DSCOPE: A Cloud-Native Internet Telescope

Eric Pauley
University of Wisconsin-Madison

Paul Barford
University of Wisconsin-Madison

Patrick McDaniel
University of Wisconsin-Madison

Abstract

Data from Internet telescopes that monitor routed but unused IP address space has been the basis for myriad insights on malicious, unwanted, and unexpected behavior. However, service migration to cloud infrastructure and the increasing scarcity of IPv4 address space present serious challenges to traditional Internet telescopes. This paper describes DSCOPE, a cloud-based Internet telescope designed to be scalable and interactive. We describe the design and implementation of DSCOPE, which includes two major components. *Collectors* are deployed on cloud VMs, interact with incoming connection requests, and capture `pcap` traces. The *data processing pipeline* organizes, transforms, and archives the `pcap`s from deployed collectors for post-facto analysis. In comparing a sampling of DSCOPE's collected traffic with that of a traditional telescope, we see a striking difference in both the quantity and phenomena of behavior targeting cloud systems, with up to 450× as much cloud-targeting as expected under random scanning. We also show that DSCOPE's adaptive approach achieves impressive price performance: optimal yield of scanners on a given IP address is achieved in under 8 minutes of observation. Our results demonstrate that cloud-based telescopes achieve a significantly broader and more comprehensive perspective than traditional techniques.

1 Introduction

Internet telescopes [25, 27] that monitor routed but unused IP addresses are an important tool for security researchers and practitioners. Data from Internet telescopes has allowed researchers to characterize emergent botnets [1], signals of denial-of-service attacks [28, 39], background radiation [30] and the proliferation of malware [26]. As long as adversaries believe that a *network* has potential targets, then a telescope can be useful for understanding their probes and attacks.

Yet, the way in which web services are deployed has shifted, with the use of shared compute resources such as public clouds becoming increasingly popular [8]. Because of this, the most valuable targets for exploitation have become increasingly concentrated in the share of the IP address space controlled by these providers. In turn, attackers can easily identify these address ranges and target attacks accordingly, avoiding characterization by traditional telescopes and putting deployed services at risk. Additionally, sophisticated adversaries target attacks based on signals of actual deployed services [1]; darknets, which do not emulate these services, would therefore not receive representative exploit traffic, thus limiting the utility of the data.

In this work, we present DSCOPE, a cloud-based distributed general-purpose Internet telescope. DSCOPE enables collection of representative traffic data across millions of public IP addresses by leveraging public cloud providers. IP addresses in cloud infrastructures are allocated in bulk for brief periods and, owing to the pseudorandom nature of these allocations [9, 32], new instances of DSCOPE continually receive and measure new IP addresses. DSCOPE is currently deployed across 16 regions of Amazon Web Services, and collects traffic from millions of sessions each day. Additionally, DSCOPE is quite cost-effective: the primary results from this paper can be collected for approximately 70 USD in cloud server costs. The resulting dataset allows researchers to effectively measure adversarial behavior targeted at cloud infrastructure. At the same time, as IPv4 addresses become increasingly scarce, DSCOPE allows researchers to collect representative telescope traffic without needing to retain control of large, valuable blocks of IP address space.

Cloud-based telescopes bring new measurement advantages and challenges. We explore DSCOPE's characteristics in four stages (outlined in Table 1): (1) comparison with an existing conventional telescope, (2) evaluation of how interactivity and the service lifecycle affect collected data, (3) analysis of how cloud regions and IP address ranges differ in traffic phenomena, and (4) using collected data to optimize the deployment of DSCOPE.

We compare DSCOPE with Merit's ORION telescope [25], and find improved coverage owing to both DSCOPE's vantage point and greater interactivity. While conventional telescope

Table 1: Findings of our measurement study. Metrics are detailed in corresponding sections.

	Finding	Metric
Cloud Targeting (Section 4)		
(F1)	An interactive cloud telescope receives traffic from substantially more IP addresses.	73% more traffic
(F2)	Cloud IP traffic is more variable than darknets.	95% higher σ_{IP}
(F3)	Scanners target cloud IP ranges or avoid telescopes.	450× higher than expected under H_0
(F4)	Scanners that are seen by both darknet/cloud telescopes are largely untargeted.	N/A
(F5)	Scans targeting existing telescopes are primarily random.	N/A
Interactivity & Service Lifecycle (Section 5)		
(F6)	Some scanner IPs demonstrate clearly non-random behavior.	1.7% of traffic ($p < 10^{-4}$)
(F7)	Delayed scanners leverage information from other sources to target responsive IPs.	> 90% discernible source
(F8)	Delayed scanners are not seen by existing darknet telescopes.	90% telescope avoidance ($p < 10^{-4}$)
Intra-cloud Targeting (Section 6)		
(F9)	Quantity of scanners differs across cloud regions, but intra-region variance dominates.	$\pm 0.3\sigma$ variation between regions
(F10)	Source IP variance differs between regions.	6× variation in σ
(F11)	Scanners target cloud IP addresses based on outdated data.	21% fewer scanners to 2021 AWS IPs
(F12)	Traffic to individual regions is largely consistent with untargeted scanning.	< 10% regional targeting
(F13)	Some sophisticated scanners precisely target physical regions within cloud IP blocks.	4× background rate for region/port
(F14)	Scanners show minimal preference to groups of regions in similar geographies.	0.02 lower overlap in same-geography
Optimizing Collection (Section 7)		
(F15)	Observed traffic increases over time after instance deployment, but only to a point.	67% increase
(F16)	Scanners targeting ORION are less likely to be reactive.	34% increase
(F17)	Short-lived use of IP addresses maximizes economical yield of new behavior.	< 10 min for max yield
(F18)	Extended measurement on a given IP is not necessary to achieve high coverage.	90% IP coverage at 72 minutes

traffic is largely consistent with random scanning or backscatter, our analysis shows that DSCOPE receives 450× as much cloud-targeted traffic as would be expected under random scanning. Further, DSCOPE's interactivity and movement through the IP space allows characterization of scanners reactive to the service lifecycle. Specifically, our statistical analysis identifies scanners reactive to interactivity from DSCOPE and demonstrates information-flow relationships between initial and follow-up scans.

Focusing on DSCOPE's deployment, we study how IP addresses within AWS differ in traffic phenomenon. We find that scanner targeting is relatively homogeneous across cloud regions. Surprisingly, we find that much of the observed cloud-targeted traffic is correlated with *when*, not *where*, AWS provisioned IP address space: IP addresses recently acquired by Amazon receive traffic from 21% fewer scanners than long-standing IP ranges. This suggests that, not only are scanners targeting cloud IP address ranges, but the IP lists used in this targeting are in many cases outdated or hard-coded.

Finally, we seek to understand optimal strategies for cloud telescope deployment. By estimating the overall distribution of traffic timing towards cloud services and accounting for the fixed and variable costs of deployment, we demonstrate that an adaptive telescope strategy that moves between IP addresses achieves remarkably high coverage in very little time. Indeed, our results suggest that optimal price performance is achieved in under 8 minutes, and 90% of the steady state traffic to a given IP address will be seen after only 72 minutes.

Cloud-based measurement provides a new capability for understanding scanning and attack phenomena on the Internet and opens access to data previously only available to connected organizations. By sharing our collection methodology, we anticipate that subsequent studies can use our data and techniques to answer new measurement questions and provision countermeasures more rapidly and effectively.

2 Background

Measurement-based studies of Internet traffic and systems have provided important insights on performance [6] and security [21]. Standard techniques for measurement-based study of security-related issues include: (1) instrumenting deployed systems [14, 36, 40], (2) deploying specialized systems in the network [10, 35], (3) harnessing existing data feeds [7, 44], and (4) active probing [13].

In the space of (IP) endpoint measurement, conventional techniques can be broadly categorized into *telescopes* (systems deployed on unused IP addresses that passively collect traffic from the broader Internet) and *honeypots* (interactive systems deployed on unused IP addresses that emulate real systems to elicit interesting/adversarial behavior). Telescopes have been deployed on large IP address blocks, which in turn require expensive hardware and complex techniques to manage the flood of incoming traffic [11]. Studies that have developed monitoring systems that are interactive *and* scalable [5, 47] align with our goals and inform our work.

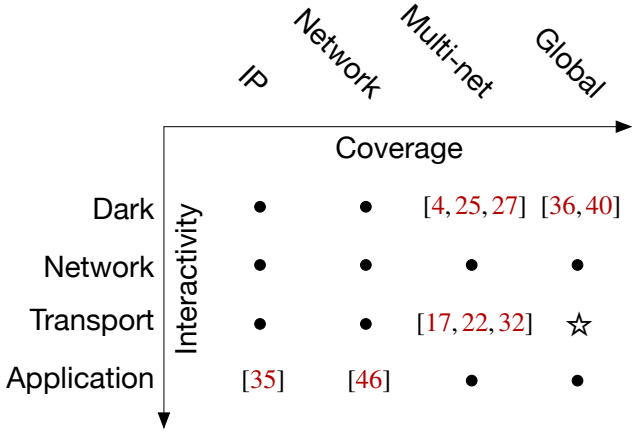

Figure 1: Taxonomy of prior work on inbound Internet measurement. This work's contribution is signified by the star.

2.1 Internet Telescopes

Internet telescopes, collections of IP endpoints specifically designed to receive and analyze network-layer behavior, have proven useful for analyzing emergent trends and security threats [28, 47]. For instance, UCSD [27] and Merit [4] have used surplus IP ranges for many years to collect traffic received from the broader Internet. The traffic collected by these approaches is referred to as *Internet background radiation*. This background radiation contains not only direct scanning by adversaries and others, but also indirect indicators of other attacks, such as backscatter from DDoS attacks. In this way, Internet telescopes have proven to be a critical component for understanding the security of the Internet.

While data from these systems has been the basis for many studies, they are limited in several ways. First, they typically do not respond to incoming traffic, which limits the perspective they can provide on attacks. Second, the contiguous nature of the IP ranges also makes them easy for adversaries to avoid. This has led to development of techniques for hiding monitors within IP address ranges that include live hosts [43]. As adversaries become more sophisticated, however, telescopes must likewise evolve to ensure continued ability to characterize their behavior.

Telescopes have also been deployed in more specialized settings, such as residential networks [36], to gain insights on targeted attacks. Here, the telescope addresses are situated with and indistinguishable from target hosts (such as vulnerable residential endpoints). Results comparing known Akamai addresses, unknown addresses in the same IP ranges, and darknet addresses showed substantially differing behavior, implying that adversaries are avoiding telescopes and targeting residential and CDN IPs.

2.2 Interactivity: Honeypots & Honeynets

While Internet telescopes can yield useful insights, interacting with traffic sent to unused addresses allows for the collection of data at higher levels of the protocol stack [23], and could cause more sophisticated actions by adversaries [35]. Interaction is typically defined by the extent to which responses are sent. Low-interaction honeypots only send an ACK to received SYN scans [1]. High-interaction honeypots emulating multiple network resources, which can expose instances of attempted lateral movement [46] by adversaries or elicit transmission of exploit packets [45]. As such, interactive honeypots (emulating a single machine) [35], and honeynets (emulating groups of machines in a network) [46] are popular approaches to measuring adversarial behavior.

2.3 Large-scale interactive measurement

Recent studies have explored interactive telescopes that can be deployed at scale. Hiesgen *et al.* [17] deployed a transport-layer interactive telescope to multiple `/24` blocks to understand Internet wide scanning at the application layer. Commercial providers have also aggregated transport-layer data across addresses to provide insights [22]. Another study [32] deployed a collection apparatus in a cloud region to explore the effect of latent configuration on cloud services. In contrast to these prior studies, and as indicated in Figure 1, our work focuses on developing an interactive system for large-scale, distributed cloud-native monitoring toward the goal of understanding malicious and unwanted activities that target these critical infrastructures.

3 Deploying a Cloud-based Internet Telescope

Based on our hypothesized shift in adversarial behavior towards targeting cloud systems, we develop DSCOPE, a cloud-based Internet telescope. DSCOPE collects traffic on unused IP addresses from a cloud provider by provisioning low-cost compute instances under those addresses and measuring inbound traffic. Here, we describe design considerations of DSCOPE, including our choice of cloud provider, collection and analysis of traffic, and resulting dataset characteristics. Ethical considerations of DSCOPE are discussed in Section 3.3.

3.1 Cloud Provider Selection

Because the implementation of a cloud-based Internet telescope is dependent on the target provider, we first analyzed major offerings to determine the most promising vantage point. Two key factors impact the suitability of a provider for deploying a telescope: (1) coverage of the overall IP space, and (2) hourly cost to control IPs. We evaluated these factors for three largest cloud providers. Relevant metrics for each provider are shown in Table 2.

Table 2: IP footprint and provisioning cost of major providers.

Provider	IPs	# /8s	Cost (USD/IP-Hr)
GCP [15]	11.5 M	34	0.005
Azure [3]	35.7 M	13	0.044
AWS [2]	134 M	82	0.0016

Between major US-based cloud providers, there is vast difference in the number and diversity of IP addresses. When deploying DSCOPE, optimizing these will provide the most general coverage of Internet phenomena. Amazon Web Services, being the earliest of the providers to offer public IP leasing, seemingly structured their offerings with less consideration for the looming shortage of IPv4 addresses seen today (e.g., allocation of public IPs to instances by default). Potentially to maintain backwards compatibility, Amazon has acquired a large portion of the total IPv4 space, controlling nearly 3× as many IP addresses as the other two providers combined. Further, since these addresses have been acquired over many years, they occupy different regions of the IPv4 address space.

The structure of cloud provider offerings also impacts the cost of measuring data on distinct IP addresses. Both Azure and GCP charge separate fees to lease public IP addresses, even when used with running instances. This, combined with higher costs of the lowest-price instances on GCP, means that compared to AWS an IP-hour of measurement costs 3× as much on GCP and 27× as much on Azure. Based on both of these factors, we concluded that Amazon Web Services was the most suitable platform to study the effectiveness of DSCOPE. After accounting for fluctuating compute costs, support overhead, and oversampling, our cloud costs for this paper's week-long study window were roughly 70 USD.

Traffic phenomena across providers In addition to cost and overall IP address space, several factors make AWS an especially strong platform for Internet measurement. For instance, Amazon's legacy emphasis on IPv4 leads to a stronger record of servers being directly exposed to the Internet in their IP ranges, increasing potential yield for adversaries. Additionally, Amazon's continual acquisition of new IP ranges (evidenced by the high number of /8s covered) implies that AWS provides a more representative view with respect to historical IP address space structures.

While we focused on a single cloud platform in this study, the high-level techniques employed by DSCOPE are applicable to other providers as well, and we expect that study of additional providers could yield new insights into behavior that are specific to those settings. For instance, adversaries targeting protocols specific to certain providers (e.g., forging traffic from Azure Service Bus to Microsoft Azure IPs) would likely only be measurable within those providers. While study of untargeted (i.e., randomly distributed across the IPv4 address space) traffic will naturally favor the lowest-cost provider, augmenting DSCOPE's vantage point with additional cloud providers and deployment scenarios is a promising direction for future work.

3.2 Implementation

Based on our study of cloud provider offerings and theoretical ability to measure traffic across their respective IP address spaces, we develop DSCOPE, a distributed network telescope designed for deployment to a public cloud. Two high-level requirements for DSCOPE were *scalability* and *interactivity*. To these ends, DSCOPE (Figure 2) consists of components to provision instances, collect network traffic, aggregate and preprocess that traffic, and analyze the resulting data.

Provisioning Servers Within our deployment target of Amazon Web Services, DSCOPE is designed to be scalable, maximizing yield of new IP addresses at minimal cost. Within each region, DSCOPE uses a provider-managed *spot fleet* to continually provision new compute instances (and therefore new IP addresses) from spare capacity on AWS Elastic Compute Cloud (EC2). These spot instances can be preempted by Amazon at any time for capacity reasons, though the `tg4.nano` instances used by DSCOPE saw a preemption rate of only 0.05% during the period studied, immaterial to the overall data collected. Spot instances run a minimal Linux distribution on 2GB of storage and 0.5 GB of memory, minimizing associated costs. Upon booting, each instance downloads the DSCOPE binary from a copy stored in each cloud region, then immediately starts collection. As a result, we estimate that each instance is billed for only 60 s of compute where network traffic is not measured. Instance management is performed by Amazon's control plane itself, with the deployment of this configuration being automated by ~300 lines of Python code.

Collecting Traffic To interactively collect network traffic across all TCP services, DSCOPE leverages network address translation (NAT) within the Linux kernel to route all incoming TCP traffic to a single service. The DSCOPE service then accepts these connections, collects the original connection information from the kernel, and can interact based on the nature of the connection. While DSCOPE is capable of arbitrary interaction, including inferring client protocols and hosting (fake) application-layer services, our current deployment (and that evaluated in this work) completes transport-layer (TCP) handshakes and emulates an unresponsive application-layer service. In this way, banner-level information is received from clients while reducing the possibility of evoking the transmission of sensitive data (e.g., deployment of an SSH honeypot might cause the transmission of real user credentials when deployed on cloud systems). Once data is collected for a given instance, it is stored encrypted within Amazon S3. Collection code is implemented in ~2500 lines of Go.

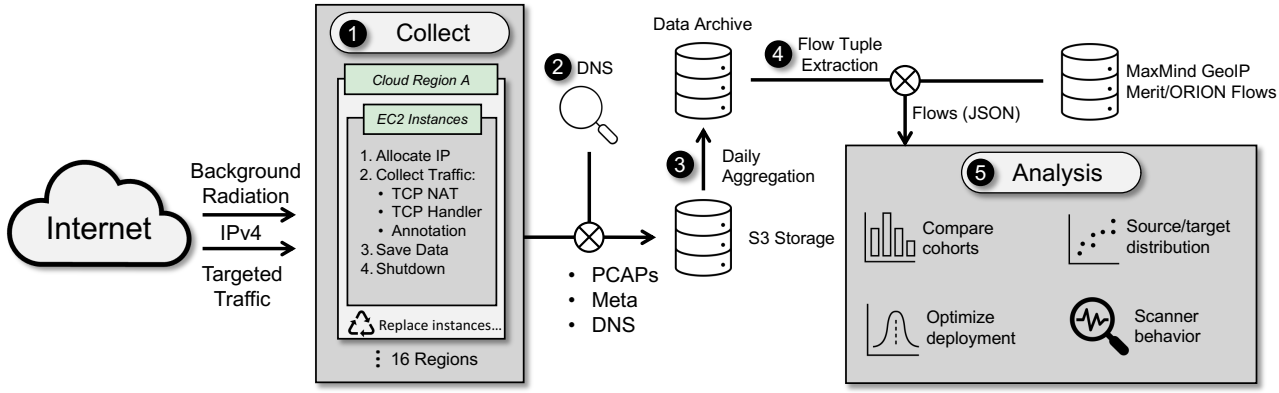

Figure 2: DSCOPE architecture. ❶ IPv4 traffic is collected by Amazon EC2 instances deployed across regions, availability zones, and IPs. ❷ Requests are dissected in real-time to perform time-sensitive analysis (e.g., DNS lookups) before being stored . ❸ Recorded traffic is aggregated daily into compressed data products. ❹ Aggregated traffic is inspected and TCP flow tuples are output, combined with network and geolocation information [24]. ❺ Data format supports premade and custom analysis routines.

Aggregation and Preprocessing To convert the thousands of raw `pcap` files generated daily by DSCOPE's collection into useful artifacts for analysis, the data is preprocessed and aggregated. Some preprocessing is done in real-time, such as resolution of certain DNS references in the traffic (since these could change in the future). `pcaps` and metadata collected by all instances are then combined daily into compressed archives.

While some analyses might involve direct inspection of `pcap` files, the majority of those discussed below focus on trends across sessions, IP networks, and geographies. To this end, we extract flow tuples from archived `pcap` files, annotate those flows with geographic and network data (such as source country and ASN) [24], and export JSON data products. We are developing processes through our institution to make these data products available to the broader community.

During collection, we found that existing `pcap` processing tools incur super-linear runtime for large files. We developed a new set of utilities to efficiently perform the `pcap` processing tasks in this section. Because the collected data is from multiple instances and IPs, tooling can leverage this to more precisely manage state and perform analysis in linear time with the number of instances. This tooling is essential for efficient analysis of the data products produced by DSCOPE, and components will be made available to researchers in conjunction with shared data. Other components are also applicable to general-purpose `pcap` processing and will be shared as open source. DSCOPE's `pcap` processing utilities are implemented in ~1600 lines of Go.

Analysis To establish the efficacy of DSCOPE in collecting representative network traffic, we perform a variety of preliminary analyses to compare our collected data with an existing Internet telescope (Merit's ORION [25]), and between cohorts (groups of telescope IPs) within the dataset. The tooling to generate the figures and statistical tests described in this paper is implemented in ~1500 lines of Python using standard data analysis libraries [16, 19, 29, 41, 42].

3.3 Ethical & Compliance Considerations

When designing any Internet measurement study, it is critical to maximize positive study outcomes while minimizing or eliminating potential harms to end-users. Because our study investigates behavior of Internet scanners, it is explicitly *not* human subjects research per official HHS recommendations [18]. That being said, any interactive Internet measurement carries risks, and so steps were taken to mitigate these to the extent possible. Through these, the benefits of DSCOPE's new vantage point can be achieved while minimizing potential harms.

Personal data collection Any publicly-routable IP address has the potential to receive sensitive information, with public cloud addresses potentially being more likely [32]. While DSCOPE does not have any privileged position that would allow it to receive more data than any cloud user, steps were still taken to protect any personal information that may have been received:

1. *Collection Server Security.* Collection servers run latest patched versions of operating systems and userspace software. Remote access is protected using existing best practices, and network data is not persisted to disk. Collection servers have write-only access to an encrypted data repository (Amazon S3), to which packet captures and initial analysis are written.

2. *Data Access.* Collected data is stored encrypted, and access control is limited to researchers working on the specific project. Security review of our cloud deployment strategies has been conducted at the institution-level, in addition to consultation with engineers at Amazon.

To ensure that DSCOPE meets the institutional standards fitting to the potential sensitivity of collected data, we worked closely with our institutional risk management department. Through a set of standards for cloud-deployed workloads, we identified, created, and implemented plans for mitigating the risk of sensitive data access. For instance, our data risk management plan includes documented steps for auditing all user/programmatic access to data, and rotation of access credentials in the event of exposure.

Controls and Infrastructure Efficiency DSCOPE's approach is strongly tuned for collection cost optimization, a goal that is often at odds with the security and compliance controls offered by cloud providers. In our case, DSCOPE's continuous allocation of servers leads to substantial ($\sim 100\times$) cost overhead when naively deploying cloud controls. To achieve cost efficiency and data security we use a split-account architecture. DSCOPE collection instances are deployed on a special-purpose *collection* cloud account with extremely limited capabilities (no permanent data storage) and reduced controls. A separate *analysis* account is used to retain, aggregate, and analyze collected data. DSCOPE servers in the collection account are given tightly-scoped, *write-only* access to storage in the analysis account. When reasoning about this architecture, we consider the collection of individual DSCOPE instances to be less sensitive (as they are simply recording traffic that any EC2 user would observe), but the aggregate data (in the analysis account) as potentially sensitive.

Impact on clients We explicitly design our collection architecture to minimize impacts on clients while maximizing data collection. Here, we use an approach suggested by [31], wherein they describe modeling client impact at the protocol level. Specifically, DSCOPE collection servers perform TCP session handshakes but do not reply with any application-layer data. This behavior is a strict subset of what would be expected from a legitimate server in the event of network or system-level failures. In addition to previously-discussed protections, here we also considered the risk of frozen connections causing denial-of-service to clients. For this reason, we explicitly terminate TCP sessions without sending response data after 10 seconds, such that clients do not hang indefinitely waiting for server responses.

Impact on Cloud Providers Because DSCOPE leverages leased resources, there could also be risks to the cloud provider's infrastructure itself. DSCOPE allocates thousands of servers each day, and so steps are taken to limit the impact of this allocation:

1. *IP Starvation.* Cloud providers age out IP addresses before reuse [32], so allocating too many addresses at once could deplete this pool. We ensure DSCOPE smoothly allocates resources to prevent large spikes of IP consumption.

2. *Compute Instance Starvation.* We use preemptible instances to ensure that DSCOPE does not impact available compute capacity. While this means that data is occasionally unavailable in some zones, this prevents impacts on other cloud tenants. Additionally, because spot instances are continually released and reallocated, the cloud provider can preference denying these new allocations instead of preempting running spot instances from other tenants–behavior we have empirically observed to be the case on AWS. In this way, DSCOPE does not prevent non-spot allocation by other tenants, and is unlikely to cause increased preemption of other tenants' spot instances.

3. *Network Amplification.* DSCOPE has been evaluated to ensure it does not amplify received traffic. DSCOPE TCP responses are never larger than inbound traffic, protecting cloud provider network resources and other network services against denial of service.

3.4 Experimental Dataset

DSCOPE has collected data continuously on Amazon Web Services since 2021 and has been deployed across regions globally since October 2022. During this period, it has collected data on 4.6 M AWS IPv4 addresses spanning 105 k `/24` subnets. We sample one week's worth of data from October 11-17, 2022 for comparative analysis in this work. Because behavior of remote hosts varies over time, a shorter study window reduces the incidence of such behavior changes being interpreted as a single composite behavior. We emphasize that the complete history of data collected by DSCOPE can be useful for other purposes (especially time-series analyses). As data continues to be collected, we look forward to myriad new insights on network behavior.

Because IP allocation and spot instance availability is random, the amount of data collected varies day-to-day and region-to-region. To allow for uniform comparison across regions and with existing datasets, we subsample allocated servers such that each sample has a unique IP address (for sample independence), and exactly 750 unique IP addresses are allocated per region per day (125 IP-hours of data collected per day). In total, data collected from 84 k IP addresses are characterized in the study, or 14 k IP-hours of overall data. For comparison, Merit's telescope collected 80 M IP-hours of data during the study period.

4 Cloud vs. Darknet Telescopes

Given DSCOPE's new perspective, the most immediate question is one of coverage: how does the traffic received by DSCOPE differ from that received by a conventional network telescope? To answer this, we examine the quantity of scanners observed, and how this differs across the address

Table 3: Characteristics of collected data. Counts refer to number of unique source ASNs and IPs, and total number of TCP sessions seen. ORION refers to comparison data taken across all ORION IPs during the study period.

VP	ASNs	IPs	Sessions	$\overline{IP}$	σ_{IP}
ORION [25]	23.4k	2539k	21.9B	40	13.9
DSCOPE	9.1k	190k	15207k	69	27.1
ap1	2.8k	35k	966k	69	14.2
ap2	4.2k	47k	1009k	72	17.7
ap3	3.1k	38k	859k	62	31.5
ap4	4.0k	49k	1133k	72	77.2
ap5	2.4k	30k	852k	64	12.2
ca1	2.3k	28k	905k	70	13.2
eu1	3.1k	33k	982k	73	21.6
eu2	2.4k	32k	845k	64	25.9
eu3	2.7k	32k	1026k	65	13.2
eu4	2.6k	34k	973k	70	15.2
eu5	2.6k	32k	953k	69	13.3
sa1	2.6k	30k	882k	66	12.8
us1	2.5k	31k	908k	68	12.0
us2	2.8k	35k	1026k	73	35.3
us3	2.8k	34k	908k	71	13.5
us4	2.8k	35k	981k	77	17.2

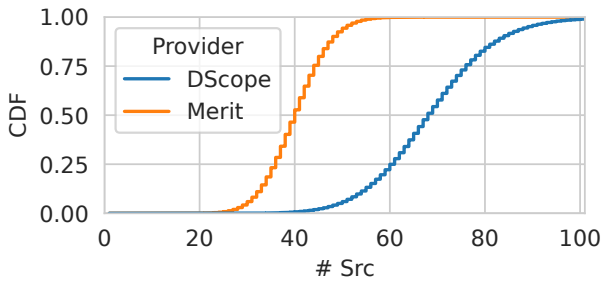

Figure 3: Distribution of scanner IP quantity seen by each telescope IP during a given study window.

space of each telescope. From this, we can infer the underlying behavior of scanners in targeting regions of IP space and determine the relative strengths of each vantage point.

4.1 Quantity of Observed Scanners

To compare the quantity of scanners observed by cloud and darknet IPs, we first segment ORION into windows of 10 minutes each (the same duration cloud instances measure). In each window, we count the number of unique source IP addresses that contact the instance. Numerical results are shown in Table 3.

Finding 1 - *An active cloud telescope receives traffic from substantially more unique IP addresses.* Looking at the CDF of

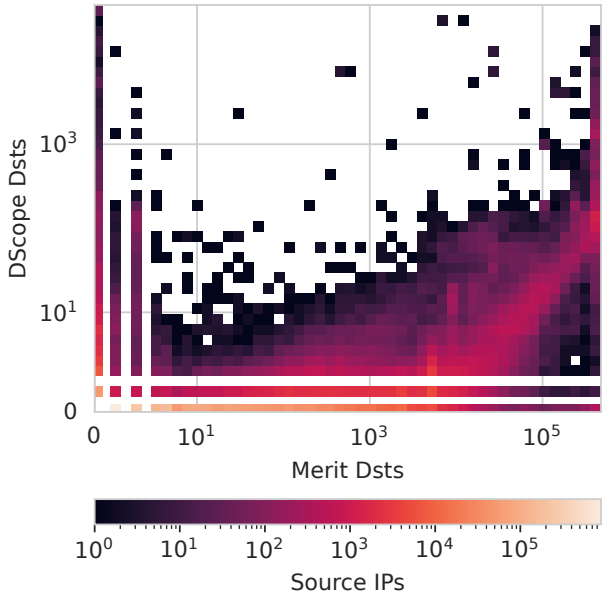

Figure 4: Distribution of DSCOPE vs ORION IPs contacted by scanners. ORION benefits from its large number of unique IPs, while DSCOPE receives cloud-targeted traffic not seen in a conventional telescope. As observed in prior work [36], the linear relation seen on the right is a natural consequence of the varying footprint of each telescope.

unique source IPs ($\overline{IP}$) across collection windows (Figure 3), the mean cloud instance receives traffic from 73% more IP addresses than the mean darknet IP in a given window.

Finding 2 - *Cloud IP traffic is highly variable.* As with prior work [36], we see a low-variance distribution of scanning IP addresses towards darknet IPs ($\sigma_{IP} = 13.9$), with quantity of sessions consistent with UCSD-NT. In contrast, IPs monitored by DSCOPE are more variable ($\sigma_{IP} = 27.1$).

While previous work [36] has shown the benefit of vantage point for improving traffic coverage, ours is the first such work to demonstrate this in the cloud setting. We next aim to characterize how scanner behavior and targeting leads to this improved coverage.

4.2 Telescope Avoidance & Cloud Targeting

Next, we compare targeting of ORION and cloud IPs as measured by DSCOPE (Figure 4). For each source IP, we compute the number of cloud telescope IPs and ORION IPs contacted. While we see a trend of randomized or Internet-wide scanning consistent with prior work [36], the histogram additionally shows a clustering of clients that connect to large numbers of cloud IPs without contacting the conventional telescope. While this prior analysis was enabled by Akamai's large residential IP footprint, studies of cloud traffic have

historically had limited footprint. In contrast, DSCOPE's larger footprint of short-lived instances allows for a novel, statistically rigorous study of targeting in the cloud setting.

To characterize whether scanners target a specific telescope, we segment scanner IPs into three bins: (1) scanners that contacted only DSCOPE (2.8%), (2) scanners that contacted only ORION (92.7%), and (3) scanners that contacted both (4.5%). For sources that contacted one telescope, we can compute the probability that such behavior would occur under random scanning. For sources that contact both, we can look at the overall distribution to infer the type of scanning behavior.

To understand what portion of single-telescope traffic is by chance (i.e., due to a random scanner happening to select one telescope's IPs) as opposed to some form of targeting, we perform a statistical test under the null hypothesis of random targeting. Under this null hypothesis, the number of IPs contacted from each telescope is a Bernoulli process, with the probability of contacting each telescope proportional to the size of that telescope. In the case of our experiment, ORION measures 80 M IP-hours, while DSCOPE measures 14 k IP-hours. The number of times a given telescope t_a is contacted instead of t_b is then Binomial-distributed as

$$T \overset{H_0}{\sim} B\left(n, \frac{|t_a|}{|t_a|+|t_b|}\right),$$

where n is the total number of IPs contacted and $|t|$ is the number of IP-hours observed by a given telescope. For a given scanner that contacts n IPs from t_a and none from t_b, the probability under the null hypothesis (and therefore p-value in rejecting the null-hypothesis) is

$$P(T = n | T > 0).$$

The conditional distribution arises because we characterize only those scanners that connect to the telescope at least once. We then perform a hypothesis test for each IP to determine the confidence that the behavior is actually targeted scanning.

Finding 3 - *Scanners target cloud IP ranges or avoid telescopes.* Of the 73 k IP addresses that only connected to DSCOPE, 14.9 k were statistically significant for targeting at $p < 10^{-6}$. In contrast, of the 2.4 M IP addresses that connected to only ORION, 869 were statistically significant for targeting. Longer-term data collection can prove targeting from additional scanners with high significance.

We can also reason about cloud targeting by the overall number of cloud-only scanners compared to what would be expected under random scanning. Modeling each observed scanner as above, we would expect to see an average of 162 scanners that only contact DSCOPE. In reality, we see 73 k, 450× expected. In contrast, ORION sees 96% of its expected number of distinct scanner IPs in our study. Our data strongly suggest that cloud IPs see large amounts of distinct traffic that either targets the cloud or avoids darknet telescopes.

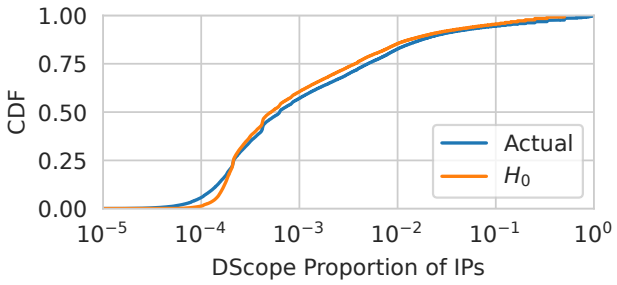

Figure 5: Null-hypothesis H_0 vs. Actual distribution of scanners seen by both telescopes.

4.3 Distribution of Random Scanning

Next, we look at the bin of scanners that contact both telescopes. Here, we hypothesize that such scanners are primarily the results of random scanning. For each IP, we look at the portion of destination IPs observed by DSCOPE and compare with the expected distribution under the null-hypothesis (random scanning) based on the total number of IPs contacted.

Finding 4 - *Scanners that are seen by both darknet/cloud telescopes are largely untargeted.* Results (Figure 5) show that actual scanner behavior is strongly consistent with random untargeted scanning. The slightly longer tails could be linear scanning [46], where the locality of each telescope leads to increased tail proportions.

Finding 5 - *Scans targeting existing telescopes are primarily random, rather than sequential, across the broader IP space.* Owing to the diverse nature of IPs measured by DSCOPE, our dataset contains 7.4 k telescope IP addresses in the same `/8` subnet as ORION (many directly following it in IPv4 space). Despite this, DSCOPE IPs in this `/8` received no more shared traffic with Merit than the median across the entire telescope. The median DSCOPE IP received traffic from 26 scanner IPs that also contacted ORION, and the population in the same `/8` had the same median 26 shared scanners. Consistent with prior observations, this suggests that the majority of unique scanner traffic received by conventional telescopes is random in nature. This observation also suggests increased adoption of newer scanning tools (such as ZMap [13]), which randomize target IP addresses throughout the scan. Notably, this finding is uniquely enabled by Amazon's sharing of the `/8` block with Merit's ORION telescope (along with DSCOPE's high coverage of this block), and could not be performed on another provider or with existing techniques.

4.4 Controlling for Cloud-Specific Phenomena

DSCOPE's vantage point nearby real deployed cloud services–both nearby in IP space and across time because of IP reuse [32]–is a compelling advantage of representative measurement. That being said, it is important to understand how IP reuse in particular affects aggregate study results. To that end, we next seek to evaluate our comparative results when accounting for the effect of IP address reuse and cloud services.

To achieve this validation, we need to filter out traffic that could be caused by these phenomena. While prior work has developed methods for isolating IP reuse for study [32], we can also apply these steps in reverse to remove this data from the dataset. Motivated by this work, we filter out two types of traffic in this section:

- *Cloud-sourced traffic.* We remove all traffic sourced from IP addresses within AWS. This is likely to be sourced from cloud services such as DNS, CDNs, or load balancers.
- *Single-target sources.* We remove all traffic that only contacted a single DSCOPE IP address and did not contact Merit's ORION telescope. End-users that connect to services because of IP reuse and latent configuration are likely to fall within this bucket.

As a result of our filtering, we are left with 136 k (vs. 190 k) source IPs contacting DSCOPE. We re-run experiments for findings 1-5 (comparing DSCOPE with ORION) to determine which differences are still present.

Results While overall traffic volume comparisons remained largely unchanged ($\overline{IP} = 68$ vs 69 in the overall data), we do observe differences in the variability of this traffic. Whereas $\sigma_{IP} = 27.1$ in the overall data, this is reduced to 14.5 when filtering out possible IP reuse traffic, barely above that seen in ORION. This result aligns with expectation, as the cloud IP addresses that receive large amounts of traffic–and therefore increase variance–likely do so due to IP reuse and latent configuration. This reduction in variance provides promising validation that our filtering is working as intended.

Filtering possible cloud targeted traffic results in minor differences in cloud targeting results, though the overall conclusions remain unchanged: rate of DSCOPE-specific traffic (F3) is 125× expected under random scanning compared to 450× in unfiltered data. Note that, because our filtering is conservative, the actual rate of cloud targeting likely lies between these numbers. Results for scanners seen across both vantage points and those seen in the same `/8` as ORION were unchanged.

Overall, our analysis of bulk traffic distribution confirms that the traffic phenomena observed on cloud IP address ranges is substantially different from that seen by conventional telescopes, including more and more varied traffic, and evidence of cloud targeting or telescope avoidance. Whereas traffic received by darknets is largely consistent with untargeted scanning, cloud IPs receive over two orders of magnitude more targeting scanning compared to expectations. In addition, while DSCOPE's approach provides for collection of cloud-specific phenomena such as IP reuse, this does not detract from its utility as a general-purpose telescope. From this, we argue that cloud-deployed telescopes are an essential tool for representative coverage of the behavior of modern scanners.

5 Telescope Service Lifecycle

Unlike network telescopes deployed via routing of darknet IP addresses [4], cloud-based telescopes can leverage interactivity to elicit further scanner behavior, as well as move around the IP address space to achieve higher coverage or measure scanner responses. Here, we evaluate how the lifecycle of deploying interactive telescope services improves coverage, and also what this can tell us about the reactive behavior of scanners over the lifetime of deployed services. To this end, we posit modes of scanner behavior, then develop statistical tests to characterize the prevalence of these modes in DSCOPE's dataset. After classifying behavior, we can compare the sources of these against Merit's ORION telescope to understand which phenomena are visible under existing approaches.

5.1 Models of Client Behavior

Scanners connecting to cloud honeypots follow some policy to target instances. Some example policies are:

- *Random.* A client could connect to instances randomly via a Poisson process, with some arrival rate λ. Because DSCOPE's IP addresses are allocated randomly [32], linear scans [46] would also appear random with respect to the telescope IPs.
- *Interval.* A client can connect at routine intervals to measure changes over time. Certain cloud services such as health checks [37] behave this way.
- *Delayed.* A client could perform an initial basic scan, then later (potentially from a different address) run a more sophisticated exploit. The Mirai botnet [1] generally identifies vulnerable hosts and reports to a command-and-control server for later exploitation.

In the case of delayed and interval scanning, some minimum observation period is needed to fully characterize the traffic. In many cases, this period may be too long for a cloud-based telescope to achieve coverage economically (e.g., a client connecting once per day would not necessarily be seen in a shorter collection duration). However, in the case of random scanning we can characterize this behavior statistically and determine optimal server allocations to

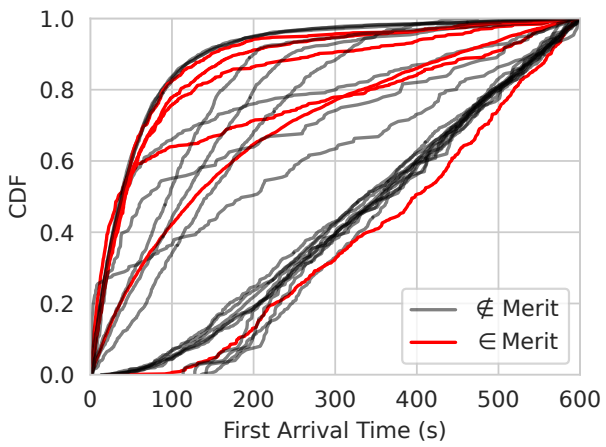

Figure 6: CDFs of first-arrival-times from most aberrant source IPs (by KS-test p-value). Lower cluster represents delayed scanning, while upper cluster represents leading traffic.

maximize coverage (Section 7). Note that, even when traffic arrival times follow some pattern, IP addresses are allocated randomly, so the arrival time is still random with respect to the time IPs are allocated.

Characterizing Random Behavior Because random client behavior is the most readily modeled, we begin by validating the assumption that observed client behavior follows this distribution. Suppose some client (e.g., a source IP address or ASN) connects randomly via a Poisson process to a subset of the IPs observed. For a given IP address allocated for measurement, the time-to-first-contact from that client will follow an exponential distribution. Further, suppose we measure a given IP for a fixed duration m. For clients that *do connect* to our IP address within that duration, the time they are first observed will follow a right-truncated exponential distribution [12] with the following density function:

$$f_T(t) = \frac{\lambda e^{-\lambda t}}{1 - e^{-\lambda m}} \qquad (0 <= t <= m)$$

where m is the maximum collection duration. If a client is not observed on a given IP address during the measurement window there are two possibilities: (1) the client would have connected at a future time $t > m$, or (2) the client would never have connected. Given this distribution, and the times of first observation for some client on all measured IPs, we can perform a parameter estimation of λ using maximum-likelihood-estimation.

Identifying Non-random Behavior With our model of random behavior established, we identify when client traffic does not follow this pattern. For a given client (IP address or ASN), we compute a null-hypothesis arrival time distribution using MLE, then perform a Kolmogorov-Smirnov test[1] of arrival times against that distribution. This statistical test evaluates the null hypothesis that the arrival times are exponential-distributed with the estimated parameter, and low p-values reject the null hypothesis and suggest the data is not exponential-distributed. Because our IP allocations are randomly distributed in time, a client showing non-exponential behavior *must* be leveraging information about the server before connecting, such as traffic measured by another client or targeting a specific IP address range. In this way, DSCOPE's collection architecture offers a unique opportunity to rigorously characterize the time-series behavior of scanners.

Finding 6 - *While the majority of traffic is consistent with random scanning, some IPs demonstrate clearly non-random behavior.* Applying our test statistic to inbound traffic finds that the vast majority of source addresses (99.99%) exhibit Poisson scanning. 25 addresses reject the null hypothesis at $p < 0.0001$, implying that scanning activity is not Poisson distributed. Plotting these first-arrival-times from these IPs (Figure 6) shows a stark contrast with expected Poisson behavior. Despite being only 0.01% of source IPs observed, these sources account for 1.7% of overall traffic. It is likely that additional traffic is also non-random and would be identified by analyzing DSCOPE's data over a longer period.

We see two distinct modalities of non-Poisson traffic:

1. *Delayed scanning*. Traffic is delayed from the expected distribution. This may be caused by the scanner reacting to scans from some other source. For instance, a command-and-control server for a botnet could receive initial scanning results and then perform more targeted scans. 11 sources were statistically significant delayed scanning.

2. *Leading traffic*. Traffic exhibits high-rate Poisson behavior, but also has lower-rate behavior. The most likely explanation of this is that the scanner is scanning different IP ranges at differing rates. 14 sources were statistically significant for leading traffic.

Leading traffic shows substantial overlap with scanner IPs seen by Merit. As this traffic is not reactive to service lifecycle, a non-responsive telescope could still receive such traffic due to heterogeneous targeting of the IPv4 address space by a scanner. However, the ability to characterize such traffic is only possible because of DSCOPE's varied IP address footprint.

[1]Note that while use of a KS test against distributions with estimated parameters is not valid in general, the induced error for an exponential distribution has been well-characterized and is not material at the p-values used [33].

Table 4: Instances of delayed scanning

ASN	# Src	# Dst IPs	# Dst Ports
34665	1	447	12
49505	1	390	1 (3389)
54098	3	109	1 (18080)
56046	4	2667	630
399629	2	8151	2 (8080/8443)

5.2 Case study: Delayed Scanning

To better understand delayed scanning, we perform further analysis on the 11 source IPs that were statistically significant in our dataset. These IPs fell into 5 ASNs (Table 4). Some ASNs targeted a variety of ports, with others targeting just one or two.

Given these scanners exhibit statistically significant delayed scanning, there must be some source of information used to target DSCOPE's IPs. To attempt to isolate this source, we look at each scanner's relationship with other IPs contacting DSCOPE. For each other scanning IP, we look at what proportion of contacts from that IP were then followed by contact from the delayed scanner. A high proportion implies that information from the original scanner is used by the delayed scanner to target instances. We additionally correlate delayed scanners to IPs seen by ORION to further understand the effect of interactivity.

Finding 7 - *Delayed scanners leverage information from other sources to target responsive IPs.* Of the 11 delayed scanner IPs, 10 had at least one related IP address that consistently contacted the telescope before the delayed scanner. For instance, a single source IP contacted 5.6 k DSCOPE IPs, with 82% of those being followed by a delayed scanner (this number is unsurprisingly under 100%, as some delayed scans would fall outside of the collection window). In this way, DSCOPE allows us to measure information flow between scanning parties. Our results also offer a generalization over those shown by prior work [17], which demonstrated two-phase scanning sourced from a single IP address but did not analyze the relationship between scans across source IPs. Unlike single-IP two-phase scanning, determining relationships across IPs is uniquely enabled by DSCOPE's large and time-varying IP address footprint.

Finding 8 - *Reactive scanners are not seen by existing darknet telescopes.* Of the 11 addresses that reject the random null hypothesis for delayed scanning, one contacted Merit's telescope during the study period (further analysis of this IP showed a separate, untargeted scan from the same source that contacted ORION). This is likely because initial probes from other IP addresses did not receive responses and so follow-on scanning was not conducted. The ability to collect reactive, cloud-targeted, and darknet-avoiding traffic is a key advantage of DSCOPE over conventional telescopes.

6 Examining Intra-cloud Targeting

In addition to interactivity and movement, a cloud-deployed telescope also has the ability to measure IP addresses with varying geography, previously-deployed-services, and ownership history (as cloud providers have expanded their offerings over time). These added dimensions can also allow us to infer when scanners are using cloud-specific strategies (e.g., regional, geographical, or service-based targeting) as opposed to naively scanning cloud providers along with other endpoints. Throughout this section, statistical analysis is based on targeted IP addresses, rather than number of connections or similar phenomena. This approach is resistant to the noise observed on individual cloud instances and is uniquely enabled by DSCOPE's large and dynamic IP address footprint.

To this end, DSCOPE samples traffic from 16 cloud regions, 30 /8 IPv4 subnets, and on IPs acquired by Amazon over 6+ years. Our comparative analysis is scoped to 84 k IP addresses over 7 days, though DSCOPE collects data continuously across many more addresses. This approach ensures that our results are not due to sampling error within the IP space of a single region, prefix, or block of IP address acquisitions, but instead any discrepancy between cohorts must be due to actual differences in client behavior.

6.1 Quantity of Observed Scanners

We first perform a coarse comparison between regions based on the overall number of scanners observed. For each region, Table 3 contains the mean and standard deviation of source IPs contacting each instance.

Finding 9 - *Quantity of scanners differs across cloud regions, but intra-region variance dominates.* IP addresses on the Internet as a whole vary greatly in received traffic based on the services deployed, and we see similar trends in cloud regions for individual IP addresses. All regions received within $\sim 10\%$ (0.3σ) of the global mean number of source IP addresses, implying that the variance within regions has more of an effect on data received by a given IP than the region measured.

Finding 10 - *Traffic variance differs between regions.* Some regions exhibit much higher variance than others (e.g., ap3, ap4, us2, eu2). Looking at eu2 in particular (Figure 7), we see a clear bimodal distribution, implying that some phenomenon within regions is impacting received traffic.

Effect of IP Prefix While regions differ slightly in their mean traffic, and more substantially in the variability of that traffic, it is not immediately clear why this is the case. To determine whether this variation is due to differences in scanner behavior across geographies, we must first eliminate other possible sources of bias, especially the structure of IP address space itself. To this end, we segment DSCOPE IPs by

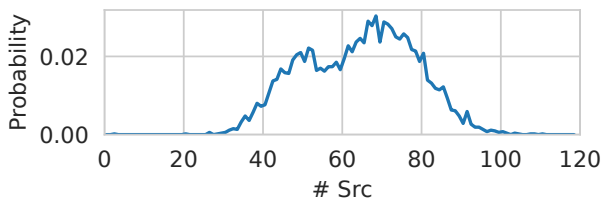

Figure 7: Source IPs contacting each DSCOPE IP in `eu2`.

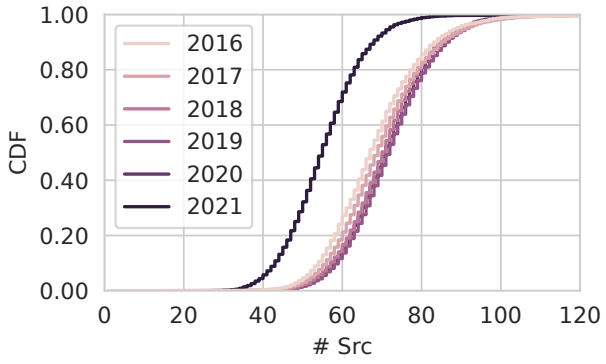

Figure 8: Number of sources contacting DSCOPE IPs based on year addresses were added to AWS. IPs more recently added receive substantially less traffic than those held for longer. Densities align with those seen in Figure 7.

their subnet. Initially, looking at just the `/8` prefix of DSCOPE IPs, we see a much higher variation across these prefixes than across geographic regions. Two possible reasons for this could be (a) residual scanning behavior targeted at previous owners of the IP ranges, or (b) scanners targeting the cloud based on outdated information on IP ranges. To evaluate this second possibility, we determine for each DSCOPE IP address the date it was first advertised by Amazon as a cloud IP.

Finding 11 - *Many scanners target cloud IP addresses based on outdated lists.* Segmenting DSCOPE's IPs by year added to AWS (Figure 8) shows a striking difference in scanning behavior based on age of IP. For IP addresses added prior to 2021, we see materially identical scanning patterns. However, for IP ranges added in 2021, the median number of distinct scanners is reduced by 21%. This strongly suggests that there is a group of scanners targeting cloud IP addresses, but using stale IP address lists. Such behavior could be explained by a distributed system such as a botnet without a centrally-distributed target list.

6.2 Sources of Received Traffic

Next, we analyze the specific scanners that are interacting with cloud regions. Here, we can use a similar methodology

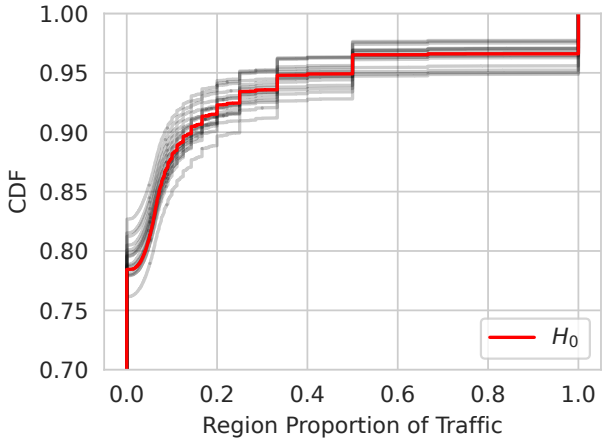

Figure 9: CDF of region proportion for each AWS region measured by DSCOPE. Most regions closely follow the null hypothesis, with two exceptions. Note the range of the y axis: over 70% of scanners globally do not contact a given region.

to that of Section 4, comparing actual proportions of traffic to each region with results under the null hypothesis of random scanning. In this case, each region is compared against all others as a baseline. The resulting distributions of region-targeting are shown in Figure 9.

Finding 12 - *Traffic to individual regions is largely consistent with untargeted scanning.* Based on a regional distribution under random scanning, we would expect 3.6% of scanners to exclusively target any given region. 14 of 16 regions fall within 2% (overall) of this figure, and though this is a statistically significant difference, region-targeted traffic accounts for at most 9% of overall sessions (as 91% of traffic is sourced from scanners that connect to multiple regions).

Case study: Regional Targeting For two of the regions (`ap2`, `ap4`), region-exclusive traffic is much higher than expected under the random sampling hypothesis. Closer analysis finds that these are largely due to large-scale targeted scanning operations against port 445 (NetBIOS/ActiveDirectory) in the affected regions. Discounting this, all regions see exclusive targeting rates within 2% of expectations.

To better understand the targeting of this traffic, we can analyze the trends of this traffic across regions and IP prefixes, as well as traffic sources. Scanning traffic against region `ap2` starts before the beginning of data collection and continues until November 15. During this time traffic is elevated to 4× ambient in the region, with the additional traffic being sourced from a wide variety of ASNs and countries.

Finding 13 - *Some sophisticated scanners precisely target physical regions within cloud IP blocks.* Plotting a Hilbert diagram of telescope IPs within AWS's `3.0.0.0/8` block

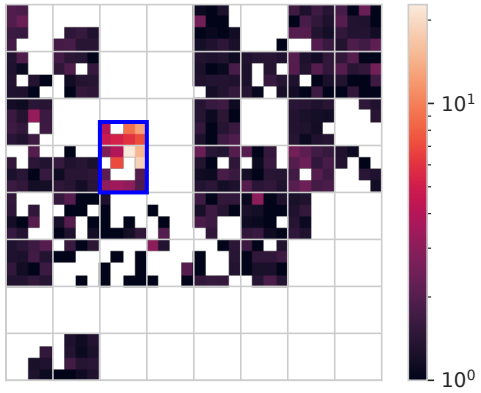

Figure 10: Hilbert diagram of traffic towards port 445 seen by DSCOPE `3.0.0.0/8` IP addresses during the study. IPs assigned to the `ap2` region are highlighted in blue. White areas were not measured by DSCOPE.

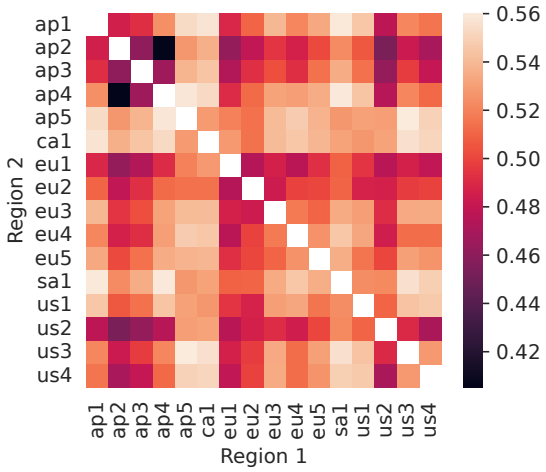

Figure 11: Overlap of source IPs observed between regions.

(Figure 10) shows elevated scanner traffic in one region, with no spill-over to neighboring parts of the IP space. As this traffic is sourced from many IP addresses across geographies and has defined time ranges, it is highly suggestive of an adversary centrally controlling many endpoints and using published AWS IP ranges to specifically target regions.

6.3 Geographic Targeting

To evaluate whether scanners preference geographic groups of regions, we analyze the overlap coefficient between scanning IPs targeting each region. Given regions with correlated scanning behavior, we would expect to see higher overlaps between these regions than other pairs in the data.

Finding 14 - *Scanners show minimal preference to groups of regions in similar geographies.* Our results (Figure 11) show no notable increases in overlap between regions within the same geography. Looking at the three geographies with multiple regions (`ap`, `eu`, and `us`), the mean overlap coefficient between same-geography regions is 0.50 vs 0.52 for different-geography regions–further demonstrating the lack of same-geography scanning preference.

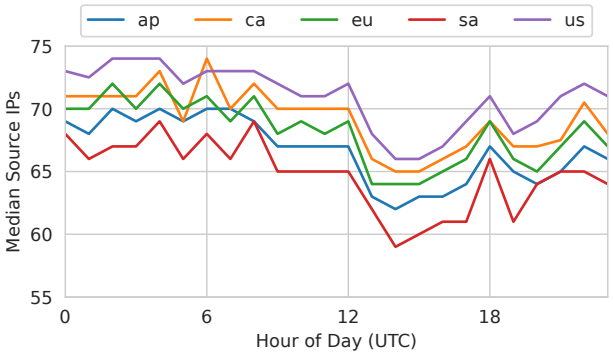

Figure 12: Median unique source IP addresses per telescope IP by hour of day (UTC). Global regions show similar aggregate trends regardless of local time zone.

6.4 Time Variance Across Regions

Cloud regions exist largely to position compute resources near end-users, and so legitimate traffic to these regions would be expected to align with the waking hours of nearby users. However, in scan traffic to cloud IPs we see no such correlation. Figure 12 shows median inbound traffic across studied geographies. From this, we see clear parallel trends across geographies, with minimal time-based regional preference. This suggests that scanning is being performed across regions without regard to underlying geography. Further, the lack of geographic dependence suggests that the bulk of data received by our telescope is some form of scan traffic, rather than legitimate clients attempting to connect to that region.

7 Efficient Cloud Measurement

Because DSCOPE relies on achieving high coverage through random sampling of the cloud IP address spaces, two questions arise compared to existing telescopes and cloud-deployed honeypots: (1) do short-lived deployments to IP addresses obtain adequate coverage of the phenomena targeted at that IP, and (2) how should DSCOPE deploy to addresses to ensure maximum coverage at minimum cost? To characterize these, we leverage our prior observation that the bulk of traffic observed by DSCOPE is randomly distributed across time. From this, we can characterize the distribution of scanning behavior over time and estimate the coverage achieved by telescopes of limited collection

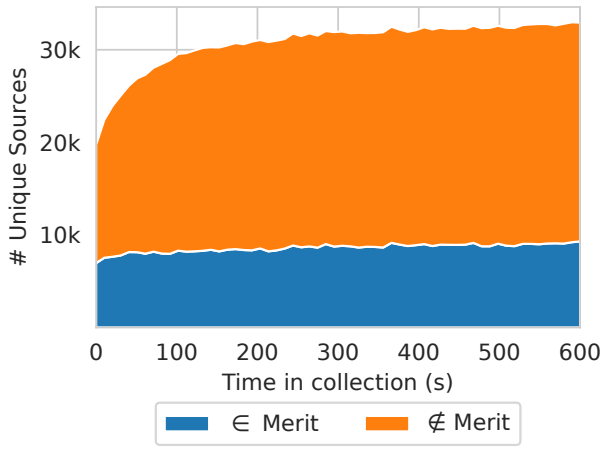

Figure 13: Number of unique scanner IPs seen by DSCOPE over 10-minute collection windows. Source IPs also seen by Merit's ORION show minimal reactive behavior.

duration. These time-series results across collection windows are only achievable because of DSCOPE's dynamic footprint, a key benefit of cloud-based telescope deployments.

Quantity of scanners over time While the majority of traffic seen by DSCOPE (and other telescopes) is untargeted and non-reactive to service lifecycle, we would still like high coverage of reactive traffic as well. To characterize this, we bucketize scanner IPs over the course of DSCOPE's collection window and compute how many unique IPs are seen over time.

Finding 15 - *Quantity of observed scanners increases over time after instance deployment, but only to a point.* Results (Figure 13) demonstrate that each DSCOPE collection IP receives traffic from progressively more distinct scanners through the collection lifecycle. Here, we see a drastic increase in unique scanners seen in the first few minutes, followed by steady-state behavior. The overall rate of unique scanners increases by 67% for all sources, and by 85% for scanners that were only seen by DSCOPE. While this demonstrates the advantages of DSCOPE's interactive collection approach for maximizing coverage, it also shows that limited collection duration is needed to reach steady state behavior on a given IP. Based on this, we argue that beyond a point it is more beneficial to collect data on new IPs than continue to observe the steady-state behavior on just one.

Finding 16 - *Scanners targeting ORION are less likely to be reactive.* Figure 13 also demonstrates that the behavior of scanners targeting ORION exhibit less response to services deployed at targeted addresses (34% increase). This implies not only that they are less targeted based on information from other addresses (lagging first contacts, Section 5), but also that these scanners do not exhibit follow-on scanning such as

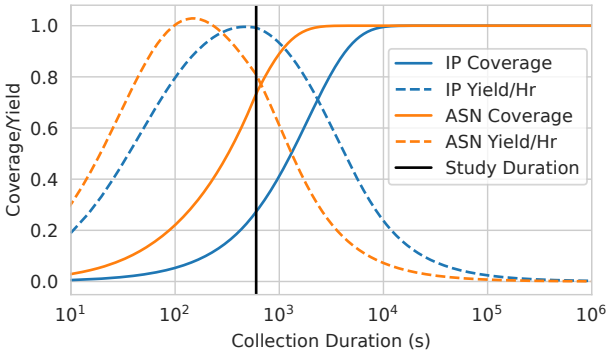

Figure 14: Coverage and relative yield of distinct traffic over varied collection durations. Results for less than 600 s (to the left of the black line) are empirical, and results to the right are extrapolated via parameter estimation. Empirical results demonstrate that maximum yield occurs in under 10 minutes for both cases, and high coverage of overall scanning behavior can still be achieved with limited collection duration. Note that relative yield is computed as a fraction of estimated maximum, so values over 1 are possible when the empirical distribution does not perfectly fit the estimated curve.

attempting further exploit payloads on responsive services. In contrast, scanners targeting cloud IPs exhibit increased scanning in response to signs of a deployed service, in addition to leveraging signals from scanning activity of other IPs.

Maximizing Yield of Random Traffic Based on our measurements that the vast majority of client behavior is random, we can determine optimal duration of data collection. Deploying a cloud telescope is ultimately a matter of cost, with the goal of increasing yield of interesting behavior per IP address hour purchased. For random connections, the yield y of distinct source behaviors for collection duration d follows an exponential distribution

$$y = 1 - e^{-\lambda d},$$

where λ is the bulk rate parameter for new behavior. This can be measured empirically for new source IP addresses ($\lambda = 1/1884$ s) or ASNs ($\lambda = 1/452$ s) by fitting parameters to the truncated exponential distribution (as in Section 5). Collection servers also incur a fixed startup duration d_s, during which they incur costs but do not collect data, experimentally observed in our setup to be $d_s \approx 60$ s. From this, we can compute the yield of new behavior per compute hour y' for a given collection:

$$y' = \frac{y}{d + d_s},$$

where d_s is 60 s of fixed compute time for DSCOPE.

Finding 17 - *Short-lived use of IP addresses maximizes economical yield of new behavior.* Optimizing this yield based on our measurement data, we find $d = 456$ s for source IPs and $d = 215$ s for ASNs, similar to the initial designed collection duration of 600 s in our measurement study. Looking at empirical yields (Figure 14), we see that maximum yields for both ASNs and IPs occur during the 10 minute collection window.

Finding 18 - *Extended measurement on a given IP is not necessary to achieve high coverage.* We can also use our model of random behavior to predict coverage for a given collection duration. Short lived collection empirically achieves acceptably high coverage of overall random scanning behavior: 50% of random scanning IPs to a given destination will be measured after just 22min (90% after 72 min), while 50% of scanning ASNs to a given destination will be seen after just 5 min (90% after 17 min). Depending on measurement goals, it may be desirable to exceed the optimal yield duration to achieve higher coverage of distinct scanner IPs, ASNs, or some other phenomenon. Alternately, one might collect for shorter than the optimal yield time to measure high-rate or non-random (e.g., periodic) phenomena across a larger number of IPs.

Limitations Our analysis of collection duration is performed under the assumption of random (Poisson) traffic. While this holds true for the vast majority of traffic seen, the lagging traffic discussed previously is one such example that does not conform. Measuring this traffic would preference a longer study duration, though observed instances of lagging traffic suggest average delays of only 100-200s, having no material impact on optimal measurement duration.

8 Discussion

DSCOPE provides a new lens for research on the Internet. We anticipate that the representative data provided by cloud-based measurement will enable researchers to answer questions about emergent trends in scanner behavior, security of cloud and other deployed systems, and traffic classification. Here, we discuss broad limitations and considerations in cloud-based measurement and provide thoughts on future applications of cloud-derived datasets.

8.1 Statistical Validity in Sampled Cloud Data

Unlike existing telescopes, which by construction achieve complete coverage of the traffic targeted at some given IP range, cloud-based Internet measurement is fundamentally a stochastic process. Further, cloud IP spaces are heterogeneous across regions, IP histories, and previously-deployed services. Because of this, each individual cloud IP can receive vastly different traffic, and the distribution of phenomena across these IPs may not obey well-defined distributions, which form the basis of many conventional statistical tests. Further, large-scale deployments and scans can lead to correlations among a subset of cloud IPs that do not hold for the population as a whole. For instance, as seen in `eu2` (Finding 11) sampling from a small number of IPs in the region may inadvertently capture trends due to IP history, rather than phenomena arising from targeting of the region itself. It is also important to carefully consider what counts as a data point for purposes of statistical significance; in our dataset we saw individual servers receiving orders of magnitude more traffic than background due to configurations left by previous tenants. When analyzing cloud IPs, tests and estimations used must be insensitive to the underlying distribution (i.e., nonparametric) and carefully consider data sample dependence.

8.2 Telescope Footprint and IPv4 Exhaustion

A key advantage of DSCOPE over conventional telescopes is that it opportunistically measures unused cloud IP addresses. In contrast, conventional telescopes require long-term control of large regions of IP address space. As IPv4 addresses become scarcer, and prices correspondingly rise [34], the capital cost of telescopes has likewise increased.

By taking a capital-cost approach, we can obtain a rough estimate for the operational cost of running a conventional telescope. In the case of Merit's ORION telescope (consisting of 475 k IP addresses), and at current market prices of 40 USD per address and a federal funds rate of 4.5%, we obtain an ongoing cost of capital of 855 kUSD/year ($0.0002/IP-Hr). While these addresses can undoubtedly serve as an investment, the recent peak in the price of IPv4 addresses [20] suggests this may not always be the case. As organizations divest unused IP address blocks [38] for use by public clouds, novel means of measuring Internet phenomena with lower address footprints–such as DSCOPE–may allow these ranges to be reclaimed for other uses.

8.3 Future Work in Internet Measurement

DSCOPE's large-scale measurement of representative scanning phenomena on cloud IPs offers a new dataset for researchers to study the Internet. From this, we anticipate three primary future research thrusts: (1) further characterization of known phenomena in the context of deployed cloud systems, e.g., measurements of how scanner behavior varies over time against cloud services, (2) analyses enabled by the interactivity of DSCOPE, such as large-scale characterization of application-layer payloads, and (3) characterization of cloud-specific phenomena, such as how traffic from other cloud services is distributed across the address space, or how cloud use of the IP address pool evolves over time. We have designed DSCOPE to be a general-purpose Internet measurement apparatus, and plan to share data products useful in these and other future research directions.

9 Conclusion

As adversarial behavior on the Internet evolves, measurement methodologies must likewise remain agile to characterize and defend against emergent threats. DSCOPE's dataset provides researchers a vantage point *in situ* with valuable public cloud targets, and through its adaptive deployment and interactivity achieves broad coverage of sophisticated adversaries. Further, DSCOPE's methodology can be implemented by anyone with a cloud account, enabling researchers to confidently reproduce results previously only obtainable by a limited set of connected parties. In these ways, DSCOPE will enable researchers to reach new insights about the security of deployed services.

Acknowledgments

The authors would like to thank John Domico at Penn State, as well as Kelly Rivera and Pam Snyder at UW-Madison, for their indispensable help with logistics relating to the deployment of DSCOPE. The authors would like to thank Ryan Sheatsley and Yohan Beugin for helpful feedback during the writing process. We also thank the anonymous reviewers and our shepherd for helpful insights throughout the reviewing process. We additionally thank Michalis Kallitsis for insights and assistance with using Merit's ORION telescope data in our analysis.

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE1255832. This material is based upon work supported by the National Science Foundation under Grant No. CNS-1900873, CNS-1703592, CNS-2039146, and CNS-2106517. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

[1] Manos Antonakakis, Tim April, Michael Bailey, Matt Bernhard, Elie Bursztein, Jaime Cochran, Zakir Durumeric, J. Alex Halderman, Luca Invernizzi, and Michalis Kallitsis. Understanding the mirai botnet. In *26th USENIX security symposium (USENIX Security 17)*, pages 1093–1110, 2017.

[2] AWS IP address ranges - AWS General Reference. https://docs.aws.amazon.com/general/latest/gr/aws-ip-ranges.html.

[3] Azure IP address ranges. https://www.microsoft.com/en-us/download/details.aspx?id=56519.

[4] Michael Bailey, Evan Cooke, Farnam Jahanian, Jose Nazario, and David Watson. The internet motion sensor-a distributed blackhole monitoring system. In *NDSS*. Citeseer, 2005.

[5] Michael Bailey, Evan Cooke, David Watson, Farnam Jahanian, and Niels Provos. A Hybrid Honeypot Architecture for Scalable Network Monitoring. page 18, October 2004.

[6] Vaibhav Bajpai and Jürgen Schönwälder. A survey on internet performance measurement platforms and related standardization efforts. *IEEE Communications Surveys & Tutorials*, 17(3):1313–1341, 2015.

[7] H. Ballani, P. Francis, and X. Zhang. A Study of Prefix Hijacking and Interception in the Internet. In *In Proceedings of ACM SIGCOMM*, 2007.

[8] Todd Bishop. Amazon Web Services posts record $13.5B in *profits* for 2020 in Andy Jassy's AWS swan song. https://www.geekwire.com/2021/amazon-web-services-posts-record-13-5b-profits-2020-andy-jassys-aws-swan-song/, February 2021.

[9] Kevin Borgolte, Tobias Fiebig, Shuang Hao, Christopher Kruegel, and Giovanni Vigna. Cloud Strife: Mitigating the Security Risks of Domain-Validated Certificates. In *Proceedings 2018 Network and Distributed System Security Symposium*, San Diego, CA, 2018. Internet Society.

[10] J. Caballero, C. Grier, C. Kreibich, and V. Paxson. Measuring Pay-per-Install: The Commoditization of Malware Distribution. In *In Proceedings of USENIX Security Symposium*, 2011.

[11] CAIDA. Sustainable Tools for Analysis and Research on Darknet Unsolicited Traffic, 2023.

[12] Ulhas J. Dixit and Parviz N. Nasiri. Estimation of parameters of a right truncated exponential distribution. *Statistical Papers*, 49(2):225–236, December 2007.

[13] Zakir Durumeric, Eric Wustrow, and J Alex Halderman. Zmap: Fast internet-wide scanning and its security applications. In *USENIX Security Symposium*, volume 8, pages 47–53, 2013.

[14] DomainTools/Farsight DNSSDB, 2023.

[15] GCP IP address ranges. https://www.gstatic.com/ipranges/cloud.json.

[16] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, September 2020.

[17] Raphael Hiesgen, Marcin Nawrocki, Alistair King, Alberto Dainotti, Thomas C. Schmidt, and Matthias Wählisch. Spoki: Unveiling a New Wave of Scanners through a Reactive Network Telescope. In *31st USENIX Security Symposium (USENIX Security 22)*, pages 431–448, Boston, MA, August 2022. USENIX Association.

[18] S. Hills. Considerations and recommendations concerning internet research and human subjects research regulations, with revisions. *HHS. gov*, 2013.

[19] J. D. Hunter. Matplotlib: A 2d graphics environment. *Computing in Science & Engineering*, 9(3):90–95, 2007.

[20] Geoff Huston. IP addressing through 2022. https://blog.apnic.net/2023/01/23/ip-addressing-through-2022/, January 2023.

[21] Xuyang Jing, Zheng Yan, and Witold Pedrycz. Security data collection and data analytics in the internet: A survey. *IEEE Communications Surveys & Tutorials*, 21(1):586–618, 2018.

[22] Jeremy Kepner, Michael Jones, Daniel Andersen, Aydin Buluc, Chansup Byun, K Claffy, Timothy Davis, William Arcand, Jonathan Bernays, David Bestor, William Bergeron, Vijay Gadepally, Daniel Grant, Micheal Houle, Matthew Hubbell, Hayden Jananthan, Anna Klein, Chad Meiners, Lauren Milechin, Andrew Morris, Julie Mullen, Sandeep Pisharody, Andrew Prout, Albert Reuther, Antonio Rosa, Siddharth Samsi, Doug Stetson, Charles Yee, and Peter Michaleas. Temporal Correlation of Internet Observatories and Outposts. In *2022 IEEE International Parallel and Distributed Processing Symposium Workshops (IPDPSW)*, pages 247–254, May 2022.

[23] C Kreibich and J Crowcroft. Automated nids signature generation using honeypots. *Proceedings of the Special Interest Group on Data Communication (SIGCOMM'03)*, 2003.

[24] MaxMind - GeoLite2 Free Geolocation Data.

[25] The Merit Network, Inc. ORION - October 11-17, 2022. https://www.merit.edu.

[26] David Moore, Colleen Shannon, and K Claffy. Code-red: a case study on the spread and victims of an internet worm. In *Proceedings of the 2nd ACM SIGCOMM Workshop on Internet measurment*, pages 273–284, 2002.

[27] David Moore, Colleen Shannon, Geoffrey M Voelker, and Stefan Savage. Network Telescopes: Technical Report. page 14.

[28] David Moore, Geoffrey M Voelker, and Stefan Savage. Inferring Internet Denial-of-Service Activity.

[29] The pandas development team. pandas-dev/pandas: Pandas, February 2020.

[30] Ruoming Pang, Vinod Yegneswaran, Paul Barford, Vern Paxson, and Larry Peterson. Characteristics of internet background radiation. In *Proceedings of the 4th ACM SIGCOMM conference on Internet measurement - IMC '04*, page 27, Taormina, Sicily, Italy, 2004. ACM Press.

[31] Eric Pauley and Patrick McDaniel. Understanding the Ethical Frameworks of Internet Measurement Studies. In *The 2nd International Workshop on Ethics in Computer Security (EthiCS 2023)*, San Diego, CA, February 2023.

[32] Eric Pauley, Ryan Sheatsley, Blaine Hoak, Quinn Burke, Yohan Beugin, and Patrick McDaniel. Measuring and Mitigating the Risk of IP Reuse on Public Clouds. In *2022 IEEE Symposium on Security and Privacy (SP)*, pages 558–575, May 2022. arXiv:2204.05122 [cs].

[33] Egon Sharpe Pearson and Herman Otto Hartley. *Biometrika Tables for Statisticians*, volume 2. Biometrika Trust, 1st edition, 1976.

[34] Lars Prehn, Franziska Lichtblau, and Anja Feldmann. When wells run dry: the 2020 IPv4 address market. In *Proceedings of the 16th International Conference on emerging Networking EXperiments and Technologies*, pages 46–54, Barcelona Spain, November 2020. ACM.

[35] Niels Provos et al. A virtual honeypot framework. In *USENIX Security Symposium*, volume 173, pages 1–14, 2004.

[36] Philipp Richter and Arthur Berger. Scanning the Scanners: Sensing the Internet from a Massively Distributed Network Telescope. In *Proceedings of the Internet Measurement Conference*, IMC '19, pages 144–157, New York, NY, USA, October 2019. Association for Computing Machinery.

[37] Amazon Route 53 - Amazon Web Services. https://aws.amazon.com/route53/.

[38] Aftab Siddiqui. MIT Goes on IPv4 Selling Spree. https://www.internetsociety.org/blog/2017/05/mit-goes-on-ipv4-selling-spree/, May 2017.

[39] Raffaele Sommese, KC Claffy, Roland van Rijswijk-Deij, Arnab Chattopadhyay, Alberto Dainotti, Anna Sperotto, and Mattijs Jonker. Investigating the impact of ddos attacks on dns infrastructure. In *Proceedings of the 22nd ACM Internet Measurement Conference*, pages 51–64, 2022.

[40] Johannes Ullrich. DShield - SANS.edu Internet Storm Center.

[41] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C J Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, Paul van Mulbregt, and SciPy 1.0 Contributors. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nature Methods*, 17:261–272, 2020.

[42] Michael L. Waskom. seaborn: statistical data visualization. *Journal of Open Source Software*, 6(60):3021, 2021.

[43] V. Yegneswaran, P. Barford, and J.Y. Cai. Camoflauging Honeynets. In *In Proceedings of IEEE Global Internet*, 2007.

[44] V. Yegneswaran, P. Barford, and J. Ullrich. Internet Intrusions: Global Characteristics and Prevalence. In *In Proceedings of ACM SIGMETRICS*, 2003.

[45] V. Yegneswaran, J. Giffin, P. Barford, and S. Jha. An Architecture for Generating Semantic-aware Signatures. In *In Proceedings of USENIX Security Symposium*, 2011.

[46] Vinod Yegneswaran, Paul Barford, and Vern Paxson. Using honeynets for internet situational awareness. In *Proceedings of the Fourth Workshop on Hot Topics in Networks (HotNets IV)*, pages 17–22. Citeseer, 2005.

[47] Vinod Yegneswaran, Paul Barford, and Dave Plonka. On the design and use of internet sinks for network abuse monitoring. In *International Workshop on Recent Advances in Intrusion Detection*, pages 146–165. Springer, 2004.