of median time between observations for IP addresses by IBR component. 360 Safe traffic heavily influences the overall behavior of IP addresses: 50% of IP addresses associated with 360 Safe have a median time between observations of greater than 21.2 hours (presumably because they receive updates about once per day). The median time between observations is substantially shorter for the other IBR components. As a result, our ability to conduct fine-grained analysis comes from IBR components other than Qihoo 360 Safe. Scanning traffic has the shortest time between observations: for over 90% of IP addresses the median time between observations is less than 4 minutes. One type of misconfiguration causes hosts infected by a botnet to send C&C traffic to the UCSD darknet and wait either 15 minutes or 1 hour between communication attempts. 360 Safe traffic does not heavily influence the time between observations at the /24 block, AS or country levels.

6.4 Lessons learned

We find that many sources repeatedly contact our darknets. We almost always observe traffic from most countries and many ASes, e.g., we observe them in nearly all time bins, throughout the entire observation period, and with a short time between observations. We continually, but not constantly, observe most /24 blocks and prefixes, e.g., they have a long contact duration but the median time between observations is often over an hour. At the IP level, a diurnal bug in Qihoo 360 Safe generates traffic that heavily influences the contact duration and time between intervals. When we exclude the Qihoo 360 Safe traffic, three-quarters of IP addresses have a contact duration of less than one day (i.e., we observe the source in a single day of our 34-day observation period). As a result, IBR is not well suited for long-term inferences at the IP address granularity.

7. SENSITIVITY ANALYSIS

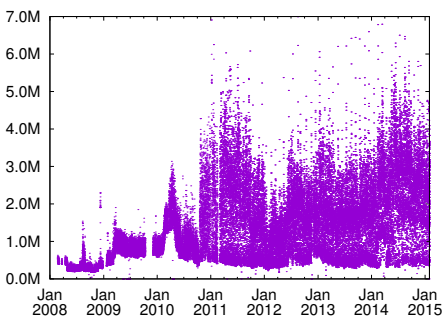
In this section, we examine the dependence of IBR on the time and site of data collection. We discover a number of differences, which can be attributed to the properties of influential IBR components. These results (1) confirm that the findings presented in the previous sections are representative in terms of number of IBR sources, the mix of components and visibility, (2) identify aspects of IBR that limit its ability to make inferences about remote networks, and (3) set expectations for the performance of other darknets.

7.1 Dependence on time of collection

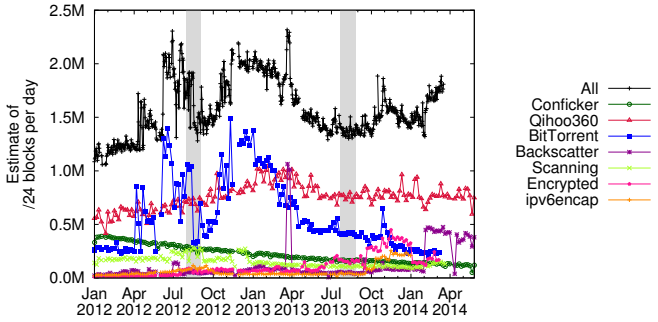
Over time, IBR evolves. Not just in terms of its constituent packets and bytes, as studied by Wustrow *et al.* [48], but also in terms of the number of sources sending IBR. To identify times when significant changes occurred, we consider: (1) the number of IP addresses observed per hour for most of 2008–2015 (Figure 7a); (2) the per-day contribution of the major components[3] over the 28-month period from January 2012 to April 2014 (Figure 7b); and (3) the total number of /24 blocks per component during the *2012* and *2013 census* (Table 2). Our ability to make network inferences is influenced by both the trends and erratic nature of IBR, including the following events:

- *November 2008*: Conficker worm outbreak
- *March 2010*: Significant BitTorrent traffic observed
- *October 2010*: Start of traffic from 360 Safe bug
- *March 2013*: A spike in Backscatter traffic as the result of a DoS on Spamhaus
- *February 2014*: Increase in backscatter containing responses to DNS queries
- *2012 census vs 2013 census*: Due to activity by the Carna Botnet in 2012, the number of /24 blocks labeled as Bro Scanners in `UCSD-12` is three times the amount in `UCSD-13`

[3] We extract some IBR components with a pcap signature. When operating on flow-level data, we use heuristics instead. E.g., for BitTorrent traffic we use popular message lengths (with low false positive rate) instead of examining the payload.



(a) **IP addresses observed per hour over a 7-year period (UCSD-NT).**



(b) **/24 blocks observed every 6th day at UCSD-NT by IBR component**. The shaded portions indicate the *2012* and *2013 census*.

Figure 7: Section 7 reviews interesting events represented in these plots.

In particular, we can attribute the increased number of sources in recent years to bugs and misconfigurations in P2P networks (Qihoo 360 Safe and BitTorrent). However, sources sending P2P traffic generally produce few connection attempts at irregular intervals. Combined with the decrease in Conficker traffic, this means that fine-grained analysis (repeated analysis on a short time scale, e.g., minutes) is becoming more difficult. To extract a predictable signal, we may need to analyze only specific components of IBR [12, 32]. On the other hand, erratic events may serve as an opportunity to increase visibility. For example, using IBR to assess IPv4 utilization during the Spamhaus event yielded more used /24 blocks [21]; locating open resolvers is possible with IBR as the result of an increase in DNS traffic. We group the events above into scanning activities (Conficker and Carna), P2P misconfigurations or bugs (BitTorrent and Qihoo 360 Safe), and backscatter (DNS responses and Spamhaus) and discuss each below.

Scanning activities. Both Conficker and Carna increased the number of sources scanning the darknet. Hosts participating in scans send many packets to the darknet—which we can leverage for repeated measurements. However, the Carna scans were a temporary phenomenon and Conficker is slowly declining. This decrease, combined with the emergence of BitTorrent and 360 Safe traffic (generated by sources which make relatively few connection attempts) results in fewer packets observed per IBR-visible host.

P2P misconfigurations/bugs. As a result of misconfigurations or bugs in P2P networks, we observe many sources, though these sources generally send few packets. We do not receive BitTorrent traffic consistently, and BitTorrent's erratic nature correlates with the total number of /24 blocks observed per day. In aggregate, 360 Safe traffic is diurnal: in `UCSD-13`, the average number of source IP addresses sending this traffic per hour varies between 165k at 20:00 UTC to 2.31M at 0:00 UTC. However, we do not observe clients using the software at predictable intervals. Thus P2P misconfigurations and bugs provide excellent coverage, but only when we do not need many packets per source or high predictability.

Backscatter. Normally, we think of backscatter from spoofed DoS attacks as coming from a small number of attacked machines or networks. Both the Spamhaus attack and the increase in DNS traffic show that the number of sources sending backscatter can actually be large. The Spamhaus attack [43] targeted Spamhaus' network, the networks carrying Spamhaus' traffic and strategically selected Internet exchange points. The increase in DNS traffic is caused by responses to spoofed queries — from many open resolvers simultaneously.

Backscatter events provide a period of increased visibility of remote networks, and it may be advantageous to infer network properties during these events. This window of opportunity may vary: the Spamhaus attack lasted a couple days, while DNS backscatter is an on-going phenomenon.

7.2 Dependence on position in IPv4 space

Wustrow *et al.* [48] find significant non-uniformity in the number of bytes and packets received by four /8 darknets in March 2010. However, we find more uniformity when considering the number of sources sending non-spoofed traffic to our /8 darknets. Intuitively, filtering out spoofed traffic removes some irregularities, and many IBR components target UCSD-NT and MERIT-NT with equal probability (e.g., scanning, backscatter, P2P misconfigurations).

In particular, we observe a similar number of /24 blocks through `partial-UCSD-13` and `MERIT-13` (2.65M and 2.76M respectively). Table 2 shows that `partial-UCSD-13` and `MERIT-13` also have a similar traffic composition. All components, except Gaming and Other, contribute approximately the same number of /24 blocks to each dataset. The Gaming difference can be explained by a misconfiguration: a single UCSD-NT IP observes 115k /24 blocks sending Steam traffic. In the Other category, 10 times as many /24 networks send TCP traffic destined to IP addresses matching {A.B.C.D | A=MERIT & C=13} than {A.B.C.D | A=UCSD & C=13}.

Additionally, many source /24 blocks send traffic to both UCSD-NT and MERIT-NT. The ∩UCSD-13 column of Table 2 shows the overlap—the number of /24 blocks observed in both `MERIT-13` and `UCSD-13` accounts for more than 84% of /24 blocks. We also observe an overlap of at least 49% in individual IBR components (Conficker produces the highest overlap, 99%) which implies that sources sending IBR likely target multiple /8 networks. Thus, it is likely that other portions of the address space receive packets from these sources.

However, we cannot examine all /8 darknets to understand the full effect of position. The non-uniform nature of IBR may cause variance when examining other darknets. Wustrow *et al.* find that many misconfigurations affect only the 1.0.0.0/8 block (e.g., traffic to 1.2.3.4) [48]; these misconfigurations may also influence the number of sources sending traffic to 1.0.0.0/8, in addition to bytes and packets. Additionally, we show in the next section that sources often do not target all subnets within a /8 darknet.

7.3 Dependence on darknet size

With smaller darknets, we expect to observe fewer sources and observe those sources less frequently. To study the effect of using a smaller darknet, we vary darknet size, from a /16 to a /8, by considering contiguous subnets of UCSD-NT as their own mini-darknet. Figure 8a reports for each darknet size, the range of source /24 blocks captured by these contiguous subnets in `UCSD-13`. We find, due to the non-uniform nature of IBR, significant differences in the number of sources captured by subnets of the same size.

Figure 8b shows for each /16 within UCSD-NT the number of /24 blocks captured during *2013 census*. In UCSD-NT, most variations can be attributed to: (1) the bug in Conficker's PRNG, (2) BitTorrent's RPC mechanism, KRPC, and (3) Encapsulated IPv6 traffic. Individual IP hotspots are observed as little spikes in Figure 8b, but create small discrepancies compared to the differences caused by the Conficker, BitTorrent and IPv6 components (for /16 or larger darknets).

Despite these discrepancies, based on median observations, the marginal utility of a single darknet IP address decreases as the size of the darknet increases (e.g., doubling the size of the darknet results in fewer than a 2x increase in the number of /24 blocks observed). In the /8 to /16 range, we observe a power-law relationship between the median number of /24 blocks observed and the number of darknet IP addresses monitored. Specifically, in the /8 to /16 range of UCSD-NT, reducing darknet size by a factor of two should yield about 89% of the original /24 blocks. As a result, we expect small darknets to also observe many /24 blocks. But this power-law relationship does not hold for all darknet sizes: the median number of /24 blocks observed by an IP in `UCSD-13` is an order of magnitude less than the number implied by the power law relationship.

8. CASE STUDIES

The previous sections identify and characterize aspects of IBR relevant to conducting opportunistic network analysis. In this section, we examine how these aspects influence network inferences with IBR. Table 4 shows 13 types of IBR-based inferences, which vary along the dimensions of packet-level information (Section 5) and number of required observations of the source (Section 6). Not all sources with the specified dimensions of (packet-level, number of observations) will be analyzable. For example, to calculate uptime with two TCP packets, the source needs to send packets with TCP timestamps from an operating system where the technique is valid.

The inferences in Table 4 include previous studies where the authors applied their method Internet-wide, previous studies where a technique used on a small scale may be usable for Internet-wide analysis, techniques used with measurement data other than IBR that may be applicable to IBR, and novel uses of IBR. While not exhaustive, Table 4 suggests that IBR is versatile in terms of the number and range of inferences it may be able to support, including existence (or active use) of a network resource, host attributes, and network behavior.

We consider three case studies in detail: locating open resolvers (Section 8.1), determining uptime (Section 8.2), and identifying path changes (Section 8.3). The goal of these case studies is to highlight some strengths and weaknesses of using IBR. The open resolver case study uses erratic but information-rich traffic; the uptime case study applies a common technique to a diverse set of hosts, but we need to take steps to ensure its accuracy; the path change case study takes advantage of repeated contact at the AS-level, but is not as accurate as the standard active technique (traceroute). Through these case studies we extend our knowledge of the state of the Internet and we identify some situations where IBR can assist in Internet-wide measurement: (1) when the presence of a source in IBR provides additional context; (2) to obtain a large sample; (3) for hosts unreachable through active probing; and (4) to reduce measurement overhead.

	`UCSD-12`	`UCSD-13`	`MERIT-13`	`UCSD-14-DNS`	Open Resolver Project [9]
Unique IPs	49,111	3,401	835	1,561,324	37,607,402
Recursion-Avail.	42,312	2,298	835	1,518,360	32,917,724
OK	48,746	2,991	329	1,437,310	32,595,867
FORMERR	43	7	7	1,422	841
SERVFAIL	317	148	43	1,445,276	919,899
NAMEFAIL	215	200	518	1,349,092	153,466
NOTIMP	7	8	7	64	166
REFUSED	173	241	35	136,328	4,433,126

Table 5: **Recursive DNS resolvers.** DNS responses reaching the darknet with the Recursion-Available bit set indicate an open resolver. The number of open resolvers sending IBR increased in 2014 (thirty-fold over `UCSD-12`), allowing us to infer their existence and provide insight into traffic reaching authoritative name servers.

8.1 Locating open DNS resolvers

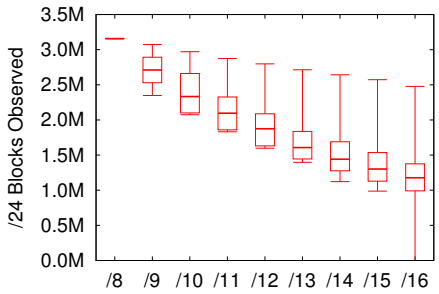
In a reflective amplification attack, the attacker sends a small, spoofed packet to a node that responds with a much larger packet to the spoofed source IP address. These attacks often use DNS recursive queries. As a security mechanism, many DNS servers only answer recursive queries within their administrative domain. DNS servers not implementing this security mechanism are known as "open resolvers." Locating open resolvers is a first step in improving DNS security.

Our objectives with this case study are: (1) to show that the changing composition of IBR can provide an opportunity to learn about the Internet; (2) to show that IBR can supplement active probing techniques by providing additional information; (3) to expose limitations in IBR's ability to determine the existence of network components.

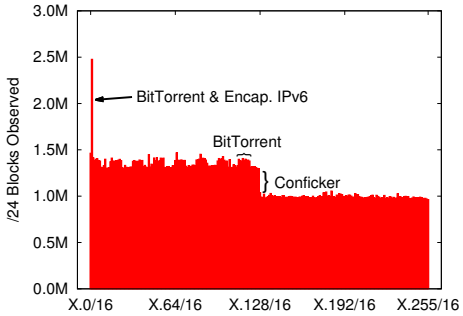
Method. If a darknet receives a DNS response, the most likely scenario is that a DNS server is responding to a spoofed query. In this section, we consider all UDP source port 53 traffic. We label an IP address as an open resolver if the Recursion-Available flag is set, as it indicates the willingness to resolve recursive queries. This way, we actually locate either a machine that accepts recursive queries from any IP address or that recursively resolves domains on behalf of a forwarding open DNS server [45]. We do not check the correctness or consistency of responses reaching the darknet, but we include response codes in our analysis.

Results. Table 5 shows that we observe few open resolvers in `UCSD-12,13` and `MERIT-13`. However, starting around February 2014, we observe a sustained increase in DNS responses (also visible as an increase in backscatter in Figure 7b). Van Nice reports that this type of attack is responsible for 3% of global ISP DNS traffic, which may result in DoS to (a) the resolvers, (b) authoritative name servers, and (c) web sites hosted by the authoritative servers [47]. To show the magnitude of open resolvers during this time period, we create a dataset, `UCSD-14-DNS`, between January 20, 2014 and March 1, 2014.

The Open Resolver Project (ORP) sends DNS queries to the entire IPv4 address space over a period of 6.5 hours, once per



(a) Contiguous blocks of the darknet and how many blocks are observed.



(b) How many /24 blocks are observed by each /16 in UCSD-NT.

Figure 8: **Effect of size and position on number of /24 blocks observed**. There is a power-law relationship between number of /24 blocks observed and size of the darknet (Figure 8a), but significant variance based on position in the darknet (Figure 8b).

Number of Observations (Section 6)	**Packet Layer** (Section 5) **Internet (IP)**	**Transport (TCP/UDP)**	**Application**	
One	Ascertaining IPv4 Utilization † (through source IP) see: [21, 22]	Discovering Services † (through TCP flags) similar to: [39]	Locating Open Resolvers † (through DNS responses)	Determining Filtering Policy † (through Conficker) see: [44]
Two	Identifying Path Changes ◊ (through TTL) extend: [12]	Determining Uptime ☐ (through TCP timestamp) apply: [37]; other: [32]	Evaluating Security Improvements ◊ (through Conficker traffic reduction)	
Many	Deducing Packet Sending Rate ☐ (through IPID) apply: [16, 33]; other: [26, 32]	Detecting NAT Usage ☐ (through TCP options and TTL) apply: [49]; other: [11, 13]	Assessing BitTorrent Client Popularity ☐ (through uTP handshake messages)	
Predictable	Detecting Outages ◊ (through number sources) extend: [20, 24]	Recognizing Packet-loss ◊ (through pkts/connection attempt) extend: [12]	Determining Number of Disks ☐ (through re-seeding of Witty's PRNG) see: [32]	

† = Existence of Resource ☐ = Attributes of End Hosts ◊ = Network Changes

Table 4: **Example inferences.** Inferences made through IBR require various numbers of packets and types of packet-level information.

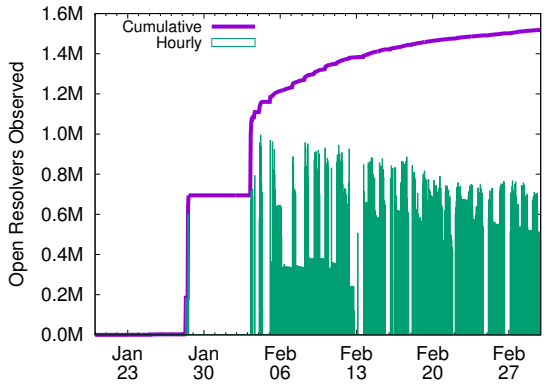


Figure 9: **Open resolvers in `UCSD-14-DNS`**

week [9]. For comparison, we consider all open resolvers[4] in ORP data in the same time period of `UCSD-14-DNS`. Table 5 shows that the 1.5M open resolvers found in `UCSD-14-DNS` are about 4% of the total found by ORP.

Figure 9 shows the number of open resolvers observed each hour, as well as the cumulative number, observed in `UCSD-14-DNS`. After an initial spike, the cumulative number of open resolvers grows slowly, despite observing over 200k open resolvers in many hour bins. Since we observe only a fraction of the known lower bound for open resolvers [9], this behavior indicates reuse, i.e., the phenomenon generating the DNS responses is repeatedly sending spoofed packets to the same set of open resolvers. Due to this repetition, smaller subnets of UCSD-NT capture a similar number of open resolvers (at least 89% of all open resolvers in `UCSD-14-DNS` with /16 subnets).

Validation. We verify that the open resolvers we include in `UCSD-14-DNS` are actually open resolvers by examining overlap with the ORP data in the same time period. Almost all (84%) of IP addresses observed in `UCSD-14-DNS` also appear as open resolvers in the ORP dataset. The remaining 16% are likely due to hosts intermittently online or were affected by Internet middleware (e.g., firewalls may drop packets from the ORP scan).

Comparison to other data sources. Although ORP has better coverage, active probing cannot reveal which open resolvers are actually used in attacks. The open resolvers ORP missed may include DNS servers that respond to packets from any source IP address, but only on certain interfaces (e.g., behind a firewall).

Through IBR, we can add to the knowledge of the phenomenon starting around February 2014. Specifically, we considered the "attack" traffic from the 462 second-level domains in `UCSD-14-DNS` that resulted in over 50k open resolvers sending traffic to UCSD-NT.

- baidu.com was the first second-level domain used in the attack – six days prior the second domain reaching our "attack" threshold. This was likely a testing phase.

[4] Specifically, the IP address of the responding DNS server, which is not necessarily the queried IP address.

- The attacks reused name servers (e.g., 36 domains had a nameserver matching *.dnspod.net), suggesting victims are repeatedly targeted.
- UCSD-14-DNS observes more sources with errors (e.g., SERVFAIL or NAMEFAIL) than ORP. Many of the open resolvers discovered in UCSD-14-DNS responded with non-errors and errors for queries for the same second-level domain, implying that the attack successfully inundated authoritative name servers with queries.

Discussion. IBR can supplement other measurement techniques, Dainotti *et al.* leveraged IBR to discover hosts that are intermittently used or behind firewalls in a study of IPv4 address space utilization [21, 22]. For similar reasons, we find additional open resolvers through IBR. Additionally, the observation in IBR or the differences between datasets may reveal additional information about our inferences, e.g., that an open resolver is being used maliciously, or that port filtering is used [44].

We observe more open resolvers due to a change in the composition of IBR. It is possible that this phenomenon could halt, in which case our coverage of open resolvers would decrease significantly. This variability is, in part, due to our dependence on a specific application. When many types of traffic contribute to a signal, we expect our ability to make inferences to improve. For example, in IPv4 address space utilization, any type of traffic can imply usage [21].

8.2 Determining uptime

We explore inferring end host uptime. Studying uptime can help understand human behavior, identify machines that have not applied security updates, and select resources with better availability.

Our objectives in this case study are: (1) to explore an inference requiring repeated contact; (2) to highlight the benefit of relying on information from the transport layer over upper-layer information from a specific application; (3) to show how IBR can provide unique insights, unavailable through other data sources.

Method. We use TCP timestamps to calculate uptime [37], a technique already implemented in Nmap [2] and p0f [49]. RFC 1323 specifies that TCP timestamps should be obtained from a clock that is approximately proportional to the real time [30]. Under the assumptions that (1) the OS zeros the counter at boot time, (2) the timestamp has not wrapped, and (3) network speeds are about constant, we can compute the frequency of the timestamp increments and total uptime. Specifically, for two packets j and k received at times r_j and r_k respectively with TCP timestamps t_j and t_k, the frequency of the timestamp increments is $f = \frac{t_k - t_j}{r_k - r_j}$, and the uptime (when packet k is sent) is $\frac{t_k}{f}$.

For each hour of data, we calculate frequency and uptime for each source IP sending TCP timestamps, and use p0f to determine the operating system that sent the packets. We then aggregate over all hours of data, excluding sources when either p0f reports conflicting OSes, or we determine that the OS violates assumption (1), or we receive packets that reveal conflicting uptimes (e.g., from two hosts behind a NAT). Additionally, we verify that the uptime is less than a year and that the frequency is close to a typically used value (e.g., one-third of IP addresses have a clock rate of 1000Hz) before including an IP address in our analysis.

Validation. To validate this technique, we analyze the accuracy of assumptions (1) and (2). Table 6 summarizes our findings in ensuring that the TCP timestamp is set to zero at boot time. First, we verify the accuracy of TCP timestamps on our own machines using p0f. We found inconsistencies for iOS and Mac OS, and exclude IP addresses with these OSes from analysis. Additionally,

OS (from p0f)	# Srcs	Verified	Distribution	Include?
Linux 2.4.x	217,989		Wraps @27 hours	no
Windows 7 or 8	102,097	✓	70% up for less than 1 day	yes
Linux 3.x	52,200	✓	Longer uptimes less likely	yes
iOS iPhone/iPad	48,360	×	Most uptimes 3 to 13 days	no
Mac OS X 10.x	32,721	×	Most uptimes 3 to 13 days	no
Linux 2.2.x-3.x	28,034		Wraps @27 hours	no
FreeBSD	21,717	✓	Reboots for patch [8]	yes
Linux 2.6.x	17,290		Longer uptimes less likely	yes
Linux 2.4.x-2.6.x	14,800		Longer uptimes less likely	yes

Table 6: We verify that the uptime inferred by TCP timestamp method matches the actual uptime of a machine, and by examining the distribution of suspected uptimes observed in UCSD-13.

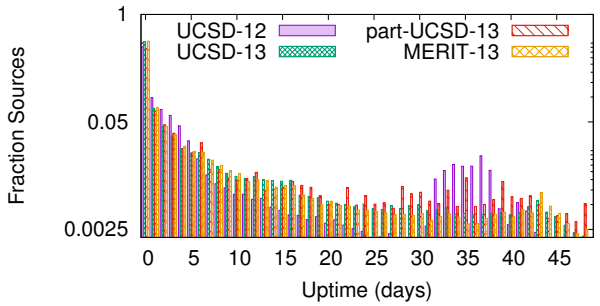


Figure 10: Distribution of uptime in days for IP addresses with TCP timestamps.

we examine the distribution of uptimes in UCSD-13 for each OS individually. We exclude two OSes, Linux 2.4.x and Linux 2.2.x-3.x, because the TCP timestamps appear to reset when the counter reaches 100M (at approximately 27 hours). We include Windows 7/8, which has a similar distribution from hour 0 to 24; but there is no evidence of a reset, implying that Windows 7/8 users generally turn off their machines every day.

Another concern is that the TCP timestamp will wrap once it meets its maximal value. The fastest timestamps we observe show clocks with frequencies on the order of 1000Hz, which will wrap about every 49 days. In Figure 10, about 0.1% percent of hosts have an uptime of 49 days, which suggest the impact of a wrapping timestamp is minimal.

Results. In UCSD-12, UCSD-13, partial-UCSD-13, and MERIT-13, we were able to infer uptimes associated with 290,697, 208,104, 57,990, and 47,122 IP addresses respectively. Both partial-UCSD-13 and MERIT-13 reveal significantly fewer uptimes than UCSD-12 and UCSD-13, showing the influence of darknet size and temporal fluctuations (Section 7). Despite the differences in coverage, the data sets provide a consistent picture of uptime. Figure 10 shows that most hosts have short uptimes in all datasets, and a significant fraction have an uptime of less than 1 day. For the next three weeks, the fraction of up hosts decays exponentially, consistent with a constant probability of being turned off/rebooted. In UCSD-12, we observe many hosts with an uptime of about 35 days, many of which run Linux. The boot times of these machines are consistent with applying a newly released kernel security fix [7]. Similarly, in 2013, FreeBSD required a reboot after an update to BIND [8], but the influence on our aggregated data is smaller.

Discussion. The main benefit of using IBR to infer uptime is the diversity in end hosts analyzed. To the best of our knowledge, this is the first study to provide an Internet-wide analysis of uptime. Nmap and p0f both use the TCP timestamp technique, but

measurements from a single vantage point (and not based on IBR) are limited in the sources they can evaluate. Active probing will not reach end hosts behind a firewall or NAT, whereas passive observation will be biased based on the population observed. Our study used over a half a million sources to validate the approach; but we could only determine uptime for 40k to 200k hosts (our analysis could improve, if instead of discarding all traffic from IP addresses used by multiple hosts, we isolated the timestamps for each host).

Kumar et al. examined IBR from the Witty worm to extract host uptimes. However, since Witty targeted a buffer overflow in network security products, the number of networks they could analyze was limited (inferring uptime for only about 800 machines) and not diverse (about a quarter of the machines were from only two institutions) [32]. Inferring properties from information extracted at the transport layer expands our coverage.

8.3 Identifying path changes

Detecting and analyzing path changes provides insight into Internet path stability [19, 42], and outages [12, 31, 51]. Our goals with this case study are to explore an inference that: (1) requires successive measurements; (2) has an element of predictability (although IBR composition is erratic, TTL is predictable); and (3) shows how to use IBR to reduce the active probing required to infer changes (similar to [31, 51]).

Method. We extend the technique of [12] to identify path changes from remote ASes to the darknet, which relies on the insight that the TTL of a received packet reflects the number of hops on the path to the darknet. If the path is unchanged, all packets from a host will have the same TTL. We calculate the number of hops by subtracting the TTL from the next highest power of 2 (a technique used in [13]), excluding any packets with a TTL less than 3 since they likely originate from traceroute and are not a predictable measure of hop count. When the number of hops from a source to the darknet increases or decreases, we infer a likely path change (similar to a previous technique for monitoring traffic at a CDN [51]). Note this method will not detect changes that result in the same length path (but through different routers).

For each IP address, we calculate for each 5-minute time bin, t, $\max_t$ and $\min_{t-1}$, the most and least number of hops taken at time t respectively. We consider a path to have changed if $\max_t > \max_{t-1}$ or if $\min_t < \min_{t-1}$. We expect most path changes to occur during a 5-minute bin, and not at time bin boundaries; our requirement for a change will identify changes that occur during a 5-minute bin (the time bin includes packets with the old TTL and the new TTL). This method should also account for a change in load balancing paths (the whole distribution shifts). The method will have some false positives due to NAT (when a new host, with a longer/short path starts transmitting) as well as false negatives.

To study changes affecting larger source granularities, e.g., a prefix or AS, for each time bin we also calculate the percentage of IP addresses that sent packets in that time bin as well as the previous one, and also indicated a path change. Using multiple sources from a prefix or AS increases our confidence that an event occurred. In particular, we can identify path changes affecting large portions of the address space (as opposed to the Internet edge).

Results. We are interested in paths that we can continually monitor, which we call *always-analyzable*. Section 6 showed that only countries and a few ASes send IBR to our darknets every minute. Table 7 confirms that few sources are always-analyzable; not shown is the significant overlap of such sources across datasets: 1300 ASes are in both UCSD-13 and MERIT-13, and 1000 ASes are in both UCSD-13 and UCSD-12. The UCSD-13 data yields the best insight into path changes for transit/access ASes. Although large ASes (announcing a /16 or more) are more likely always-analyzable, half of the always-analyzable sources are small (announce less than a /16 block).

	UCSD-12	UCSD-13		MERIT-13
			Partial	
IP addresses	2.5k	2.8k	2.4k	2.2k
/24 blocks	2.3k	2.6k	2.1k	2.0k
Prefixes	3.3k	3.6k	2.7k	2.9k
ASes	1.6k	1.7k	1.4k	1.4k
Countries	146	155	145	148

Table 7: Number of sources for which we can detect path changes throughout our measurement periods is consistent across datasets.

Validation. We validate our method using historical traceroutes from Ark nodes [6] located in always-detectable ASes in UCSD-13. The Ark infrastructure uses teams of about 20 nodes to send traceroutes to every routed /24 block over a span of 2-3 days [6]; thus, we can expect about one traceroute per minute from each Ark node to reach the darknet. Nine Ark nodes are in 8 always-detectable ASes, including five educational networks, two large transit providers, and a Regional Internet Registry.

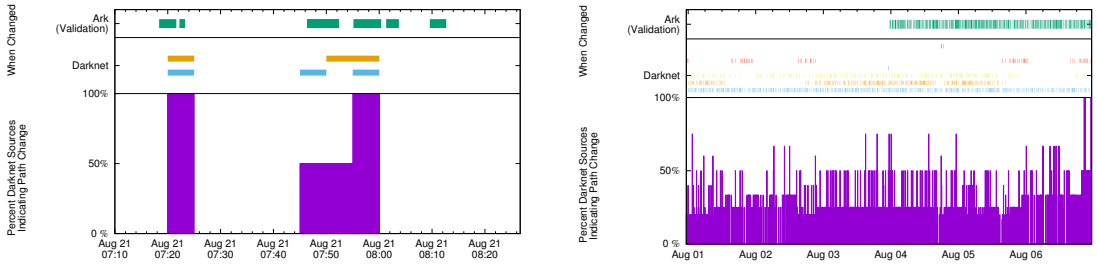
We cannot validate all path changes from the hosts sending IBR, as we do not know when these hosts start sharing links to the darknet. However, AS-level path changes should be observable in both Ark data and IBR. We find our analysis of other IBR-transmitting IP addresses frequently corroborates path changes in traceroute data. Figure 11 reports, for events in KIST (ASN1237) and Purdue (AS17), the percentage of hosts in darknet data signaling a path change, and the periods of time (the colored) periods that a path change was observed from IPs in both darknet and traceroute data.

KIST had very few path changes (in both types of data). Figure 11a includes all traceroute-inferred path changes for KIST, and all but one path change in UCSD-13. Most traceroute-inferred path changes occur around the same time as the darknet-inferred changes. Traceroute reveals that the path change occurred in the core of the network. Further investigation of the KIST sources suggests that traffic from the darknet sources used multiple paths in the 8:00 to 8:10 time bins (during these time bins the hop count was 16 or 17; outside of the time bins the hop count was 16). For one of the IP addresses, it is possible to look at a 1-minute time bins. With this granularity all darknet-inferred changes align with traceroute-inferred changes.

Figure 11b shows many path changes over a six-day period for Purdue in both Ark (8.9k changes), and darknet data (1.3k 5 minute bins with changes). Several IP addresses produce evidence of frequent path changes. Before August 4, 2013, traceroutes sent by the Ark monitor to UCSD-NT used the same route out of Purdue, but after this date, traffic from the Ark node traversed multiple routes out of Purdue's network. A likely explanation is that some Purdue sources used stable routes, while others used flapping routes; on August 4, 2013 the Ark node switched to using the flapping routes.

It is future work to fully validate our method; but from these examples, we suspect our technique can provide strong evidence of paths changes to the darknet. At a high level, our results for path-change detectable ASes are consistent with previous studies of route persistence [19, 42].

Discussion. Although IBR is an erratic data source, this example shows that it can provide insight into abnormal events and macroscopic dynamics. Our success with this case study is partially due to the aspect of IBR we are evaluating: the expectation that the initial TTL value remains the same is true regardless of the number of sources sending IBR or the volume of IBR, although in-



(a) KIST (b) Purdue

Figure 11: **Identifying path changes with IBR.** The top portion of each figure shows our validation data from Ark. The middle portion of each figure shades, for each source, the periods it inferred a path change. The bottom portion of each figure shows the percentage of darknet IP addresses signaling a path change. We identify the start of path change events at KIST, and route-flapping at Purdue.

creases in either would likely improve our coverage and accuracy. This path change detection method would work best in conjunction with other data sources. Like PlanetSeer and Hubble, passive traffic measurements such as IBR can help inform when and where active measurements would be most useful [31, 51]. IBR also provides features that traceroute and BGP data lack, e.g., no injected traffic required, and intra-AS visibility, respectively.

9. SUMMARY

We create a framework, and use case studies to apply the framework, to investigate the utility of Internet Background Radiation to support inference of a range of properties of networks across the global Internet. Using traffic from two large darknets, we carefully characterize it along dimensions applicable to macroscopic Internet measurements. We examine which networks send IBR, identify components that enable opportunistic network inferences, characterize the frequency and granularity of traffic sources, and analyze sensitivity to time of collection and position in the address space. Three case studies highlight the range of inferences possible with IBR, and show that IBR can supplement existing techniques by improving coverage and/or diversity of analyzable networks, and reducing measurement overhead. We also taxonomize 10 other potential inferences, and hope that our framework encourages additional consideration of the circumstances and properties for which unsolicited traffic is an appropriate data source for Internet research. This work demonstrates the applicability of IBR to many types of Internet measurement studies. More generally, this framework can serve as a template for evaluating the utility of other Internet measurement data sources.

10. ACKNOWLEDGEMENTS

We would like to thank Nevil Brownlee, Louis DeKoven, Kirill Levchenko, Brian Kantor and Cooper Nelson for their assistance in investigating IBR phenomena.

This research used resources of the National Energy Research Scientific Computing Center, a DOE Office of Science User Facility supported by the Office of Science of the U.S. Department of Energy under Contract No. DE-AC02-05CH11231. This work also used the Extreme Science and Engineering Discovery Environment (XSEDE), which is supported by National Science Foundation grant number ACI-1053575.

This work was supported by Department of Homeland Security Science and Technology Directorate grant FA8750-12-2-0314, cooperative agreement FA8750-12-2-0326, and NSF grant CNS-1228994.

11. REFERENCES

[1] 360 Total Security Software License and Service Agreement. `www.360safe.com/totalsecurity/en/licence.html`.

[2] Chapter 8. Remote OS Detection: Usage and Examples. `nmap.org/book/osdetect-methods.html#osdetect-ts`.

[3] Server queries. `developer.valvesoftware.com/wiki/Server_queries`.

[4] Xbox 360 network ports and router configurations for Xbox Live. `support.xbox.com/en-US/xbox-360/networking/network-ports-used-xbox-live`.

[5] Teredo Overview. `technet.microsoft.com/en-us/library/bb457011.aspx`, 2003.

[6] Archipelago Measurement Infrastructure. `www.caida.org/projects/ark`, 2006.

[7] Important: kernel security and bug fix update. `www.redhat.com/archives/rhsa-announce/2012-July/msg00014.html`, 2012.

[8] BIND remote denial of service. `www.freebsd.org/security/advisories/FreeBSD-SA-13:07.bind.asc`, 2013.

[9] Open Resolver Project, 2014. `openresolverproject.org`.

[10] Routeviews Prefix to AS mappings Dataset for IPv4 and IPv6. `www.caida.org/data/routing/routeviews-prefix2as.xml`, 2015.

[11] S. M. Bellovin. A Technique for Counting NATted Hosts. In *Internet Measurement Workshop (IMW)*, 2002.

[12] K. Benson, A. Dainotti, k. claffy, and E. Aben. Gaining Insight into AS-level Outages through Analysis of Internet Background Radiation. In *Traffic Monitoring and Analysis Workshop (TMA)*, 2013.

[13] R. Beverly. A Robust Classifier for Passive TCP/IP Fingerprinting. In *PAM*, 2004.

[14] N. Brownlee. One-way Traffic Monitoring with iatmon. In *Passive and Active Network Measurement Workshop (PAM)*, 2012.

[15] M. Casado, T. Garfinkel, W. Cui, V. Paxson, and S. Savage. Opportunistic Measurement: Extracting Insight from Spurious Traffic. In *HOTNETS*, 2005.

[16] W. Chen, Y. Huang, B. F. Ribeiro, K. Suh, H. Zhang, E. de Souza e Silva, J. F. Kurose, and D. F. Towsley. Exploiting the IPID field to infer network path and

end-system characteristics. In *Passive and Active Network Measurement Workshop (PAM)*, 2005.

[17] E. Chien. Downadup: Attempts at Smart Network Scanning. `www.symantec.com/connect/blogs/downadup-attempts-smart-network-scanning`, 2009.

[18] CIA. The World Factbook: Population.

[19] Í. Cunha, R. Teixeira, and C. Diot. Measuring and Characterizing End-to-End Route Dynamics in the Presence of Load Balancing. In *Passive and Active Network Measurement Conference (PAM)*, 2011.

[20] A. Dainotti, R. Amman, E. Aben, and k. claffy. Extracting Benefit from Harm: Using Malware Pollution to Analyze the Impact of Political and Geophysical Events on the Internet. *SIGCOMM Comput. Commun. Rev. (CCR)*, 42, Jan. 2012.

[21] A. Dainotti, K. Benson, A. King, k. claffy, E. Glatz, X. Dimitropoulos, P. Richter, A. Finamore, and A. Snoeren. Lost in Space: Improving Inference of IPv4 Address Space Utilization. Technical report, CAIDA, Oct 2014.

[22] A. Dainotti, K. Benson, A. King, k. claffy, M. Kallitsis, E. Glatz, and X. Dimitropoulos. Estimating Internet Address Space Usage through Passive Measurements. *SIGCOMM CCR*, 44(1), Dec. 2013.

[23] A. Dainotti, A. Pescapè, and K. Claffy. Issues and future directions in traffic classification. *IEEE Network*, 26(1):35–40, Jan 2012.

[24] A. Dainotti, C. Squarcella, E. Aben, k. claffy, M. Chiesa, M. Russo, and A. Pescapé. Analysis of Country-wide Internet Outages Caused by Censorship. In *Internet Measurement Conference (IMC)*, 2011.

[25] A. Dhamdhere and C. Dovrolis. Twelve Years in the Evolution of the Internet Ecosystem. *IEEE/ACM Transactions on Networking*, 19, Sep 2011.

[26] Z. Durumeric, M. Bailey, and J. A. Halderman. An Internet-Wide View of Internet-Wide Scanning. In *USENIX Security*, 2014.

[27] N. Falliere. Sality: Story of a Peer-to-Peer Viral Network. `www.symantec.com/content/en/us/enterprise/media/security_response/whitepapers/sality_peer_to_peer_viral_network.pdf`, 2011.

[28] J. Goubault-Larrecq and J. Olivain. Detecting Subverted Cryptographic Protocols by Entropy Checking. Technical Report LSV-06-13, Laboratoire Spécification et Vérification, ENS Cachan.

[29] J. Heidemann, Y. Pradkin, R. Govindan, C. Papadopoulos, G. Bartlett, and J. Bannister. Census and Survey of the Visible Internet. In *IMC*, 2008.

[30] V. Jacobson, R. Braden, and D. Borman. TCP Extensions for High Performance. RFC 1323 (Proposed Standard), May 1992.

[31] E. Katz-Bassett, H. V. Madhyastha, J. P. John, A. Krishnamurthy, D. Wetherall, and T. Anderson. Studying Black Holes in the Internet with Hubble. In *NSDI*, 2008.

[32] A. Kumar, V. Paxson, and N. Weaver. Exploiting Underlying Structure for Detailed Reconstruction of an Internet-scale Event. In *IMC*, 2005.

[33] Z. Li, A. Goyal, Y. Chen, and V. Paxson. Automating Analysis of Large-scale Botnet Probing Events. In *ACM Symposium on Information, Computer, and Communications Security (ASIACCS)*, 2009.

[34] Y. Liu and Y. Yang. Analysis of P2P Traffic Identification Methods. *Emerging Trends in Computing and Information Sciences*, 4(5), 2013.

[35] A. Loewenstern and A. Norberg. DHT Protocol. `www.bittorrent.org/beps/bep_0005.html`, Jan 2008.

[36] A. Lutu, M. Bagnulo, and O. Maennel. The BGP Visibility Scanner. In *Global Internet Symposium (GI)*, 2013.

[37] B. McDanel. TCP Timestamping - Obtaining System Uptime Remotely. `seclists.org/bugtraq/2001/Mar/182`, 2001.

[38] K. McNamee. Malware Analysis Report: New C&C Protocol for ZeroAccess/Sirefef. `botnetlegalnotice.com/zeroaccess/files/Ex_14_Decl_Anselmi.pdf`, 2012.

[39] D. Moore, C. Shannon, D. Brown, G. Voelker, and S. Savage. Inferring Internet Denial-of-Service Activity. *ACM Transactions on Computer Systems*, 24(2), May 2006.

[40] A. Norberg. uTorrent transport protocol. `www.bittorrent.org/beps/bep_0029.html`, June 2009.

[41] R. Pang, V. Yegneswaran, P. Barford, V. Paxson, and L. Peterson. Characteristics of Internet Background Radiation. In *IMC*, 2004.

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

[43] M. Prince. The DDoS That Almost Broke the Internet. `blog.cloudflare.com/the-ddos-that-almost-broke-the-internet`, March 2013.

[44] M. Sargent, J. Czyz, M. Allman, and M. Bailey. On The Power and Limitations of Detecting Network Filtering via Passive Observation. In *PAM*, 2015.

[45] K. Schomp, T. Callahan, M. Rabinovich, and M. Allman. On Measuring the Client-side DNS Infrastructure. In *IMC*, 2013.

[46] The Bro Project. TCP Scan detection. `bro.icir.org/sphinx/scripts/policy/misc/scan.bro.html`, 2014.

[47] B. Van Nice. Drilling Down into DNS DDoS. `www.nanog.org/sites/default/files/nanog63-dnstrack-vannice-ddos.pdf`. NANOG 63, Feb 2015.

[48] E. Wustrow, M. Karir, M. Bailey, F. Jahanian, and G. Huston. Internet Background Radiation Revisited. In *Internet Measurement Conference (IMC)*, 2010.

[49] M. Zalewski. p0f v3: passive fingerprinter. `lcamtuf.coredump.cx/p0f3/README`, 2012.

[50] S. Zander, L. L. H. Andrew, and G. Armitage. Capturing Ghosts: Predicting the Used IPv4 Space by Inferring Unobserved Addresses. In *IMC*, 2014.

[51] M. Zhang, C. Zhang, V. Pai, L. Peterson, and R. Wang. PlanetSeer: Internet Path Failure Monitoring and Characterization in Wide-Area Services. In *Operating Systems Design and Implementation (OSDI)*, 2004.