A Brave New World: Studies on the Deployment and Security of the Emerging IPv6 Internet

by

Jakub Jerzy Czyz

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
(Computer Science and Engineering)
in the University of Michigan
2016

Doctoral Committee:
Associate Professor Michael D. Bailey, University of Illinois, Co-Chair
Associate Professor J. Alex Halderman, Co-Chair
Provost Farnam Jahanian, Carnegie Mellon University
Professor Brian D. Noble
Professor Douglas E. Van Houweling

This thesis is dedicated to my wife, Ewa.

ACKNOWLEDGEMENTS

I would first like to thank my advisors Michael Bailey and Farnam Jahanian. They have been patiently supportive during my five years in their Michigan research group and during the prior several years that we collaborated. I have learned a great deal from you two—lessons both academic and personal, and I am inspired and privileged to watch your careers continuing to flourish. Thanks for taking a chance on me!

Second, I sincerely thank my committee co-chair Alex Halderman, as well as members Brian Noble and Douglas Van Houweling. I admire your work and am humbled to be associated with each of you. Although we have only collaborated briefly, your collective time and insight have been valuable and are appreciated. While not a committee member, I also learned a great deal during the year I spent with Prof. John Laird and his Soar AI research group. In spite of my change of course in the program, John has always been supportive, kind, and friendly, for which I am deeply grateful.

I am indebted to my collaborators, mentors, teachers, and sources of encouragement, especially Mark Allman, Michalis Kallitsis, and Manish Karir. While my name is on the front of this thesis, like most dissertations, the work described here was the product of efforts by many people, without whose hard work it would not exist. Mark's deep knowledge of networks, his ability to dissect and illuminate, as well as his quick and dependable communication style have taught me a great deal and made him a pleasure to work with. Michalis has always been patient, encouraging, and a generous collaborator, and I am grateful to know him. Manish has been a supporter and friend since we first worked together in 2009, and his equanimity, quiet persistence, and knack for seeing the bigger picture have given me faith and encouraged me to persevere.

It would be remiss of me to not thank my office mates, lab mates, friends, peers, and

lab predecessors at Michigan, who have been a truly fun, encouraging, and inspiring bunch of geeks that I am grateful and privileged to have met. You include, at least, Timur Alperovich, Mona Attariyan, Denis Bueno, Ari Chivukula, Mike Chow, Nate Derbinsky, David Devecsery, Zakir Durumeric, Denis Foo Kune, Sai Gouravajhala, Olga Kornievskaia, Alex Kuhn, Kyle Lady, Jason Lee, Jake Maes, Jon Oberheide, Jess Ouyang, Kee Shen Quah, Alex Robinson, Benjamin Wester, Eric Wustrow, Yunjing Xu, and Jing Zhang. The doctoral road is a long one and full of setbacks and pitfalls, but when our paths crossed, the blood, sweat, and tears we shed together helped make the journey bearable. Your alacrity, tenacity, humor, and career accomplishments have both humbled and encouraged me tremendously during many moments of doubt. I am confident you all will go out there and crush it; indeed, many of you already are!

To my family members, especially my mom, daughter, and father, I thank you for bringing me faith, joy, and strength. Mom, your love of knowledge and steadfast belief in me have led me here. When I was around ten years old, you were working two jobs and we were living hand to mouth, but you bought a full set of Encyclopedia Britannica for more than our car was worth. This anecdote summarizes why I persevered in school all the way to the milestone that this dissertation represents.

Lastly, were it not for the stalwart support, encouragement, empathy, and wise advice of my wife, Ewa Czyz, this dissertation would not have been possible. Having her accompanying me on the academic odyssey, with the different perspective and sagacious insight of her psychological training, has saved my sanity and kept me going countless times when I doubted myself. Words can not describe my gratitude and awe at your patience, encouragement, kindness, support, and companionship on this long journey.

TABLE OF CONTENTS

CHAPTER

LIST OF TABLES

Table

LIST OF FIGURES

Figure

ABSTRACT

Recent IPv4 address exhaustion events are ushering in a new era of rapid transition to the next generation Internet protocol—IPv6. Via Internet-scale experiments and data analysis, this dissertation characterizes the adoption and security of the emerging IPv6 network. The work includes three studies, each the largest of its kind, examining various facets of the new network protocol's deployment, routing maturity, and security.

The first study provides an analysis of ten years of IPv6 deployment data, including quantifying twelve metrics across ten global-scale datasets, and affording a holistic understanding of the state and recent progress of the IPv6 transition. Based on cross-dataset analysis of relative global adoption rates and across features of the protocol, we find evidence of a marked shift in the pace and nature of adoption in recent years and observe that higher-level metrics of adoption lag lower-level metrics.

Next, a network telescope study covering the IPv6 address space of the majority of allocated networks provides insight into the early state of IPv6 routing. Our analyses suggest that routing of average IPv6 prefixes is less stable than that of IPv4. This instability is responsible for the majority of the captured misdirected IPv6 traffic. Observed dark (unallocated destination) IPv6 traffic shows substantial differences from the unwanted traffic seen in IPv4—in both character and scale.

Finally, a third study examines the state of IPv6 network security policy. We tested a sample of 25 thousand routers and 520 thousand servers against sets of TCP and UDP ports commonly targeted by attackers. We found systemic discrepancies between intended security policy—as codified in IPv4—and deployed IPv6 policy. Such lapses in ensuring that the IPv6 network is properly managed and secured are leaving thousands of important devices more vulnerable to attack than before IPv6 was enabled.

Taken together, findings from our three studies suggest that IPv6 has reached a level and pace of adoption, and shows patterns of use, that indicates serious production employment of the protocol on a broad scale. However, weaker IPv6 routing and security are evident, and these are leaving early dual-stack networks less robust than the IPv4 networks they augment.

Thesis Statement:

The dual-stacked, phased adoption of IPv6 has weakened Internet availability and security.

CHAPTER 1

Introduction

The Internet is undergoing a large-scale transformation, arguably the biggest technological change the network has faced in the 30 years since the NSFNET was launched with ARPANET's open Internet Protocol suite [118]. After three decades of service, an eternity in computing technology years, that Internet protocol, the core and common thread that has held the network together, is being profoundly changed for the first time.

The Internet has been called the largest and most complex system built by man, and the value and importance of the network to modern life is without question [58, 144]. Thus, the study of fundamental changes to the network is obviously of considerable scientific, engineering, policy, and commercial value. Gleaning insight that can illuminate and facilitate this complex system's evolution—and, potentially, inform future system-wide changes—is the chief goal of this thesis.

The current version of the Internet protocol (version four) was first standardized in 1980 [143] and adopted by what became the Internet when the NSFNET launched in 1986 [172]. The protocol has been wildly successful, sustaining the network for longer than its creators imagined [98, 130, 148]. It forms the lingua franca in the middle of the network protocol stack that enables rich and varied innovations at both the physical medium-specific layers below it and at the application layers above. But, its sunset is near.

On September 24, 2015 the American Registry for Internet Numbers (ARIN) assigned its final Internet Protocol version four (IPv4) address [42]. This event is the most recent in a chain of IPv4 address exhaustion events during the last several years that are helping usher in a new era of rapid transition to the next generation Internet protocol—IP version six (IPv6). Indeed, while proving causation is difficult, this dissertation presents some evidence of correlation. One of the contributions of this thesis is a large-scale study on the progression of IPv6 adoption over ten years. The results of the study we describe in the next chapter indicate that a *qualitative* shift in the nature and a *quantitative* change in the trajectory of adoption has coincided with the onset of this period of acute IPv4 address exhaustion.

Aside from the well-documented progressive depletion of IPv4 address space (e.g., [107, 148]), pressure to adopt IPv6 has also come on several other fronts. In the last decade, the desire for ubiquitous connectivity coupled with advances in the miniaturization and power efficiency of electronics has spurred the mobile computing revolution [121]. The majority of new Internet users and devices are mobile, be that via smartphones, tablets, or otherwise [36]. This mobile trend is especially strong in emerging markets, where many new Internet users are skipping fixed-line connectivity [36]. New mobile networks, needed to fuel this growth, favor IPv6 for numerous reasons, including the larger per-network address space and greater routing flexibility, simpler architecture, and long-term investment [147].

Meanwhile, the search for new revenue by older industries along with low-cost, fast, and pervasive network connectivity has spawned the Internet of things (IoT) phenomenon, where previously stand-alone and usually unintelligent devices, such as refrigerators, thermostats, deadbolts, and even cars, are increasingly being augmented with embedded computing systems and constantly connected to the global Internet to provide new features [39, 104, 151]. These two powerful trends, mobility and IoT, have caused both an increase in the demand for Internet address space [93] as well as a desire for some of the features that IPv6 itself provides, most notably better mobility support, stateless address auto-configuration, as well as enabling end-to-end connectivity without the network address translation (NAT) common with IPv4—a wart on the network that can increase latency, complicate architectures, and break some applications [51, 182]. As such, IPv6 is a major enabler of the IoT [64, 104, 151, 182].

In this thesis, through three large empirical studies, we broadly examine the global state of adoption of the new Internet protocol—including the network's capability to support it as well as how the protocol is actually being used—and we explore in more depth how two important properties of the network, availability and security, are served by IPv6 in these early years of its deployment. The transition to IPv6 is not a backwards-compatible upgrade; due to this and other factors, the rollout has been slow, rocky, and disruptive [27, 97]. The research in this thesis aims to provide a holistic accounting of the state of the IPv6 network after recent events that give stronger impetus for its adoption. While we focus on carefully measuring the penetration and quality of the emerging network, as a result of these measurements and experiments we also seek to extract broader lessons about such large-scale Internet upgrades—lessons that can be applied to future transitions, beyond IPv6.

1.1 IPv6 Background

Before we begin, some background on IPv6 is in order. As the commercial Internet began to grow exponentially in the early 1990's, it became clear that the addressing structure of IPv4 would not provide enough space to accommodate demand for long (see e.g., [68]). As early as late 1990, efforts within the Internet engineering and standards community began to focus on developing a successor to the existing protocol (IPv4) [24]. Initially termed IPng (for "next generation"), the new protocol was eventually standardized as IP version six (IPv6), via Request for Comments (RFC) 1883, in December 1995 [53].[1] Three years later, in December 1998, refinements lead to RFC 2460, which is the official standard defining the IPv6 protocol [54].

1.1.1 A Slow Start

During the nearly two decades since IPv6 standardization, the global pace of adoption, as technology goes, has been very slow [27]. There is cost associated with the upgrade, but the incentives to deploy the new protocol waned after the advent of several provisional measures to extend the life of IPv4 [27]. Notably, dynamic host addressing—e.g., using

[1] Version number five was not used due to a header conflict with ST, an experimental protocol that was never widely adopted [55].

the dynamic host configuration protocol (DHCP) [57]—allowed an address to be shared temporally, while network address translation (NAT) allowed spatial sharing of a single address among multiple, even thousands, of hosts [162]. The address space use was also made more efficient, slowing depletion, via classless interdomain routing (CIDR) [69]. Like thanks in part to these stopgap measures being so effective at postponing exhaustion, the warnings of looming exhaustion of IPv4 addresses, which had been purported as imminent for years, repeatedly failed to materialize and became seen by some as akin to crying wolf (see e.g., [63,67,80]). Some of the perceived problems with IPv4 were addressed by updates on top of the existing protocol, as was the case for protocol-layer security, which was addressed with IP security extensions (IPsec) added to IPv4 and then removed as a required component for IPv6 nodes [81]. This likely further reduced incentive to move to IPv6. Lastly, as the Internet exploded and the number of IPv4 devices ballooned, the inertia of the existing broadly-deployed but older standard naturally grew to the detriment of IPv6 [64, 84]. Deployment of the new protocol also has significant associated costs, with the average enterprise needing to allocate 7% of its IT budget for the transition, according to research firm Gartner [84]. For many network operators, IPv6 deployment would mean supporting both protocols (i.e., a *dual-stack* configuration) and retrofitting the support for IPv6 onto a vast and growing population of existing devices—a significant cost with little perceived upside or urgency [29, 95]. This is especially true for incumbent operators with ample supply of IPv4 address space [64].

As the value of an internetwork protocol lies in connecting networks to each other, there is a natural "chicken and egg" dilemma that comes into play. For the new protocol to be useful to a given network or Internet stakeholder, there had to be other networks to which that network could connect [100]. Thus, until enough early adopters (by choice or necessity) deployed the new protocol, there was little incentive for anyone else to bear the expense to be one of the first to do so [27].

Starting in 2011, however, our work shows that while IPv4 address exhaustion has become more severe, the number of adopters has increased. This, along with the mobile network IPv6 push [84], the advent of IoT, and the other factors outlined above, has apparently driven adoption rates to increase markedly.

1.1.2 What the Upgrade Means

The main changes that IPv6 introduced have to do with addressing and the packet header format. The address fields in the header that represent the source and destination of packets have been extended from 32 bits in IPv4 to 128 bits for each in IPv6 [54]. The IPv4 address space theoretically supports up to 4.2 billion unique addresses (though, due to some reserved blocks, about 3.7 billion is usable [167]).[2] The IPv6 theoretical address space, on the other hand, (2^{128}, or 3.4×10^{38} addresses—340 trillion trillion) represents an estimated thousands of addresses for every atom on the surface of the earth [28]. Thus, it is assumed to suffice for unique addressing of each network-connected device, without sharing, for the foreseeable future. In fact, the space was purposefully made large and effectively sparse in order to allow hierarchical grouping of allocated addresses for efficient routing [24]. For instance, current standards and practices dictate assigning a 64-bit long address block to each subnet, which translates to 1.8×10^{19} possible addresses on each small local network, versus just 254 usable addresses on a typical, 8-bit, subnet in IPv4 [136]).

In addition to giving networks room to grow, one other consequence of the vastly larger address space—especially when combined with the idea of privacy addresses, where a host changes what address it uses periodically, as well as the advent of link-local addresses—is a difficulty keeping track of which machine is associated with which address [32]. Gone are the days of simple mappings of a single address to a single network interface. Beyond privacy and link local addresses (valid for communication on the local LAN only), IPv6 also includes: a more complex and feature-rich replacement for ARP called Neighbor Discovery Protocol (NDP); changes in fragmentation (now only done at end nodes); and, much greater use of multicast (IPv6 does not allow broadcasts) [54, 89, 137]. While each of these changes has an impact on the security and management of IPv6 networks, in this thesis, when we examine security and management, we mainly focus on core aspects of IPv6 operation and deployment that are common with IPv4.

In addition to the addressing and feature changes discussed already, IPv6 includes a new packet header that was intended to be more efficient and simpler than IPv4 [24]. We show

[2] Incidentally, there are already an estimated 15 billion Internet-connected devices in 2015, suggesting widespread use of address sharing [36].

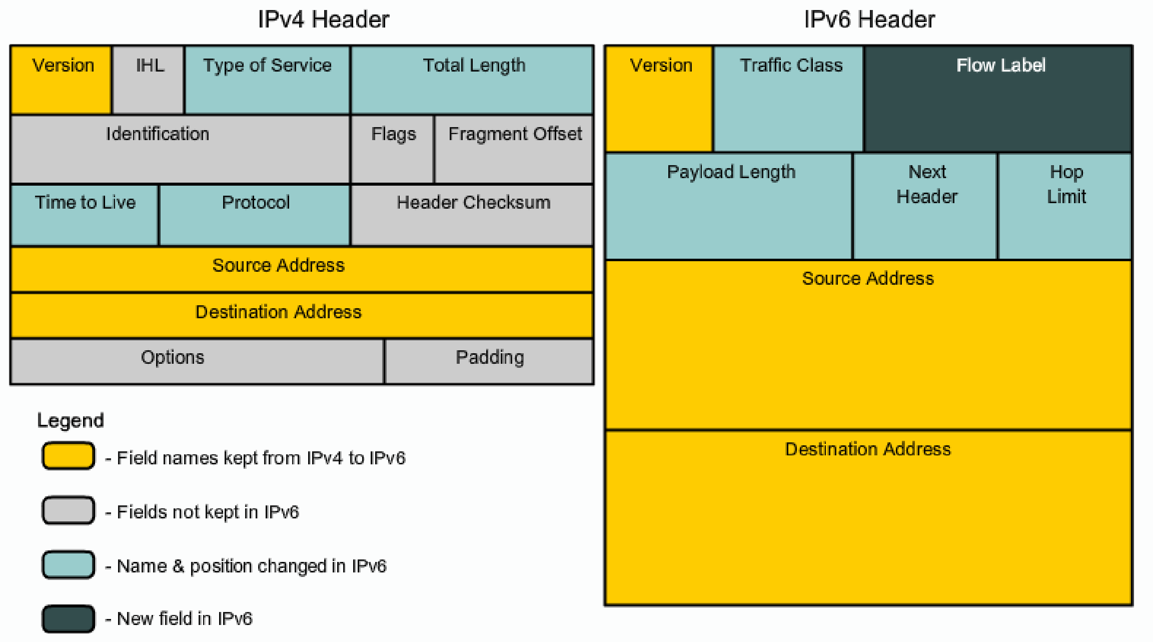


Figure 1.1: IPv4 and IPv6 headers compared. (Image source: [3]) The IPv6 header has longer addresses but includes fewer less-used fields. Importantly, it eschews a fixed header format in favor of a chain of one or more "Next Headers," which introduces parsing and state complexity.

both headers in Figure 1.1. Most importantly, rather than having a fixed length (of which many fields were rarely used in practice in IPv4) the new format allows for a chain of optional extension headers, which will typically culminate in an upper-layer header—such as for a user datagram protocol (UDP) or transmission control protocol (TCP) segment. According to the original IPv6 standards (i.e., RFC 2460), these extension headers could appear in nearly any order, including in ways where the first fragment of a fragmented packet does not contain the layer four header type (e.g., TCP) [54]. As a consequence of these header changes, parsing of IPv6 headers is substantially more complex than in IPv4 and this has lead to numerous security problems, including ones related to fragments, which we touch on later [74]. For now, the main point is that the header changes have not proven simple to implement nor without cost (e.g., [109]). On top of the hardware and software changes needed for network devices, end hosts, and management tools to support the larger and more dynamic address space, these header changes have contributed to friction for adoption and to brittle early implementations (see e.g., [5, 76]).

As we alluded to above and elaborate on more in the following chapter, it appears, however, that the practical exhaustion of IPv4 for new networks, starting around a February 2011 milestone, and likely helped by other factors, such as the mobility and IoT trends, have finally tipped the scale in favor of IPv6 deployment by a rapidly growing set of global actors [93]. In fact, as we detail in § 2.2, in 2013, for the first time, the majority of new networks appear to be allocated with IPv6 as well as IPv4. In this thesis, we set out to empirically measure the overall pace and nature of this adoption as well as hone in on understanding several key aspects—namely, routing and network security policy—of the emerging IPv6 network as it finally gains traction.

1.2 Contributions

The work in this thesis includes the following key contributions:

- **A broad and longitudinal measurement of IPv6 adoption.** We conducted a long and broad IPv6 adoption measurement study, exploring twelve metrics of adoption across a compendium of ten data sets, several spanning ten years of measurements. While some of the data we used were publicly available, we contributed several new global-scale measurements, including the largest IPv6 Internet traffic sample reported, representing around a third of Interdomain traffic. Further, by analyzing and comparing the public data with our own on the same axes, we are able to construct an updated multi-perspective view on recent adoption trends.

- **A systemic understanding of the state of adoption.** Aside from being broad and longitudinal, our IPv6 adoption measurements afford a *systemic* view of IPv6 adoption. By examining the adoption process from multiple perspectives *and* at multiple layers of the system, we are able to observe properties of the protocol's deployment that are only evident when simultaneously examining multiple aspects of the network. For instance, we learn that the perspective one takes when measuring IPv6 can yield orders of magnitude different estimates of adoption level than others. Interestingly, we find that layers of the system are being adopted in approximate sequence, with higher-level functions and properties coming online subsequently to lower-level functions. Via this holistic view of the system, we also see a pattern in recent adoption *rates* and the *character* of the emerging IPv6 network. Most notably, we find a dramatic increase in the pace of adoption, starting in 2011, across several metrics. Aside from the increasing pace of adoption, our measurements also show, across multiple datasets, a recent qualitative shift in the nature of IPv6 use, together heralding an apparent maturing of the new protocol into a first-class mission-critical component of the Internet.

- **An exploration of the covering prefix methodology in network telescopes.** Our IPv6 background radiation chapter describes the first academic study and the largest known use of a covering prefix methodology for capturing Internet background radiation in IPv6. This technique affords capturing both a quantitatively and qualitatively different traffic sample and allows us to study aspects of the IPv6 network that could not be studied with the traditional unallocated-space prefix approach. We are able to capture significantly more packets and able to study more categories of misconfiguration than is possible with the traditional approach. This new perspective on background radiation allows us to: (*i*) measure the routing stability of the new protocol; (*ii*) capture and study more typical IPv6 traffic; (*iii*) make high-confidence existence claims about classes of dark IPv6 traffic due to monitoring the majority of non-transition unused IPv6 address space; and, (*iv*) detect misconfiguration types not otherwise visible.

- **A broad and deep measurement of IPv6 Internet Background Radiation.** Using this covering prefix methodology, we conducted the largest IPv6 network telescope study to date, capturing data from prefixes that subsumed 86% of non-transition allocated IPv6 address space and including all of the unallocated space under the main prefixes from which each of the five global registries now allocate addresses. This unique perspective allowed us to detect differences among global regions, to closely examine IPv6 routing instability at scale, and to draw conclusions about the early state of global IPv6 network scanning and worm traffic. We found that IPv6 routing is not as mature as IPv4 routing in terms of stability, and the detailed view of instability in our IPv6 experiment afforded an analysis of the networks and traffic types most affected by this instability. Happily, on the security front, our broad perspective suggests that, at least during the three-month window of our larger dataset, there is no evidence of broad network scanning, a phenomenon commonplace in IPv4 [59]; nor is brute-force network worm propagation evident.

- **A measurement of global IPv6 network security policy.** We conducted the first at-scale study comparing deployed IPv6 network security policy with that of IPv4, probing a large globally-distributed sample of 25K Internet-transit routers and 520K servers across sets of common and often-attacked application protocol ports. Our findings show substantial vulnerability among the IPv6-enabled router and host population, including, most egregiously, on dual-stacked routers, which were found to allow TCP connections to Telnet, SSH, and BGP ports, for example, at rates markedly higher than over IPv4. Our study included a deep analysis of where these policy discrepancies occur, their consistency within networks, and the susceptibility of these hosts to attack by informed brute-force scanning techniques in IPv6. Contacts with a dozen network operators confirmed our findings and lead to immediate remediation on a majority of the contacted networks.

Thesis Statement:
The dual-stacked, phased adoption of IPv6 has weakened Internet availability and security.

1.3 Structure

The remainder of this thesis is structured as follows. We begin with a study of the level of IPv6 adoption, including core capability and the nature and volume of usage, in Chapter 2. Next, in Chapter 3 we report on a study of IPv6 Internet background radiation, which includes analysis and measurement of the state of IPv6 routing. Following that, in Chapter

4 we examine the broad state of IPv6 network security, wherein we show how port security policy is being deployed with the new protocol. In Chapter 5 we discuss related work, and, finally, in Chapter 6 we conclude the thesis with a review of results and contributions, a discussion of high-level takeaways, notes on recent adoption progress, and some proposed future work.

CHAPTER 2

Measuring IPv6 Adoption

After years of stalled adoption, the arrival of IPv6 is finally upon us. We start this thesis with a chapter in which we take a high-level view of adoption, examining twelve adoption metrics taken against ten global-scale longitudinal datasets, several spanning a decade of data. Our goal is to provide a *systemic* snapshot of the recent progress and current state of adoption, to compare rates and patterns seen across different types of data, and to glean insights that are only visible when taking such a broad perspective.

Four of the datasets we analyze are original ones that we contribute, including a global Internet traffic dataset that includes traffic statistics from 260 providers and represents 16,200 petabytes/month (a daily median of 58 Tbps), or approximately 33-50% of all Internet traffic [88]—the largest traffic sample reported in an IPv6 study. In addition to the traffic data, we add DNS query data from several of the largest globally-distributed IPv4-based replicas of the .com and .net top-level domains (TLDs), as well as nearly all native IPv6 replicas for these TLDs. In addition to the four new datasets we contribute, we include new analysis of six datasets that are publicly-accessible, many of which have been described in previous studies. The goal is to provide an updated view across many vantage points, as well as to compare previously-published data on the same axes with each other and with the newer traffic, naming, and content-readiness data we contribute.

Our analysis leads to several key cross-dataset findings. First, we find that over the last three years, IPv6 has reached a significant developmental milestone and is finally being used natively and for normal, production traffic, on a non-trivial and accelerating scale.

While IPv6 is still under 1% of Internet traffic, IPv6 traffic has increased over 400% in each of the last two years in our study, it relies much less on transition technologies, and it is used for similar applications to IPv4, with similar performance. In other words, both its nature of use and rate of adoption have shifted dramatically starting in 2011. Second, we find that IPv6 adoption level differs by up to *three orders of magnitude* depending on the metric examined. For instance, although IPv6 monthly address allocations are about 57% of IPv4, the percentage of traffic that is IPv6 is about 0.6% that of IPv4. Finally, we find that not only are global regions adopting IPv6 at different rates, but the relative level of adoption in each region also varies by metric. This suggests that perhaps policy incentives and barriers to adopt the new protocol do not just vary globally but also across layers of the Internet ecosystem within each region (e.g., address allocation policies constraining address assignment in spite of high IPv6 traffic in a region).

In the following sections, we first introduce a taxonomy of metrics we seek to measure, along with an overview of the datasets we bring to bear for these measurements. The sections that follow that taxonomy each describe metrics falling under one of the following categories: addressing (§ 2.2), naming (§ 2.3), routing (§ 2.4), end-to-end reachability (§ 2.5), usage profile (§ 2.6), and, finally, performance (§ 2.7). After that, in section 2.8 we draw cross-metric conclusions about the current state of IPv6, including regional differences as well as make predictions about its future. Finally, in section 2.9 we summarize the chapter.

2.1 Our Approach

Since our aim is for a systemic picture of adoption, we must first decide what aspects to study. We start by thinking about the Internet Protocol from the perspective of the three major types of Internet stakeholders: content providers, service providers, and content consumers. Although there are notable entities that straddle or defy these labels (e.g. vendors and policy makers), these three categories encapsulate the key perspectives we believe should be considered to realistically assess deployment. We next divide the key aspects of IP itself into two classes. The first is the prerequisite functions that IP performs and that must be in place for nodes to communicate, including addressing, naming, routing, and end-to-end reachability. The second class is operational characteristics that are only evident once the network begins forwarding packets, these include transition technology use, traffic volume, application mix, and performance.

In Table 2.1 we propose one or more *metrics* that characterize the adoption of IPv6 from the key viewpoints sketched above. Some of these metrics cover more than one branch in the taxonomy. We admit that our use of the term “metric” is somewhat loose. Our aim is to point to many aspects of adoption that should be measured, but whose granularity and specificity varies. Thus, each metric could itself be thought of more as a category or issue for which specific measurements should be obtained. In this study, we present one or several such measurements for each metric that we defined.

While we believe we have identified a sufficiently comprehensive set of metrics to provide a broad picture of adoption, we do not claim completeness. There are countless possible metrics that can tell a coherent and insightful story of the adoption process. Further, while a metric such as performance naturally breaks down into sub-metrics for assessing delay, loss, jitter, reordering, throughput, etc., the specific facets of IPv6 operation that are important in any given context are likely to vary by application. As such, we do not mean to discourage further assessment along different axes or granularities than we take in this study. Rather, our goal is to set a course for developing a *high-level* and *systemic* understanding of the recent IPv6 adoption process. To this end, we bring to bear several large original datasets, as well as several public or previously published results, and use them to report measurements that align with our taxonomy. Table 2.2 summarizes the datasets we analyzed, separated into the public ones we reproduce and update, and the unique ones we contribute. We next discuss analyses of these data in detail, showing how adoption level differs and varies as we move from left to right in Table 2.1 and over time.

Table 2.1: IPv6 adoption metric taxonomy, showing the main three perspectives that Internet stakeholders occupy, the prerequisites for IPv6 to be used, as well as the operational characteristics of the protocol, once deployed.

		Prerequisite IP Functions				Operational Characteristics	
		Addressing	**Naming**	**Routing**	**End-to-End Reachability**	**Usage Profile**	**Performance**
Perspective	**Content Provider**		N1: Nameservers; R1: Server Readiness		R1: Server Readiness	U3: Transition Technologies	
	Service Provider	A1: Address Allocation; A2: Address Advertisement	N2: Resolvers	A2: Address Advertisement; T1: Topology		U1: Traffic Volume; U3: Transition Technologies	P1: Network RTT
	Content Consumer		N3: Queries		R2: Client Readiness	U2: Application Mix; N3: Queries	

Table 2.2: Dataset summary showing the time period, scale, and public or new status of the datasets we analyzed.

Dataset	Metrics	Time Period	Recent Scale	Public?
RIR Address Allocations	A1	Jan 2004 – Jan 2014	≈18K allocation snapshots (5 daily)	Yes
Routing: Route Views	A2, T1	Jan 2004 – Jan 2014	45,271 BGP table snapshots	
Routing: RIPE	A2, T1	Jan 2004 – Jan 2014		
Google IPv6 Client Adoption	R2, U3	Sep 2008 – Dec 2013	millions of daily global samples	
Verisign TLD Zone Files	N1	Apr 2007 – Jan 2014	daily snapshots of ≈2.5 million A+AAAA glue records (.com & .net)	
CAIDA Ark Performance Data	P1	Dec 2008 – Dec 2013	≈10 million IPs probed daily	
Arbor Networks ISP Traffic Data	U1, U2, U3	Mar 2010 – Dec 2013	≈33-50% of global Internet traffic; 2013 daily median: 50 terabits/sec (avg.)	No
Verisign TLD Packets: IPv4	N2, N3	Jun 2011 – Dec 2013	4 global sites, 5 of 13 gTLD NS letters (.com/.net), ≈4.5Bn queries	
Verisign TLD Packets: IPv6	N2, N3	Jun 2011 – Dec 2013	15 global sites, both gTLD NS letters (.com/.net) w/IPv6, 647M queries	
Alexa Top Host Probing	R1	Apr 2011 – Dec 2013	10,000 servers probed twice/month	

2.2 Addressing

We first examine the initial steps for IPv6 deployment: address allocation and network advertisement.

A1: Address Allocation

Before wide-scale IPv6 communication is possible, IPv6 addresses must be broadly available. Therefore, our first assessment is of the status of IPv6 address allocation. The allocation hierarchy begins with the Internet Assigned Numbers Authority (IANA) allocating address blocks to the five regional Internet registries (RIRs). In turn, the RIRs make allocations to various national and local registries and ISPs. Each RIR publishes a daily snapshot of the blocks of IP (v4 and v6) addresses (i.e., the number of prefixes) allocated to entities below it in the hierarchy. We have captured ten years of these snapshots, starting in January 2004. As a minor caveat, note that the size of a typical IPv6 prefix (2^{96}) is much larger than that of an IPv4 prefix (2^{10}), thus, prefix-based comparisons should be made with caution. However, address allocations typically correspond to network deployments, no matter the protocol; so, relative allocations do shed light on protocol deployment.

Figure 2.1 shows the aggregate number of prefixes allocated each month across all RIRs. There were less than 30 IPv6 prefixes allocated per month prior to 2007, generally increasing thereafter. In the past several years, we typically find more than 300 prefixes allocated per month, with a high point of 470 prefix allocations in February 2011. By January 2004 there had been 650 IPv6 prefix allocations, while at the end of December 2013 we observe 17,896 total prefix allocations—an increase of 27-fold. Finally, we note that at the end of our dataset the allocated IPv6 prefixes cover 2^{113} (i.e., 1.1×10^{34}) addresses.

To put the IPv6 allocation data in context, Figure 2.1 also shows IPv4 prefix allocations over the same period. The number of IPv4 prefix allocations grows from roughly 300 per month at the beginning of our observation period to a peak of 800–1000 per month at the start of 2011, after which it drops to around 500 per month in the last year, as the number of available addresses at RIRs has dwindled. [1] Overall, we find nearly 69K IPv4 prefix

[1] We elide the April 2011 point such that the remainder of the plot is more readable. During that month, we

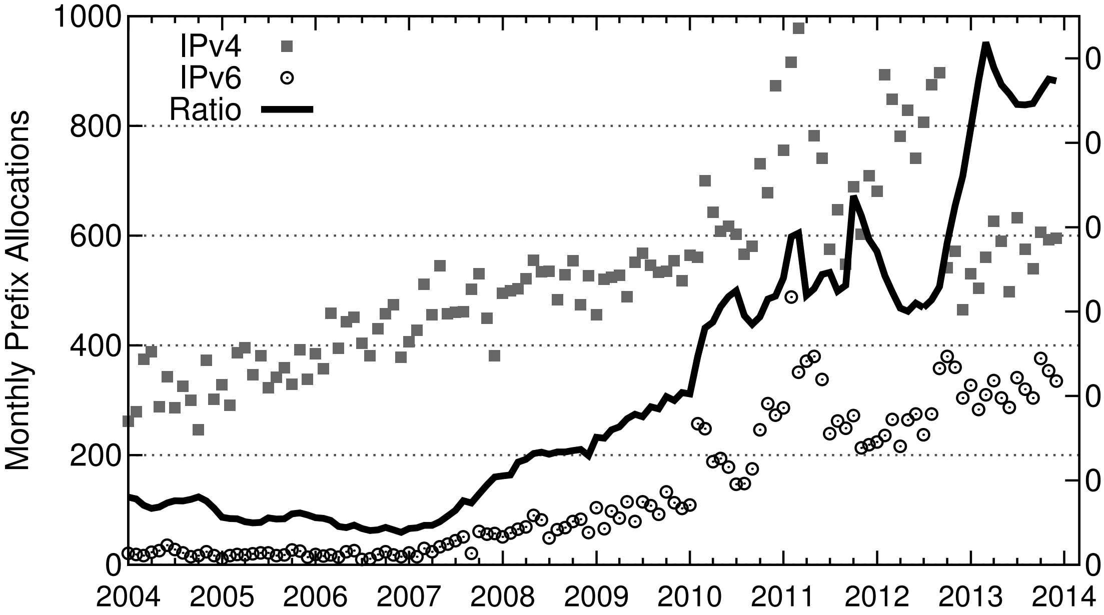


Figure 2.1: Prefixes allocated. IPv4 and IPv6 allocations accelerate leading up to IANA exhaustion in early 2011. IPv4 dropped in 2012 and was flat in 2013, while IPv6 trended upward.

allocations at the beginning of our dataset and just over 136K at the end. This represents an increase of 67K prefixes—or, less than a doubling of the number of IPv4 prefix allocations over the course of the previous ten years. The figure contains a ratio line to show the relative allocation of IPv6 versus IPv4. We find that at the end of December 2013, on a monthly basis, the ratio of IPv6 to IPv4 prefix allocations is 0.57 and following a general upward trend. Thus, *although there are still significantly more allocated IPv4 prefixes (136k) than IPv6 prefixes (18k), and the monthly rate of IPv4 allocations is still about double, we see the IPv6 allocation rate continuing to grow while the IPv4 rate declines.* The ≈300 IPv6 versus ≈500 IPv4 allocations per month suggest *IPv6 is, for the first time, being deployed or planned on a majority of new networks.*

find 2,217 IPv4 prefix allocations. This corresponds with APNIC's IPv4 pool dropping to a single remaining /8 and their "Final /8 Policy" being invoked, which caused a brief spike in allocated prefixes [14].

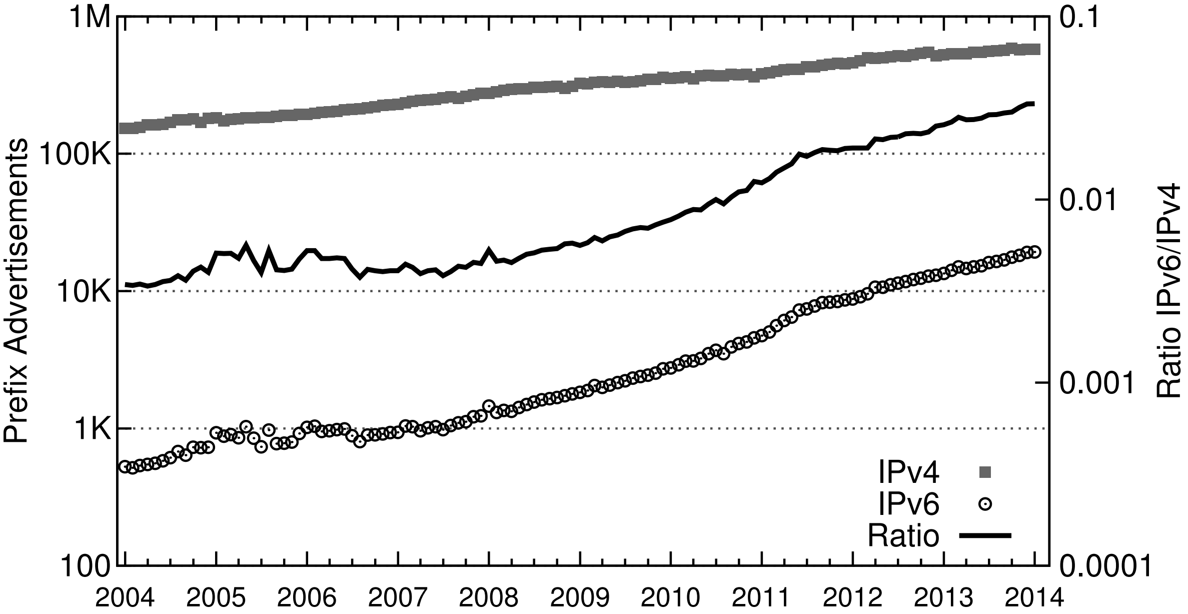


Figure 2.2: Number of advertised prefixes. Over ten years, IPv6 prefixes increase 37-fold, while IPv4 increase four-fold.

A2: Network Advertisement

Address allocation is a start, but to be used for Internet traffic IP addresses must be advertised in the global routing table. Therefore, our second metric is the number of IPv6 prefixes found in the Internet's global routing table. The Route Views project [166] and RIPE [150] both have a number of routers used for data collection, each peering with production Internet routers to obtain the routing tables from those peers. Based on routing table snapshots made available by these collection efforts, we obtain the number of prefixes announced on the first day of each month from January 2004 to January 2014. While these routing datasets are known to have biases, as we elaborate in § 2.4, these biases are not expected to affect the view of *globally-reachable* network prefixes.

Figure 2.2 shows the number of announced prefixes over time. We find 526 IPv6 prefixes on January 1, 2004. In January 2014, 19,278 IPv6 prefixes were advertised—an increase of 37-fold over the course of ten years. For comparison, we also show the average number of IPv4 prefixes advertised per day; these increased four-fold from 153K in 2004 to 578K by 2014.

While total and monthly allocations and advertisements are both still higher for IPv4, the rate of IPv6 allocations is increasing at a faster pace than IPv4. This is expected since IPv4 has been an Internet reality for 30+ years now, and, hence, the need for additional addresses is, naturally, incremental. *The rate of IPv6 prefix allocations is now where IPv4 was 8 years ago.* What is more, in 2013 the monthly volume of allocations of IPv4 has dropped significantly, to 2009 levels, likely due to the exhaustion events starting in 2011. In sum, *the allocation and advertisement numbers and rates provide the basis for wide-scale Internet adoption of IPv6 from the addressing perspective.*

2.3 Naming

Once IPv6 addresses are allocated and announced by routers, they must be used. The typical way addresses are referenced by Internet users and applications is via Domain Name System (DNS) names. Our next three metrics, therefore, focus on the prevalence of IPv6 support and use within the DNS ecosystem. A detailed description of DNS [132] is beyond the scope of this chapter, but we remind the reader of some basic terminology. The authoritative groupings of names in the DNS hierarchy are called *zones*. DNS domain names map to IPv4 address via *A* records and to IPv6 addresses via *AAAA* ("quad a") records. DNS servers that manage zones and return records are called *authoritative nameservers*, while servers that execute queries on behalf of (usually many) users are broadly called *resolvers*.

N1: DNS Authoritative Nameservers

Our first naming metric aims to understand the prevalence of authoritative nameservers that themselves can communicate via IPv6. While IPv6-addressed nameservers are not required for an organization to employ dual-stack IPv6 (i.e., it could serve AAAA records via IPv4 nameservers), we believe that the prevalence of such nameservers offers telling evidence on the adoption of IPv6, especially by content providers.

The top level of DNS has been IPv6-enabled since 2008, when root nameservers deployed AAAA records [101]. As of Jan. 2014, reports from Hurricane Electric show that 91% of the 381 top-level domains (TLDs) also have IPv6-enabled nameservers [116].

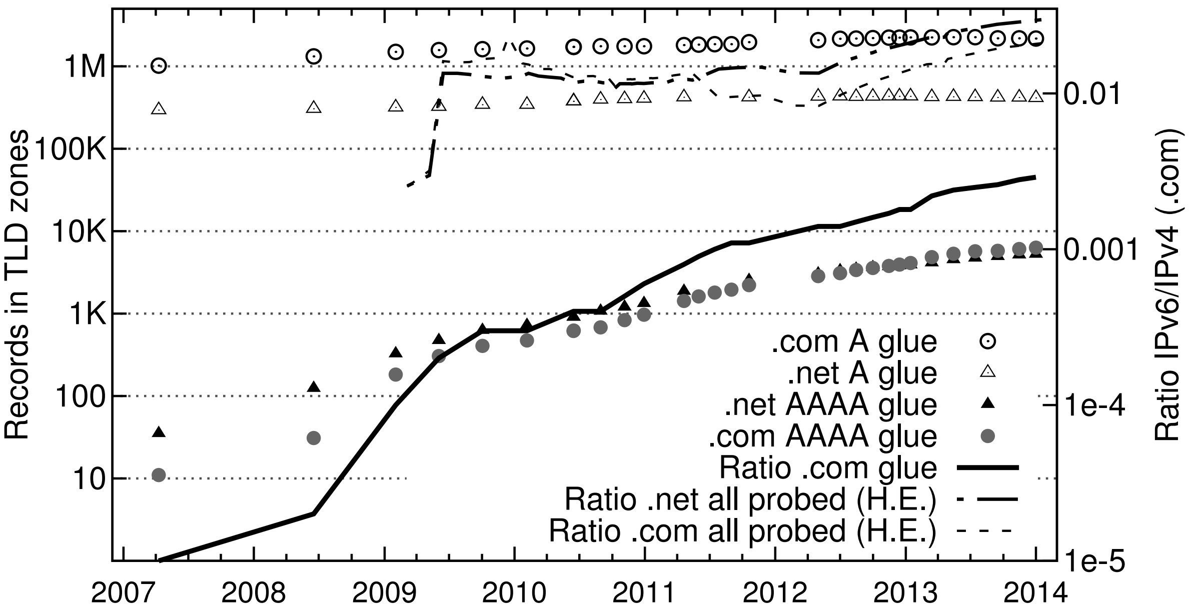


Figure 2.3: IPv6 nameserver and domain readiness. We see a steady increase in the glue records and a general increase in the probed domain names.

These include the largest TLDs, such as, .com, .net, .cn, etc. Of the thirteen .com and .net nameservers, all can serve AAAA but only two (a. and bmgtld-servers.net) are themselves IPv6-addressable. To understand the prevalence of IPv6 nameservers for second-level domains (e.g., example.com), we survey the .com and .net TLD zones. We analyzed sample .com and .net zone files between April 2007 and January 2014 to track the prevalence of DNS glue records for authoritative nameservers in the zones.

Figure 2.3 shows the number of A and AAAA glue records in the .com and .net zones over the last 7 years. IPv6-enabled nameservers (AAAA records) are dwarfed by IPv4 nameservers (A records), but both show long-term growth. Following the pattern of other metrics, the growth rate (second derivative) of IPv6-capable nameservers is higher than that of IPv4, and the ratio of AAAA to A is increasing. As of January 1, 2014, the ratio of AAAA to A glue records for .com is 0.0029. We also show the AAAA to A ratio from Hurricane Electric's published probing data, starting in 2009, wherein A and AAAA lookups for all domains in the zone are periodically performed [116]. Few nameservers in general have glue records in their zone, and IPv6-enabled ones seem to have this configuration even less often. The ratio of domains actually returning AAAA records via queries (vs. A) is an order of magnitude higher (0.02 for .com) than the glue record ratio.

In sum, *the authoritative nameserver data indicates low (0.0028 for .com glue, 0.02 for probed) but increasing (56% growth in 2013 for glue) support for IPv6 in the overall .com and .net zones.*

N2: DNS Resolvers

A second naming metric we consider is the prevalence of resolvers requesting AAAA records. Due to caching within the DNS system, this is not a direct measure of demand; however, the number of resolvers looking up AAAA records indicates the breadth of the use of IPv6, and speaks to the basic capability of resolvers to issue AAAA queries as well as the existence of at least some clients within a resolver's pool making AAAA requests. Viewed over time, this can be used to gauge whether demand for IPv6 content is widespread or only from pockets of the network.

Packet Datasets for .com and .net: As an initial assessment, we examine two large datasets of packet-level DNS query traffic to the .com and .net TLD authoritative nameservers on five sample days between June 23, 2011 and December 23, 2013. One dataset consists of IPv4 packets, while the second contains IPv6 packets. Both are from Verisign, the registry operating the .com and .net zones. The IPv4 queries were captured at between three and five of the 17 largest globally-distributed .com and .net TLD server clusters (e.g., in Feb. 2013, from Dulles, VA; New York, NY; San Francisco, CA; and Amsterdam, NL). Our IPv4 data includes transactions with several instances of the lettered X.gtld-servers.net TLD nameservers. These 24-hour IPv4 datasets range from 2.3Bn to 4.2Bn queries (except for the first IPv4 sample, which only included 30 minutes of data and $\approx$110M queries). These same 17 global Verisign clusters support IPv6 traffic. The IPv6 samples analyzed were also each 24 hours and consisted of 420M–1,052M queries. While the packet collection apparatus for both datasets is known to be lossy, we performed analysis that suggests no systemic network effects that would skew the measurements we report [43].

We note that the IPv4 and IPv6 datasets shed light on slightly different aspects of adoption. The IPv4 data give us insight into behavior of networks that are not using IPv6 for their naming infrastructure but that happen to have clients and resolvers that make AAAA queries, which—given support by the resolver—is largely determined by operating system and application behavior. On the other hand, the IPv6 packet data represent networks where DNS resolvers are able to communicate via IPv6 to the .com and .net nameservers, which suggests a more advanced level of IPv6 adoption. Thus, the latter may be more representative of the behavior of fully-capable clients, whereas the former represents clients that have software requesting AAAA records without the client necessarily having the ability to use them. For instance, Microsoft Windows XP clients that had a Teredo-configured [96] IPv6 addresses would, along with A, request AAAA records for names queried. However, these Teredo-based connections, either due to failure or preference, were found to be rarely completed by dual-stack hosts [178]. Windows Vista and later do not make AAAA queries when only a Teredo tunnel is available [50].

The resolver counts are, within an order of magnitude, stable over this period, with 3.5M seen in the most recent IPv4 sample and 68K in IPv6. Resolvers can service multiple, sometimes millions, of clients; so, this data represents the queries of many more than 3.5 million actual users. Although a single user or device can be configured to act as its own recursive resolver (e.g. by installing *bind*), we are more interested in the capabilities of resolvers serving multiple users. Therefore, in addition to aggregate results, we also report on a subset of the most active resolvers—e.g., enterprise or ISP-level—that send 10,000+ queries in a day.[2] There are 40K such active resolvers in the most recent IPv4 sample and 6K in IPv6.

In table 2.3 we show the percentage of resolvers in the two datasets that query for AAAA records. We see that nearly a third of all resolvers via IPv4 and three quarters via IPv6 make AAAA queries, as does the vast majority of active resolvers. Again, we stress this is not a measure of *use*, but an indication of *support* for IPv6 name resolution from within larger enterprises and networks. These numbers suggest that, *while AAAA records aren't in demand in every small corner of the network, at the organization or ISP level, IPv6 name resolution appears widely supported.*

[2] This threshold is arbitrary. We certainly miss smaller organization-level resolvers, but the included ones are very active.

Table 2.3: Percentage of resolvers making AAAA queries to .com and .net. While under a third of all IPv4 resolvers (N=3.5M in latest sample) make AAAA queries, most of the IPv6 packet population of resolvers (N=68K) does.

Resolvers	**2011-06-08**	**2012-02-23**	**2012-08-28**	**2013-02-26**	**2013-12-23**
IPv4 All	33%	28%	26%	30%	31%
IPv4 Active	90%	93%	83%	93%	94%
IPv6 All	74%	77%	74%	82%	76%
IPv6 Active	99%	99%	99%	99%	99%

N3: DNS Queries

In addition to the numbers of IPv6-addressable nameservers and resolvers requesting IPv6 addresses measured above, a final naming component we consider is the distribution of actual IPv6-related DNS queries. This speaks to *how* naming is being used in IPv6. We first determine whether IPv6 users are interested in the same names as IPv4 users. This will inherently be influenced by user population differences (e.g., regional and sample effects), including client OS differences (which construct DNS requests). Therefore, differences are expected. To measure the agreement between queried domains via A and AAAA records in our two .com/.net packet samples, we calculated Spearman's rank correlation coefficient (ρ) between the top 100K domains by each of the four types (IPv4 sample A and AAAA, and IPv6 sample A and AAAA). We limited analysis to the most-queried 100K domains in order to avoid skewing results by rarely-queried domains, such as typos, but we wanted a large number in order to capture a diverse set of content.

Table 2.4 shows the results. As a preface, rank correlation is, by definition, lower than set intersection, and the intersection numbers (not shown) for the three sets of domain list pairs range from 55% to 84%. We see that the domain rank correlations between the IPv6 and IPv4 samples via the same record types are moderate to strong ($\rho \approx 0.70$), indicating that *domain interest is similar between users of IPv4 naming infrastructure and those using IPv6.* Still, differences remain, and no clear trend is visible in this time period. Likewise, when we examine the within-packet-sample cross-type correlations (e.g., A vs. AAAA for IPv4), we see much less correlation. We suspect this is in part due to the fact that, whether an A or AAAA is requested by a given host is determined by applications and OSes in use, but there is greater similarity when examining the same application/OS patterns across

Table 2.4: Spearman's ρ rank correlations for top 100K domains queried by A and by AAAA via IPv4 and IPv6 ($P < 0.0001$ in all cases). There is moderate to strong (darker grays) correlation between IPv4 and IPv6 domains for same record types.

Domain Lists	2011-06-08	2012-02-23	2012-08-28	2013-02-26	2013-12-23
4.A : 6.A	0.65	0.73	0.70	0.70	0.57
4.AAAA : 6.AAAA	0.69	0.80	0.82	0.74	0.68
4.A : 4.AAAA	0.32	0.32	0.35	0.34	0.42
6.A : 6.AAAA	0.29	0.23	0.20	0.26	0.32

protocol packet samples. Some of these differences may be accounted for by the differences in set sizes. Across the five samples, the median percentage of queries that the top 100K domains account for is 55% for A via IPv4 and 60% for A via IPv6; for AAAA, it is 77% for IPv4 and 42% for IPv6. No clear trend is evident. In sum, the data suggests *differences in application use of IPv6 (which is expected due to the N1 results) but marked overlap in the domains of interest to networks using IPv6 resolvers versus those using IPv4..*

Turning from the names in the queries to the records, Figure 2.4 shows the top seven record types (plus all others under the "other" category) requested in the IPv4 and IPv6 packets on the five days of samples. We observe that, while there are still some differences in the distributions, *there is a statistically-significant convergence of query types over time (average monthly difference decrease of 1.65% with <0.05), and the query types in IPv6 are now much more similar to IPv4 than just two and a half years ago.*

2.4 Routing

Once IPv6 addresses are allocated and advertised, as well as potentially being named, the next prerequisite for using the new protocol is routing. While routing itself has many components, and we have already discussed IPv6 prefix advertisement in section A2, a key aspect of routing that deserves careful measurement is topology. The richness of the IPv6 topology, in terms of the number and length of paths and the connectivity of ASes, speaks to the resilience, and, thus, production-readiness of the network.

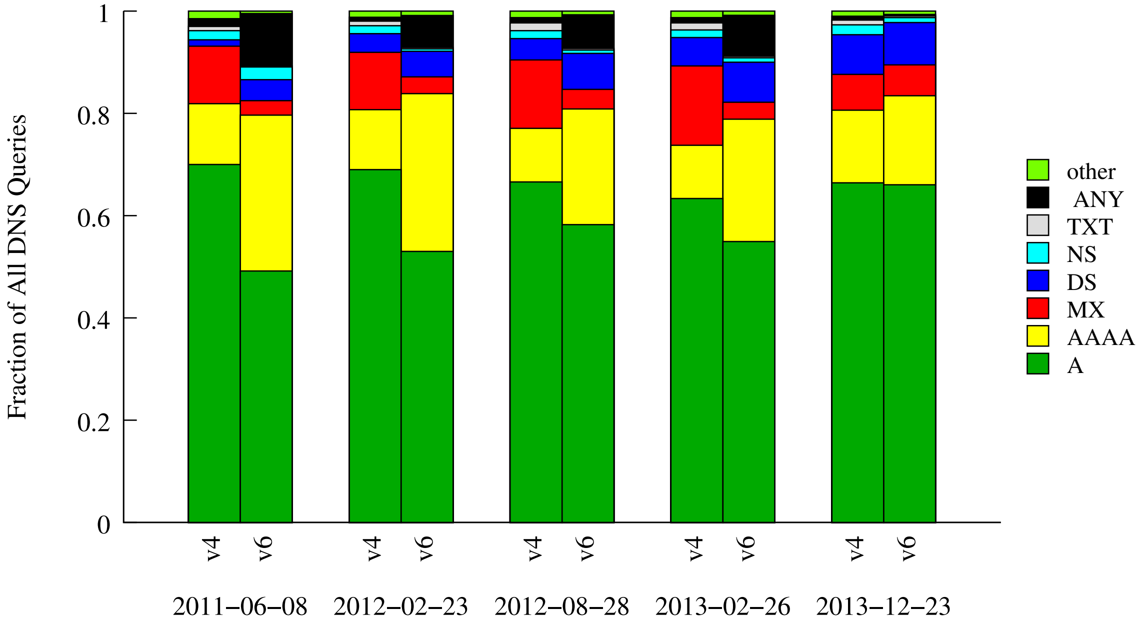


Figure 2.4: Breakdown of query types across five IPv4 and IPv6 DNS samples between Jun. 2011 and Dec. 2013 (key is in stack order). The distribution of IPv4 and IPv6 query types within each day converges over time ($p<0.05$).

T1: Topology

The IPv4 topology has been studied in depth (e.g., [125, 179]), but we also need to understand the relationships between organizations with respect to external IPv6 routing capability and connectivity to understand the overall strength or brittleness of the network. As we did for the advertisement metric (A2) we use all of the routing table snapshots collected by Route Views [166] and RIPE-NCC RIS [150] between Jan. 2004 and Jan. 2014 in the following.

Routing Table View Biases: Before we delve into the insights that these data afford, it is important to understand possible bias. As noted by earlier studies (e.g., [79]) the global public routing datasets available (e.g., Route Views, RIS) suffer from at least two forms of bias. The first is geographic, in that global routing data is collected from a finite set of collectors, whose global distribution is not uniform, leading, for example, to fewer samples from the African continent. The second bias stems from most of the data in these collections coming from volunteer networks that turn out to generally be large top-tier ISPs. Therefore, many peer-to-peer paths between smaller ISPs are not visible in the data, since these routes

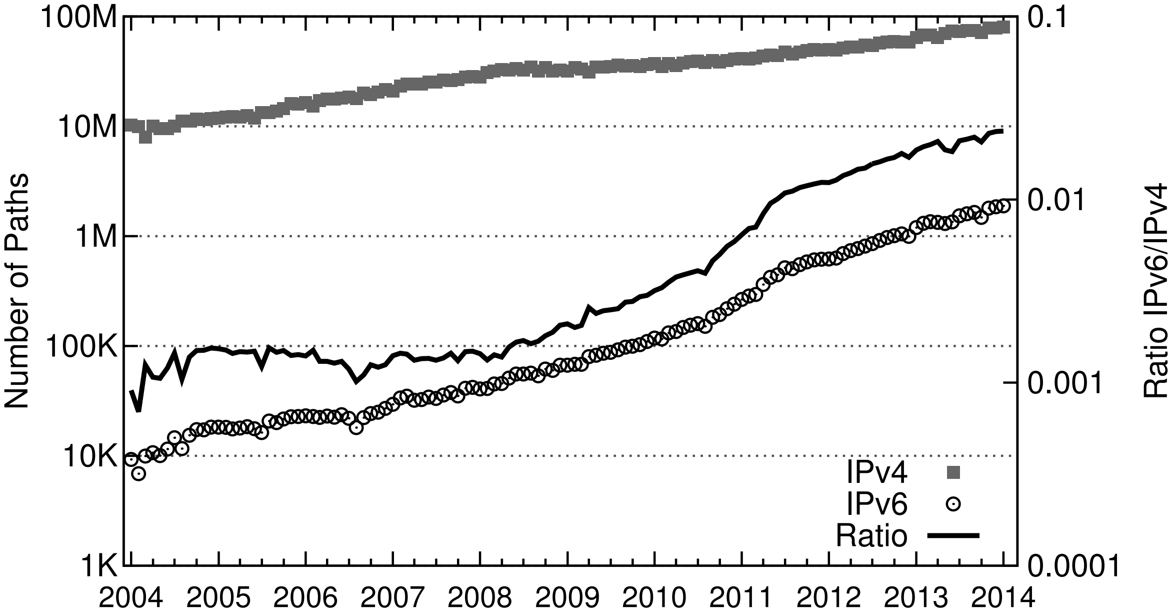


Figure 2.5: Number of globally-seen IPv4 and IPv6 paths. There is a 110-fold increase in IPv6 paths over ten years.

are never propagated to the top-tier ISPs. These biases are a limitation of the data. However, we believe that even though imperfect the data still yields useful information about IPv6 adoption for the following reasons: (*i*) no substitute data lacking bias exists (e.g., traceroute also has bias); (*ii*) the view of the global routing infrastructure of ISPs whose routing data is represented is a real view from their perspective, and any path or AS counts observed are, at worst, lower bounds; (*iii*) we have no reason to believe that the bias present in these data for IPv6 differs systematically *relative* to that for IPv4, suggesting that looking at ratios of adoption, especially over time, is reasonable; and (*iv*) in the cases of counting globally-visible prefixes or ASes seen supporting IPv6, the fact that some local paths are missed does not speak to the *global* adoption state. Therefore, we present the data knowing full well it is a less-than-full statement on the routing state. We encourage the community to collect and refine our analysis using better data.

AS Support and Connectivity: We first examine the number of ASes supporting IPv6 globally as well as the number of unique globally-visible AS-paths (i.e., the paths with unique AS sequence). Both are indicators of IPv6 adoption, mostly at the service provider level. AS adoption is indicative of *support* for IPv6, while the number of AS-paths is an indicator of maturing *connectivity* between ASes. We omit the figure showing AS-level

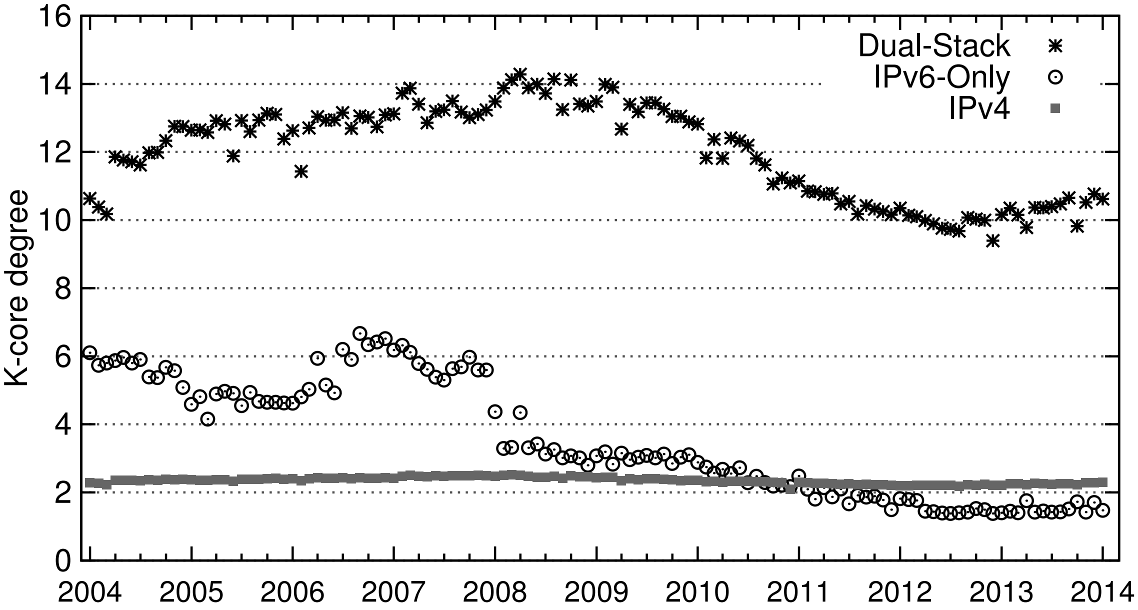


Figure 2.6: AS centrality. Pure-IPv6 and dual-stack ASes are becoming more prevalent at the edge.

adoption in favor of Figure 2.5, which shows the number of unique IPv6 and IPv4 paths announced on the first day of each month. We observe that the number of IPv6 paths has a 110-fold increase from January 2004 to January 2014, while there is only an eight-fold increase in the number of IPv4 paths. However, the IPv6 to IPv4 ratio is only 0.02 in January 2014, indicating the IPv6 routing mesh is still at an early stage of maturity. AS-level support for IPv6 is not shown, but follows a faster upward trend, with an 18-fold increase (versus two-fold for IPv4) and the current ratio of IPv6 to IPv4 ASes is 0.19 – almost ten times the path count ratio. As expected, the indicator of ASes *supporting* IPv6 leads the measure of *connectivity*. Again, we note that, because of possible bias in the view afforded by this data, raw numbers of the IPv4 and IPv6 paths seen are less meaningful than is the overall ratio.

AS Centrality: To understand the topological position of IPv6 ASes, we next compute the *k-core degree* of each AS in the topology graph. A k-core of a graph is the maximal subgraph in which every node has at least degree k. A node has k-core degree of N if it belongs to the N-core but not to the $(N+1)$-core. As used in [83], this measure represents a natural notion of the *centrality* of ASes. In other words, ASes with a high k-core represent well-connected, typically large, ISPs, while those with low k-core represent edge or stub networks. We show the average k-core degree of ASes in Figure 2.6. We find that dual-stack ASes have a much higher degree of centrality than other ASes. In 2004, the pure IPv6 ASes were located in a relatively central position. However, we see pure-IPv6 ASes, a small fraction of all, becoming more prevalent at the edge after 2008. Our results are in accordance with those of CAIDA [56], who report that IPv6 is largely deployed at the core but lags in edge networks. Note [56] uses a deeper and more robust analysis of these same public datasets, wherein, notably, they filter out transient links. The numbers we find indicate *dual-stack becoming more widely deployed among well-connected central ISPs, while single-protocol networks are mostly those at the edge. In other words, the older edge networks are the laggards.*

We caution that studying native IPv6 topology is useful but insufficient. Transition from IPv4 to IPv6 introduces a co-dependence between the protocols. Therefore, unlike when studying IPv4 topology independently, when studying IPv6, we must consider the parts of IPv4 that glue together “islands” of IPv6. An in-depth analysis is beyond scope, but we point readers to recent work in [56].

2.5 End-to-End Reachability

Having dealt with the prerequisites of addressing, routing, and naming in the previous three sections, we now turn to the readiness of Internet end hosts to use IPv6. We split this into two metrics for the readiness of service-level devices and client-level devices.

R1: Server-Side Readiness

Obviously, wide-scale adoption requires services to be capable of handling IPv6 traffic; therefore, our first approach to end-host readiness involves assessing prevalence of IPv6-enabled services.

While not indicative of all services, one way to assess IPv6 service penetration is to measure popular web servers. Much like Nikkhah et al [138] and with congruent results we use Alexa [9] to determine the most popular web sites. We then determine which sites have

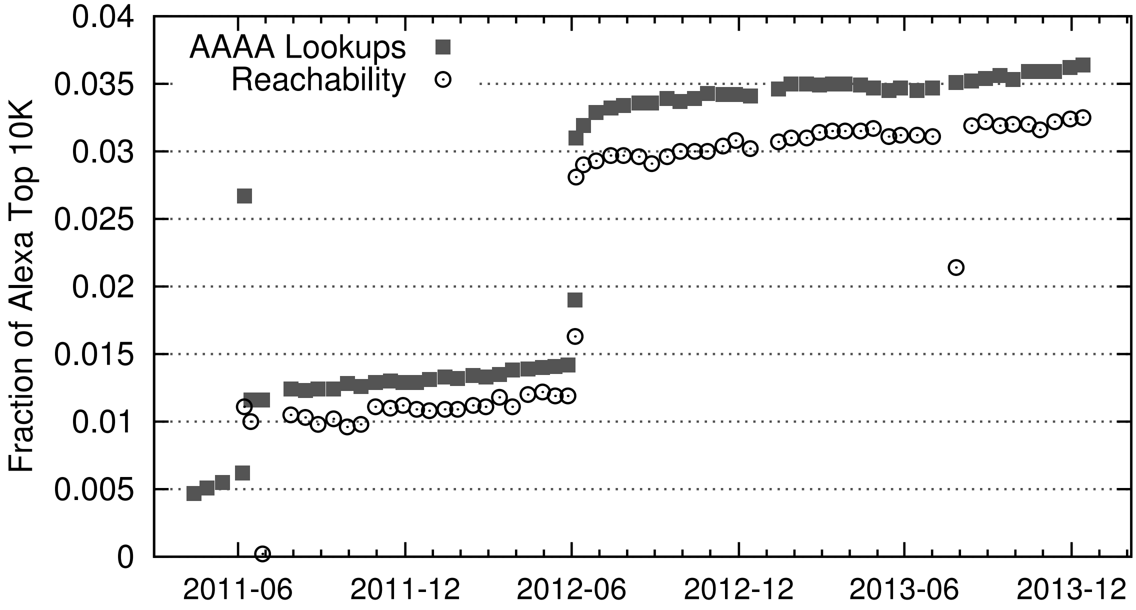


Figure 2.7: Fraction of top 10K sites with AAAA records and reachable via IPv6. Two discontinuous jumps correspond to World IPv6 Day 2011 and World IPv6 Launch 2012.

a AAAA record in DNS, and, for those that do, we then test reachability of the web site via a tunnel to Hurricane Electric. Ideally, this metric tries to assess the server, but we have no way to do so without also assessing the path to the server. Hence, our measurements offer an approximation. We have been probing the top 10K web sites for AAAA records since April 2011 and for reachability since June 2011.[3] Figure 2.7 shows our results. We first note a jump in June 2011 that corresponds to World IPv6 Day. We find a roughly five-fold increase in AAAA records available at that point. However, we also see a nearly immediate fallback. This is understandable given that the stated goal of that day was merely to serve as a “test flight” of IPv6 capabilities, rather than to permanently enable IPv6 services [102]. Subsequent to this drop off, in spite of the limited goal, we find that World IPv6 Day 2011 is responsible for a sustained two-fold increase in the IPv6-capable web sites. In the following year, the June 2012 World IPv6 Launch Day also resulted in a sustained doubling of AAAA records. Further, aside from the two jumps, *we find a slowly growing trend across time with over 3.2% of the Alexa top 10K now being reachable via IPv6.* It is notable what an impact concerted community efforts, such as the two IPv6 readiness/launch days can have on IPv6

[3] Our probing data is available [11].

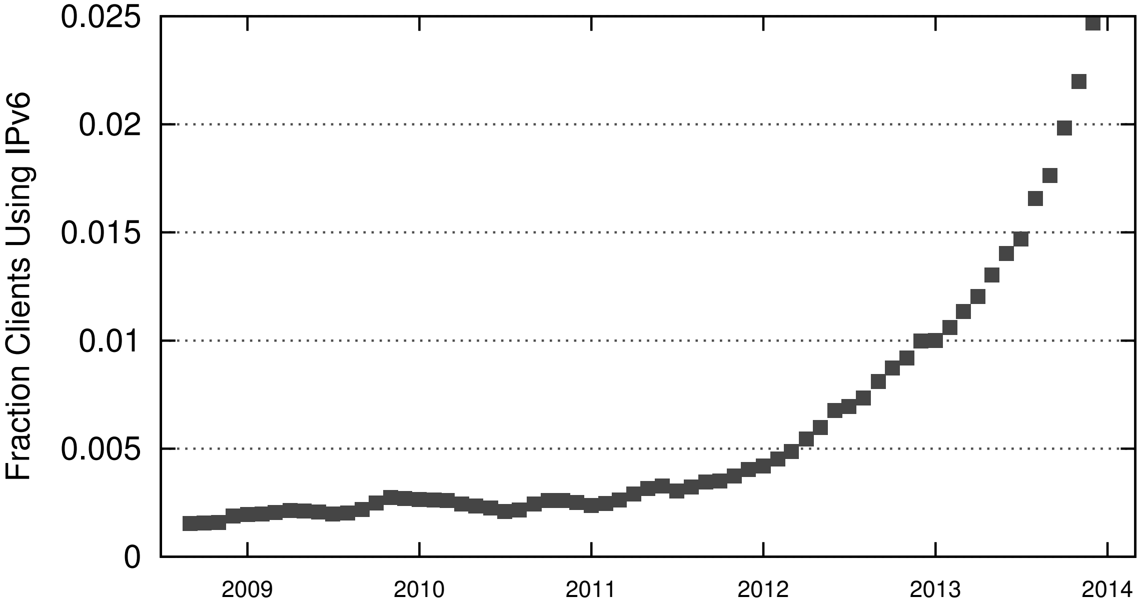


Figure 2.8: Average monthly fraction of clients able to access Google over IPv6. 2.5% of clients use IPv6, but this number has been growing sharply. The most recent two-year annual growth rate averages 150%, a more than doubling each year.

server readiness.

The second set of points on the plot show reachability. The data shows that most of the hosts for which we find AAAA records are also reachable. Further, the reachability trends generally mirror those for web servers having AAAA records. Our results generally agree with [138]. In conclusion, *while only about 3.5% of the top most popular websites are IPv6-ready, there has been significant growth in the last three years, and large jumps are possible.*

R2: Client-Side Readiness

In addition to IPv6-capable services, clients need to be IPv6-enabled as well. This metric examines the ability of client systems to employ IPv6, subsuming all that is required on the client side (i.e., working IPv6 network transport, DNS, operating system, etc.).

Google makes aggregate data about client adoption of IPv6 available on an ongoing basis [78]. Their experiment consists of adding a JavaScript applet to search results from `www.google.com` for a randomly sampled set of users [38]. The script first performs a name lookup on one of two experimental host names and then sends a request to the IP address returned in the DNS response. In 90% of the cases the script chooses a name representing a dual-stacked server, while in the remaining cases a name representing a IPv4-only server is chosen for comparison purposes. The addresses point to 2–5 data centers (in Asia, the US, and Europe). The experiment is conducted millions of times per day. Note that, as with the R1 measurements, this data again conflates the client capabilities with those of the path from the client, and, therefore, this is an approximation of the ideal metric.

Figure 2.8 shows the average monthly fraction of clients that connect to Google via IPv6 over the last 5+ years. The plot shows a growth factor of 16 over the course of the dataset—from 0.15% in September 2008 to 2.5% in December 2013. Further, most of the growth comes in the last two years, where the ratio increased markedly, by 125% in 2012 and 175% in 2013, more than doubling each year. As discussed in section A3, this measure is probably somewhat optimistic; since Google has many direct private peerings to ISPs, some clients may be able to reach Google by IPv6 but not other content. However, these numbers are roughly in line with those reported in another large client study [178], which found that although 6% of a global sample of clients were IPv6-capable, only 1-2% of dual-stack preferred IPv6. In sum, *the data shows very strong growth in Google clients' ability to use IPv6, especially in the last two years.*

2.6 IPv6 Usage Profile

While the metrics and data sketched in the previous sections set the stage for IPv6 adoption by measuring addressing, routing, naming, and end-host capabilities, in this section we aim to directly assess IPv6 traffic "in the wild". That is, we aim to understand the operational characteristics, or usage profile, of how IPv6 is actually employed by those that have adopted it.

U1: Traffic Volume

Our first traffic-related metric simply aims to understand how much of Internet traffic is using IPv6. We begin by introducing the Internet traffic datasets we contribute.

Arbor Internet Traffic Data: We assembled two datasets describing the traffic traversing customer networks monitored by devices from Arbor Networks, a provider of traffic analytics and security devices for large networks. Arbor's customers include a significant number of large ISPs in the world, and, in aggregate, they have visibility into nearly half of Interdomain (i.e., Internet) traffic. These two datasets consist of traffic summaries (daily netflow statistic aggregates, including protocol, port, and volume information), from peering, aggregation, and customer-facing routers at participating networks. The first, older, dataset tracked daily peak five-minute traffic volume for both IPv6 and IPv4, measured a sample of 12 Arbor customers providing anonymous data, and included data from the second quarter of 2010 through February 2013. The second Arbor dataset (for 2013) reports daily average volume, includes approximately 260 providers of anonymous data that together represent an *estimated 33-50% of global Internet traffic*, and collects data using the same methodology as Labovitz et al. [115] (now with more providers). We include the larger dataset starting in January 2013; however, even the smaller, 12-provider sample covers an aggregate of over 400 routers and 55K links, representing a cross-section of different-mission and varying-size Internet organizations. Both samples include global Tier 1 ISPs, national and regional Tier 2 ISPs, content/hosting providers, and universities. In the newer dataset, which represents 19 Tier-1 and 92 Tier-2 providers plus over 100 enterprises, content providers, etc., each continent is represented, as are both fixed-line and mobile Internet providers. We refer to the older (smaller) and newer (larger) datasets as $\mathcal{A}$, and $\mathcal{B}$, respectively. In the fourth quarter of 2013, the daily median Internet traffic in dataset $\mathcal{B}$ was 58 Tbps.

We normalized the traffic measurements by the number of Arbor Networks providers in the samples, to distinguish organic traffic growth from growth due to changes in the number of customers. Figure 2.9 shows the median daily peak and average traffic volume for each month in our two datasets, respectively. Since one set of points and line represent 5-minute

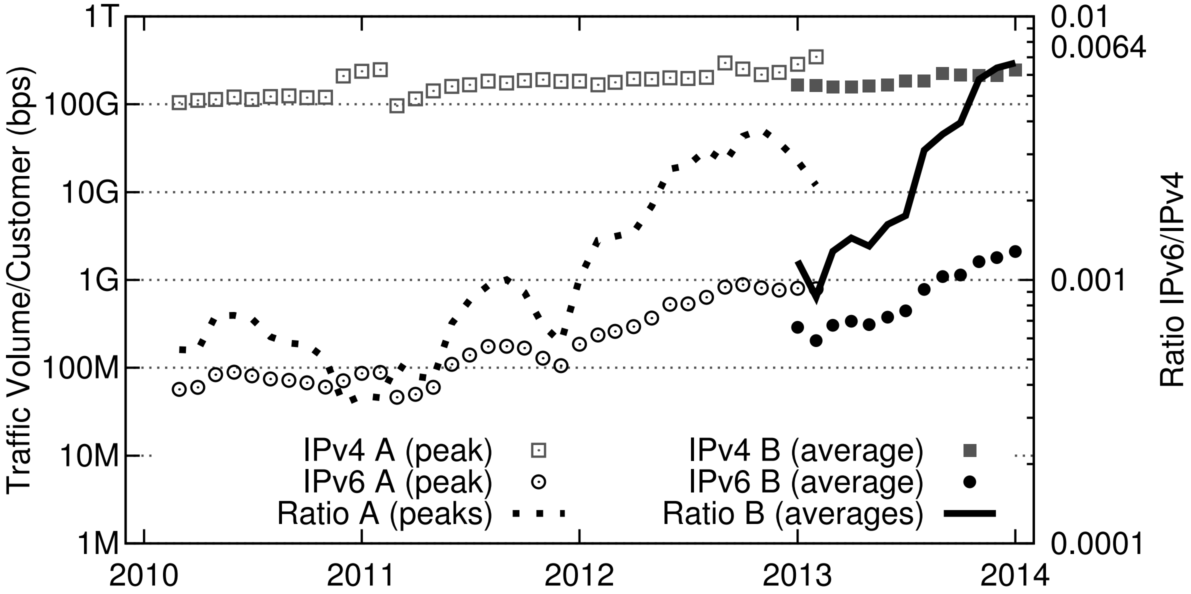


Figure 2.9: Global Internet traffic data in two datasets: $\mathcal{A}$ for Mar. 2010–Feb. 2013, monthly median *peak* 5-minute volume for 12-provider sample; and $\mathcal{B}$ for 2013, the monthly median of daily *average* traffic volume for ≈260 providers. IPv6 is 0.6% of traffic, and 2-year growth relative to IPv4 is 451% annually.

peaks, and the other averages, we present the data as separate points. This difference helps explain the shift between the lines visible for the two months for which we had both datasets in January and February, 2013.

The figure shows that IPv6 is still dwarfed by IPv4 traffic (by roughly two orders of magnitude). However, both IPv4 and IPv6 peak traffic volumes are generally increasing, and IPv6 is on a strong upward trajectory. Over our measurement period we find roughly an order of magnitude increase in the median daily peak volume for both protocols. As the ratio line (representing the raw traffic, not normalized by customers) shows, we do find a significant rise in IPv6's relative contribution to Internet traffic. In March of 2010, the ratio of IPv6 to IPv4 is 0.0005, while in December 2013 it is 0.0064—a 13-fold increase. In sum, *while, overall, the proportion of IPv6 traffic on the Internet is still under one percent, it has grown, relative to IPv4, by 433% percent year-over-year in 2013, 469% in 2012, and 71% in 2011, a rapid pace.*

U2: Application Mix

Another important metric when studying the IPv6 usage profile is what applications are used. This can, for instance, inform our understanding as to whether IPv6 is being used for typical user activity or for specialized use, as was reported in the past (e.g., [111]).

Table 2.5: Application mix (%) between Dec. 2010 and 2013. HTTP/S increases from 6% to 95% for IPv6, surpassing IPv4, while back-end services (e.g., dns, ssh, rsync) decline significantly. IPv6 usage looks much more like IPv4 than in the past.

Application	**Dec 2010**	**Apr/May 2011**	**Apr/May 2012**		**Apr–Dec 2013**	
	IPv6	**IPv6**	**IPv6**	**IPv4**	**IPv6**	**IPv4**
HTTP	5.61	11.81	63.04	62.40	82.56	60.61
HTTPS	0.15	0.88	0.39	3.91	12.66	8.59
DNS	4.75	9.11	4.09	0.14	0.33	0.22
SSH	0.56	3.73	2.65	0.11	0.27	0.20
Rsync	20.78	5.11	2.65	0.00	0.13	0.00
NNTP	27.65	5.84	1.03	0.13	0.00	0.25
RTMP	0.00	0.05	0.11	2.39	0.00	2.74
Other TCP	*	*	18.72	3.20	1.66	4.08
Other UDP	*	*	1.73	11.90	0.27	2.82
Non-TCP/UDP	*	*	4.94	14.10	2.11	20.21

We have application information from Arbor Networks for the same traffic samples described earlier (although we are missing IPv4 data prior to 2012). The network flow monitors classify traffic by port number, and, hence, the categorization may not be completely accurate. For instance, we note that HTTP port 80 is often used for tunneling non-web applications, as it tends to be open in firewalls. However, we believe this categorization is useful as a first order analysis. Table 2.5 shows the proportion of traffic for each application that makes up at least 1% of either IPv6 or IPv4 traffic. We see in the 2012 and 2013 samples that HTTP dominates within both IPv6 and IPv4. In the 2013 sample, HTTPS has increased significantly in IPv6, surpassing even that in IPv4. Likewise, we see a dramatic uptick in HTTP/S traffic across the four years. The large fraction of likely web traffic now observed, at 95%, is a major departure from the patterns observed in earlier studies of IPv6 dating from 2008 and early 2009, which report HTTP traffic volume below one percent [111, 160]. Interestingly, Karpilovsky [111], Savola [157] and Hei [87] also report large amounts of DNS traffic (e.g., 80-90% in 2008 according to Karpilovsky *et al.*), which continued to rank highly in our own data until 2012. Our 2013 sample is the first time we see DNS decreasing to IPv4 levels and actual content (HTTP/S) surpassing even what is seen for IPv4; this is a significant evolution. We note that one possible factor at play in the reduction of DNS traffic is the behavior of newer Windows operating systems (starting with Vista), which, as mentioned earlier, no longer make AAAA queries when their only IPv6 connection is via Teredo [50]. Likewise, we see a substantial reduction in IPv6 NNTP traffic [62]. We believe the previously large volume seen was related to (*i*) the fact that a large NNTP path is part of the traffic sample and (*ii*) NNTP was heavily used for piracy over IPv6 when several USENET services that otherwise required paid subscriptions offered free IPv6 access[4].

Additionally, in the 2013 sample, we find 4.04% and 27.11% of the traffic volume not ascribed to a particular application for IPv6 and IPv4, respectively, a large decline from 2012 for IPv6 and a smaller one for IPv4. However, the distribution of non-identified traffic is different between IPv6 and IPv4. For example, while most of the bytes in IPv4 are non-TCP/UDP at 20.21%, such traffic only contributes 2.11% of the overall bytes in IPv6. Although we were unable to investigate this "other" category more deeply, we speculate that the usage of peer-to-peer and similar popular non-well-known-port applications still differs between IPv4 and IPv6 in the TCP/UDP categories, while ICMP and tunneling protocol mix differ in the non-TCP/UDP category. The method of aggregation of untracked protocols for 2010 and 2011 does not allow comparison, unfortunately. To summarize, *over the last three years we see a dramatic evolution in IPv6 application use, wherein content packets now far outnumber infrastructure service packets (DNS, ICMP), and the profile now resembles IPv4.*

U3: Transition Technologies

IPv4 and IPv6 coexistence is greatly complicated by the lack of backward compatibility. In what is now acknowledged as one of the most significant IPv6 design limitations, native IPv6 network devices cannot communicate with their IPv4 counterparts without an explicit network translation layer [127]. As a result, the success of any large-scale IPv6 transition depends on the complex interplay between the cost and scalability of translation

[4] e.g., http://www.techjawa.com/2011/02/10/guide-get-free-usenet-access-with-ipv6/

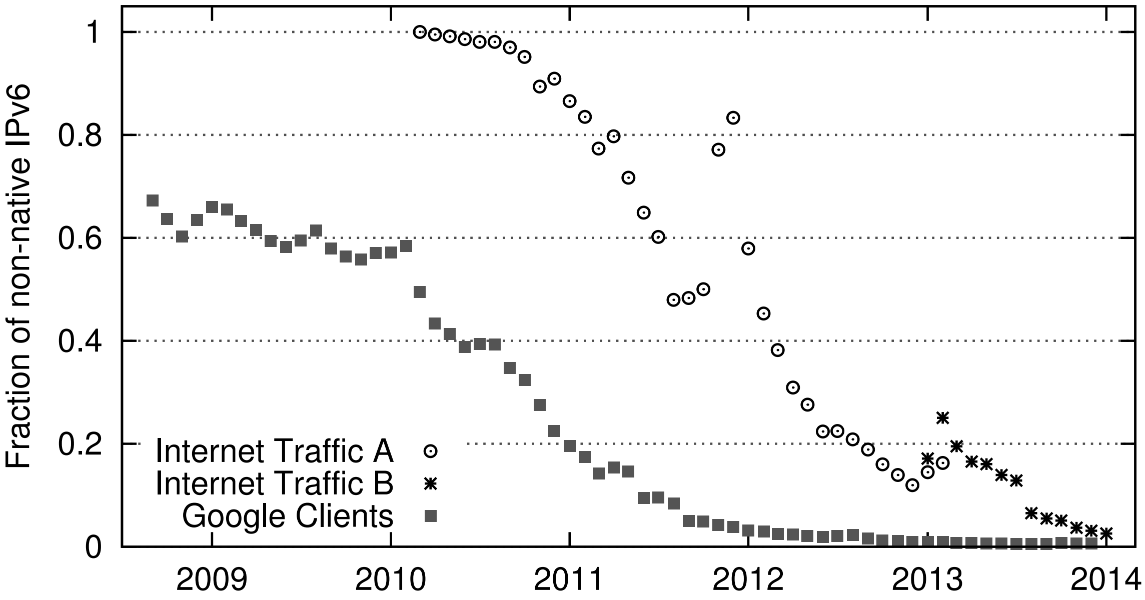


Figure 2.10: Fraction of IPv6 traffic carried by the two most prevalent transition technologies in the Internet traffic and Google client samples. Non-native IPv6 traffic was the majority of packets 3-4 years ago, but currently represents less than three percent of traffic and one percent of Google clients.

technologies and the commercial incentives (or disincentives) motivating the transition to native IPv6 infrastructure. A common transition technology is tunneling. Tunneling technologies interconnect "islands" of IPv6 using encapsulation across IPv4 infrastructure, or vice versa. In addition to tunnels, Teredo [96] provides IPv6 connectivity to hosts behind IPv4-NATs using UDP-encapsulation. Our next metric aims to understand the prevalence of the most common transition technologies being used in the wild where end-to-end IPv6 addressing is not fully in place. As IPv6 matures, we expect a smaller fraction of its traffic to use these technologies and more of it to be native.

Both the Google client and Arbor Networks datasets described earlier include information on the prevalence of various transition technologies. The Google perspective provides a view on the capabilities of end hosts, while the Arbor view is an assessment of actual Internet traffic. Figure 2.10 shows the prevalence of non-native IPv6—which is defined as Teredo and IP protocol 41 traffic (used by 6to4 and 6in4). The two Internet traffic data points $\mathcal{A}$ and $\mathcal{B}$, refer to the Arbor Internet traffic datasets, $\mathcal{A}$ and $\mathcal{B}$, described earlier. The

Google data shows that, while in 2008 only 30% of IPv6-enabled client end-hosts could use native IPv6, that number has increased to above 99% over the last four and a half years.

In 2010 we find the Internet traffic data shows nearly all IPv6 traffic using some tunneling technology. However, as of the end of December 2013, nearly 97% of the traffic is now native. We note that, of the tunneled IPv6 traffic in late 2013, IP protocol 41 dominates, contributing over 90% of the tunneled volume compared to less than 10% for Teredo. The Arbor numbers between mid-2011 and February 2012 correspond roughly to earlier measurements from that time, (e.g. [156] and [178]), while the Google numbers show much less transition technology used than those studies; this may be explained by the direct peerings phenomenon, described above. Overall, the data shows that *native traffic is now the vast majority of IPv6 traffic, a dramatic change from just three years ago. The Internet's IPv6 traffic is now real IPv6.*

2.7 Performance

A crucial metric of IPv6 adoption is performance—which can mean different things depending on the measurement perspective we take (e.g., the speed of a site to load for a user, or, for an ISP, the bandwidth across peering links). Several works predating IPv4 address exhaustion offer initial results in the area of IPv6 performance (e.g., [33,38,170,181]). Further, there are some performance results since the exhaustion milestones [56, 138]; both of these latter studies report that performance over IPv6 paths that align with IPv4 at the AS-level is similar for the two protocols but differs when paths diverge. More recent work exploring a methodology for passive measurement of IPv6 and IPv4 performance was contributed by Plonka and Barford [141], who found great variability in relative performance of the protocols in a campus traffic sample. The data we present here aims for a global and longitudinal, if less granular, examination of relative IPv6 network performance.

P1: Network RTT

We suspect that hardware, software, and configuration differences could result in different quality of data transmission in IPv6 versus IPv4; indeed, previous measurements and ones we report here have shown differences. Although actual client-to-service network performance for large global sets of clients and services would be a more ideal metric, we use the approximation of average 10- and 20-hop round trip time (RTT), tested from dozens of global perspectives, as a proxy. This hides the details of path differences and heterogeneity of end points, allowing a simple apples-to-apples comparison of raw network performance over the same number of IPv4 versus IPv6 nodes. We don't claim that this is the only or the best IPv6 network performance metric, but it serves as a reasonable approximation of long-term performance evolution.

Our analysis is conducted on the traceroute-based performance data collected by CAIDA Archipelago Measurement Infrastructure (Ark) [26] to measure RTT in IPv4 and IPv6. Globally-distributed Ark monitors probe all IPv4 /24s and all announced IPv6 prefixes continuously. We analyze data from December 2008 to December 2013. While this dataset is also the basis of earlier work ([56]), we re-analyze an updated longitudinal version to observe performance in the context of the other metrics we report.

Figure 2.11 shows the median RTT with hop distances 10 and 20 for each month. We find that in 2009 RTTs were roughly 1.5 times longer for IPv6 than for IPv4. While the IPv4 RTTs have increased slightly over this time period, IPv6 RTTs have decreased slightly. In 2013, the RTT for hop distance of 10 is almost identical for IPv4 and IPv6. IPv6 had better RTTs than IPv4 at 20 hops in 2012 through mid-2013. To compare relative performance, we show the IPv6 to IPv4 ratio for the 10-hop RTT, as it has been less favorable for IPv6. Since, the better the performance the smaller the RTT, we show the ratio of the reciprocal of RTT for each protocol. As noted in [56], the sample of IPv6 data is small and the results might be dominated by a few paths. Also, evolving tunnel use likely impacts RTT and hop count. Thus, we cannot conclude that IPv6 has better RTT performance than IPv4, overall. However, *the long-term trend shows clear improvement for IPv6, and it has approached parity with IPv4 (≈95%), for the first time, in the last several years.*

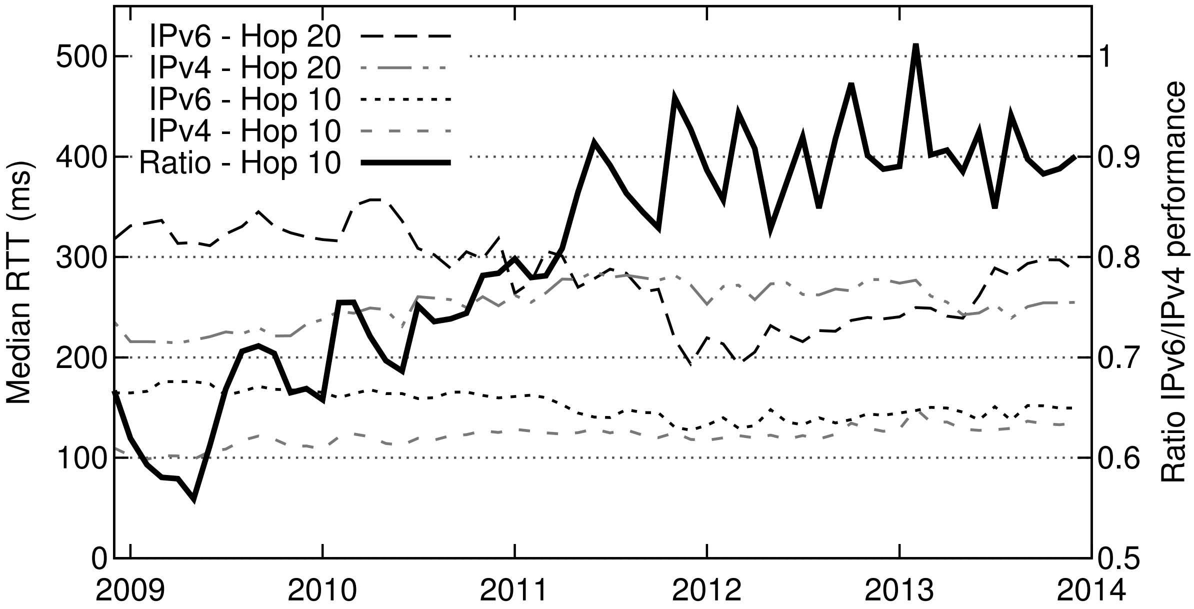


Figure 2.11: Median Round Trip Time (ms) with hop distance 10 and 20 for IPv4 and IPv6. IPv6 showed poorer performance before 2010, but the last several years have seen performance converging to within about .90–.95 of IPv4.

2.8 Cross-metric Findings

We analyzed a set of twelve metrics for assessing the adoption of IPv6 based on a large set of ten longitudinal datasets—some original, some publicly available, most large and global. In some cases, we updated and replicated similar measures reported in years past (e.g. RIR allocation data, performance data); in others, we presented new large data samples (e.g. the Internet traffic data, the Verisign IPv6 .com and .net DNS packet data). Here, we first highlight what the current state of adoption looks like, when examined through the entire broad set of perspectives we consulted. After that, we provide rough estimates of where we expect adoption to be in five years, based on recent trends.

2.8.1 IPv6 Present

The Value of a Broad Approach: In Figure 2.13 we show five-year ratios of IPv6 relative to IPv4 for seven of our metrics. Most prominent is the result that different metrics give entirely different insight into the adoption of IPv6, and suggest orders of magnitude different progress. For instance, while roughly 36% of new monthly (and 12% of cumulative) allocated prefixes are IPv6, we find just 0.63% of average traffic is carried over IPv6—a two-order-of-magnitude difference. These differences across metrics serve to highlight that multiple viewpoints must be considered to fully understand the progression and true state of adoption. In addition to the differences seen when examining different types of data—i.e., different prerequisites or operational characteristics—differences within the same type, but from distinct perspectives, are also important to consider. For example, recall that the difference in non-native IPv6 traffic as seen by Arbor Networks versus that seen by Google in metric U3 has been noticeable. What is more, *the order of adoption, as reflected by the relative rank of metrics, generally follows the prerequisites for IPv6 deployment* (e.g., allocation precedes routing, which precedes clients, which precedes actual traffic).

IPv6 is Now Real: Compared to prior work and earlier data, *we see in our recent data a dramatic qualitative and quantitative evolution in the state of IPv6 adoption, indicating a major shift in how the protocol is being used in the last three years.* Table 2.6 summarizes the usage profile of IPv6, and how it has evolved over this time. Traffic data shows that IPv6, while just 0.63% of measured Internet packets, is growing at a rate of over 400% in each of the last two years; application mix data shows content packets now dominating traffic; transition technology data shows that virtually all IPv6 traffic and Google clients are now native; and, performance data shows IPv6 now nearly on par with IPv4. *IPv6 is finally being used natively, for production, and at a rapidly-increasing rate.*

Table 2.6: Measures of actual operational characteristics of IPv6, recently and three years ago. These suggest that IPv6 is now mature. We contend that IPv6, as a real, production protocol, has finally come of age.

Metric: Operational Aspect Measured	**IPv6 Status at End of:**	
	2010	**2013**
U1: IPv6 Percent of Internet Traffic	0.03%	0.64%
U1: 1-yr. Growth vs. IPv4 (*Mar-2010 – Mar-2011)	−12%*	+433%
U2: Content's Portion of Traffic (HTTP+HTTPS)	6%	95%
U3: Native IPv6 Packets vs. All IPv6	9%	97%
U3: Native IPv6 Google Clients	78%	99%
P1: Performance: 10-hop RTT^{-1} vs. IPv4	75%	95%

Inter and Intra-Regional Differences: Our cross-metric analysis allows us to see stark regional differences in adoption. For example, when breaking down the cumulative allocation data by RIR (A1), we find RIPE responsible for 46% of allocations, while ARIN

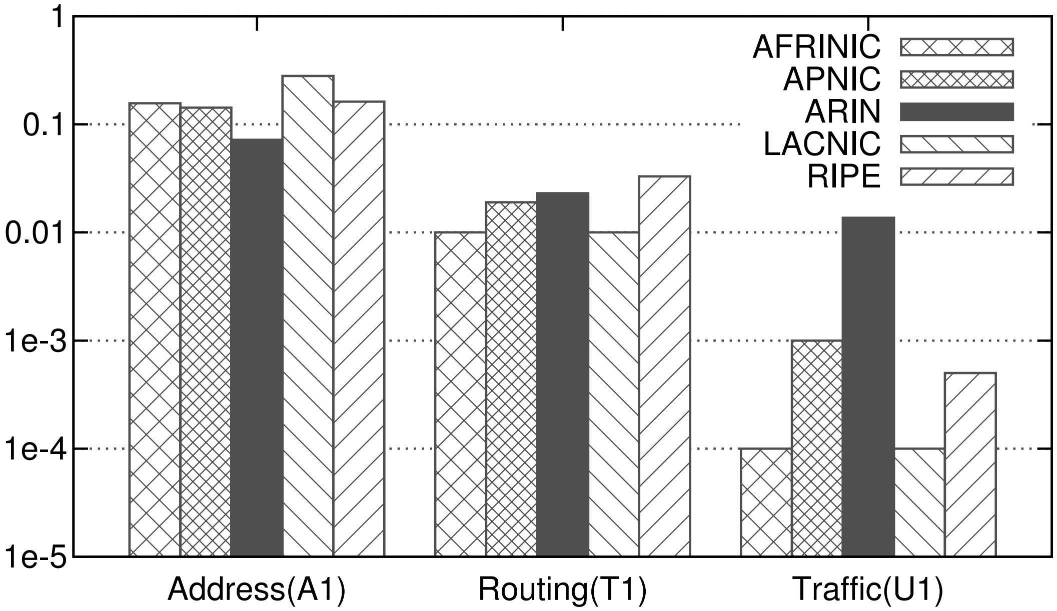


Figure 2.12: IPv6 to IPv4 ratio for three metrics, broken down by region. We see that, not only do different regions have different levels of IPv6 adoption, but that the level of adoption varies by layer; i.e., the relative rank of regions differs across metrics.

is responsible for 21%, and APNIC 18%. These three RIRs represent the most-connected portions of the Internet, and, therefore, it is not surprising that they allocate most of the new prefixes. We also observe that LACNIC and AFRINIC are responsible for 12% and 2% of the allocations, respectively. However, although the well-connected regions dominate the absolute number of prefixes, the *ratio* of IPv6 to IPv4 prefix allocation per region tells a slightly different story. Here, we find that LACNIC has, by far, the largest ratio at 0.280, followed by RIPE at 0.162, AFRINIC at 0.157, APNIC with 0.143, and we see only half as much IPv6 prefix allocation, 0.072, for ARIN. Part of the likely reason is that the ARIN region was an early adopter of IPv4, accumulating many prefixes before resources became constrained.

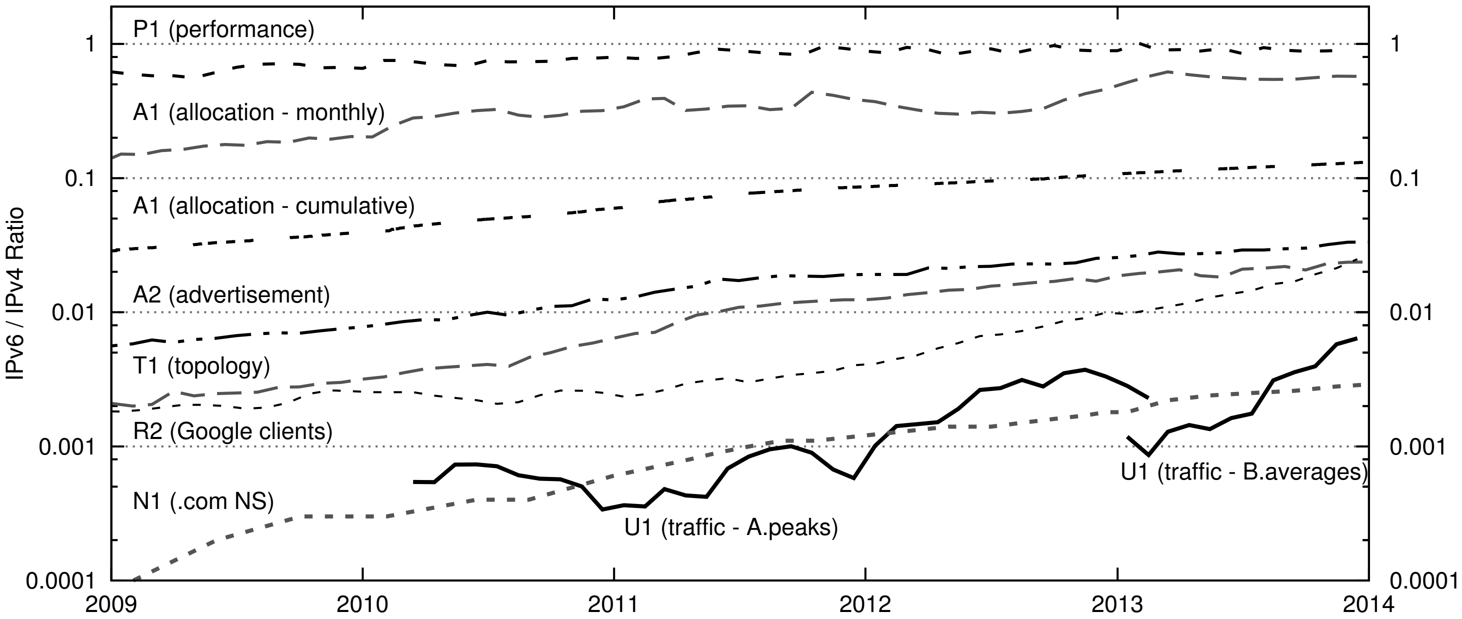


Figure 2.13: The ratio of IPv6 to IPv4 for seven metrics over the last five years, showing adoption level ranges by two orders of magnitude depending on metric, and accelerated growth. For instance, the two traffic lines, $\mathcal{A}$ (peak, through 2013-02) and $\mathcal{B}$ (average, for 2013), show IPv6 traffic has increased over 400% in each of the last two years.

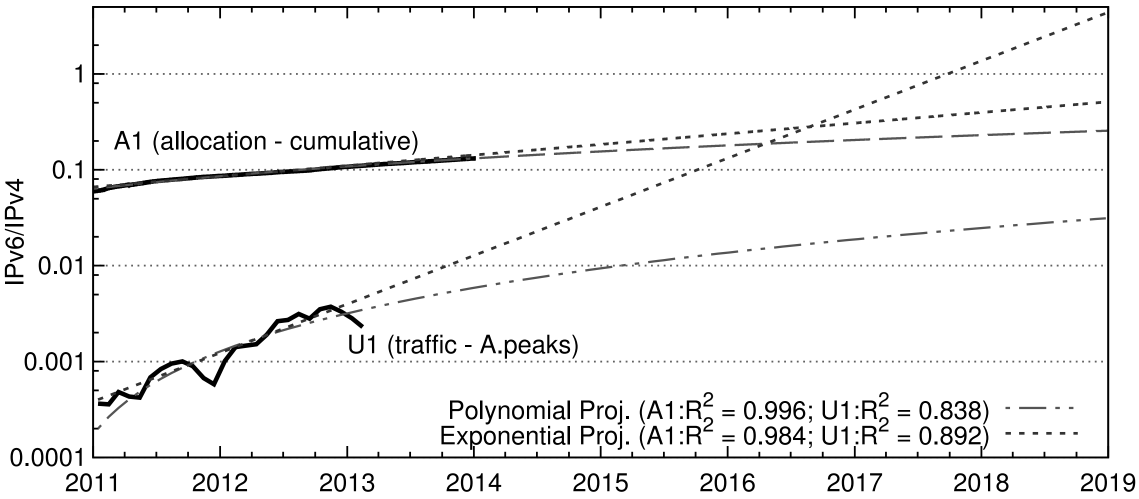


Figure 2.14: Trends for Allocation and Traffic (using the older $\mathcal{A}$ [peak] traffic data), starting with 2011, when IPv4 exhaustion pressure increased, and five-year projections. We caution that trends are volatile and prediction is hard.

In Figure 2.12, we show an analysis of the IPv6 to IPv4 ratio for allocation (A1) as well as two additional metrics whose data allowed region differentiation (T1 [announced AS paths] and U1 [average traffic, $\mathcal{B}$]). Adoption level varies considerably across metrics for the regions (note the log scale), with the highest measured region for each metric at least three times higher than the lowest. We do not show its numbers, but the A2 metric ranks closely match T1, as they both track routing. While we might expect different regions to adopt IPv6 at different rates, surprisingly, we also see that the same ordering of regions does not persist across metrics. For example, the Address Allocation metric shows that ARIN lags behind. However, ARIN performs much better on the other two metrics. First, this affirms our argument that a single metric cannot fairly reflect IPv6 adoption status. More surprisingly, this suggests that, *not only are different regions adopting IPv6 at different rates, but there are different incentives and obstacles (perhaps resource constraints, policy, etc.) that vary the rate for different layers of adoption (i.e., metrics) in any given region.* We intend to explore the cause of these varying pressures in future work.

2.8.2 IPv6 Future

We close by attempting to model how future IPv6 adoption may proceed. As a baseline, for the metrics for which we have four or more years of data, we have already seen that, over that time, nearly every measure of adoption of IPv6 relative to IPv4 has increased by an order of magnitude. In Figure 2.14 we show the IPv6 to IPv4 ratio for A1: Address Allocation (cumulative) and U1: Traffic ($\mathcal{A}$, peak), between 2011 and 2013 or 2014. We start with 2011, as this is when IPv4 exhaustion pressure became more acute and we began to see larger increases in several of our metrics, including traffic. We chose A1 and U1 as they bookend the gamut of adoption metrics, showing the highest and lowest level, respectively. These also represent the first and last step in deployment. We use the older traffic sample, which ends in 2013, instead of the newer, as the former is for a longer period (but is more conservative than the newer, whose rate of increase in 2013 was 433%). The figure also includes projections out to 2019 based on both polynomial and exponential fit functions (R^2 values as shown). Of course, it is possible that upcoming IPv4 exhaustion milestones or other events will lead to discontinuities in the adoption trends. Additionally the growth rate may shift for a number of reasons. Thus, we caution that, while these models represent predictions for where IPv6 adoption may be in five years based on recent trends, the same trends have been volatile in the past and even a small shift could result in much different outcomes. With those caveats in mind, according to the model's projections, *we expect that by 2019 the number of IPv6 prefixes allocated will be about .25–.50 of IPv4, while the IPv6 to IPv4 traffic ratio will be somewhere between .03 and 5.0. In other words, IPv6 appears headed to be a significant fraction of traffic.*

2.9 Summary

In this chapter we report on the analysis of ten global datasets that shed light on twelve aspects of IPv6 adoption. The datasets we analyze comprise both publicly-available data and original data, including the largest traffic dataset reported in an IPv6 study, which resulted from capturing flows for an estimated third to a half of all inter-domain traffic. Our multi-year, multi-perspective, and multi-metric view allows us to draw conclusions about the recent trajectory and current state of global IPv6 deployment.

We find that the level of adoption measured differs substantially depending on the metric applied, suggesting that no single measurement should be relied on in isolation when discussing the state of such a large, federated, and complex system. Second, we also discover that adoption differs across and within regions (again, depending on metric), suggesting that there is room for policy to impact adoption. Finally, we find that both the character and rate of adoption in the two years we measured since 2011 has shifted dramatically, to the point that the capacity for IPv6 use is large and much higher than actual traffic would suggest. Due to this, the field is open for both traffic and end-user capability to grow very rapidly in the coming decade, which is what we are starting to witness based on our latest data.

CHAPTER 3

IPv6 Internet Background Radiation

In the previous chapter we drew a broad picture of the state of IPv6 deployment, showing that the protocol is being adopted rapidly starting in early 2011 and that the nature of its use has shifted toward real, production traffic. Nevertheless, the IPv6 network is still young and we expect that it lacks both the operational maturity of IPv4 as well as the abundance of malicious traffic that the dominant protocol continually faces. We set out to explore routing stability, as a proxy for operational maturity, and to study the level of mailicious traffic via the new protocol. One tool that has previously been successfully employed for studying both of these categories of network events at-scale is the network telescope (e.g., [99,140]).

IPv4 Network telescopes that capture traffic to unreachable destinations have, in the past, been used to observe both security and management phenomena that result as side effects of various events on the Internet, such as scanning activity, backscatter from denial of service (DoS) attacks, as well as misconfiguration and Internet censorship (e.g., [59, 140, 175]). Observations of such Internet background radiation serve to help understand the original phenomenon and aid in raising awareness regarding systemic issues (e.g., bugs, bad practices, or misconfigurations). Because the IPv6 network still lacks maturity, given that the hardware, software stacks, and operator toolkits and skillsets have not yet been hardened by as many years of operation, we expect misconfiguration or mismanagement phenomena to be more common on the new network, while malicious activity may still be low (as the network is a smaller target due to fewer users). Identifying configuration problems early in the adoption process minimizes the cost of fixing them and can inform best-practices for future IPv6 network operators. Moreover, characterizing malicious activity early is similarly of value.

The primary contribution this chapter makes is to report on a large sample of IPv6 background radiation. We conduct the broadest IPv6 network-telescope-based study to date by concurrently announcing BGP prefixes that cover the majority of allocated IPv6 space used by each of the five (RIRs)—86% of allocated /64 networks outside of the transition-only 6to4 block. This complete announcment lasted for several days, after which RIPE, the European RIR, due to privacy concerns, asked us to withdraw a quater of their address space. Their remaining space, along with that of the other four RIRs was advertized for a second period of three months. This perspective and scale allow us to discern spatial and temporal features of the network telescope data, in addition to understanding root causes of the traffic. We also compare a week of our results to a week of similar data we captured using a more traditional IPv4 network telescope.

Our second key contribution is that we show that a *covering prefix* methodology for network telescopes in IPv6 leads to a larger sample of data—and of qualitatively different data—than that afforded by a traditional network telescope based on only unallocated address space. We find that 95% of the packets we captured were from allocated space, arriving at our censor thanks to this methodology, and we show that the captured traffic is clustered close to used prefixes.

3.1 Methodology

In this section, we describe the design of our experiment and our data collection methodology, as well as the mitigating steps and proactive measurements we conducted to ensure a minimal impact of our covering routes. A long-running study by Huston *et al.* demonstrates that it is possible to conduct a safe IPv6 covering prefix experiment [99]. We sought both to replicate and to greatly expand the scope of that experiment with our work.

3.1.1 The BGP Announcements

In early October 2012, we contacted each of the five RIRs to request permission to announce the entire /12 IPv6 address block that had been allocated to them by IANA. After deliberation, each RIR granted us a Letter of Authority (LOA) temporarily allowing us to announce these prefixes via BGP for the duration of our four-month experiment.

Next, we coordinated with AT&T and Hurricane Electric, the upstream IPv6 providers to Merit Network, Inc., our ISP and research partner. This was necessary to ensure that they would accept our announcements of these unusually large blocks. As anticipated, they both needed to execute special configuration changes (i.e. removing sanity filters) to allow us to make such short covering prefix announcements.

After a series of test announcements, on November 8, 2012 we began announcing all five of the covering /12 prefixes for our experiment. We set up a collection server at Merit Network and advertised it as the final destination for each route. All IPv6 traffic on the Internet that was destined for RIR address space—*but not explicitly claimed by another network via a more specific route*—was routed to our collector and archived. The only large, in-use blocks of IPv6 addresses that fall outside the covering prefixes we advertised were 2001::/12 (used by all RIRs in older allocations), 2002::/16 (used for 6to4 transition), and 2003::/18.

On November 13, RIPE NCC asked us to limit our announcements by withdrawing the /12 route and replacing it with a /13 and a /14, the portion of their address space that hasn't been allocated to customers yet. As explored more fully in the following sections, this change gave us a unique control that helped us discern the nature of Internet background radiation in IPv6. During the course of our experiment, we received four inquiries regarding our BGP announcements from the Internet operations community, which were addressed by providing pointers to a detailed study description. With the exception of the RIPE region's change, we did not need to modify our experiment in throughout the four-month experiment.

3.1.2 Routing Visibility

Since our own providers needed policy changes to accept our routes, we believed that policy could play a role in how broadly visible these routes were. To determine the extent to which the BGP announcements were being accepted and propagated across the broader Internet, we analyzed data from both the Route Views [166] and RIPE NCC Routing Information Service [150] BGP archives.

As seen in Table 3.1, our announcements were visible from 8 of the 9 IPv6-capable monitors from the Route Views project, including Australia, Brazil, California, Georgia (USA), Japan, South Africa, Virginia, and the UK. The only Route Views monitor that did not see our routes was KIXP in Kenya. In addition, 9 of the 12 IPv6-capable monitors maintained by RIPE NCC saw our announcement, including Austria, California, Italy, Japan, the Netherlands, New York, Sweden, Switzerland, and the UK. We were partially visible by DE-CIX (Germany), which saw two of the six routes (2600::/12 and 2400::/12). Our routes were not visible from MSK-IX in Russia or PTTMetro-SIP in Brazil.

Examining the Route Views peer perspectives more closely, we found that, on average, of the 93 peers at all sites during period A, 74 peers saw our /12 announcements. Likewise, during period B, 75 of the 98 peers saw our /12 announcements. This compares favorably with the number of peers that saw the average IPv6 prefix known to Route Views (66 and 68, respectively). The smaller /13 and /14 RIPE prefixes during period B were just as highly visible for the first half of the period (through mid-January), but only visible to 5 (of 98) Route Views peers in the second half.

Based on our analysis, we conclude that these announcements were visible at the vast majority of IPv6-capable route monitors.

3.1.3 Validating Data Plane Effects

In general, Internet traffic is routed to the most specific prefix in the BGP routing table; therefore, as our experiment consists of announcing shorter (i.e., less specific) prefixes, we would expect to only capture unclaimed IPv6 traffic. However, due to the immature nature of the IPv6 Internet, we were concerned that the longest-match rule, a core routing

Table 3.1: Visibility in Route Views and RIPE Monitors

Route Server	**LACNIC 2800/12**	**ARIN 2600/12**	**APNIC 2400/12**	**RIPE 2a04/14+ 2a08/13**	**AFRINIC 2c00/12**
Route Views					
r-v.eqix	✓	✓	✓	✓✓	✓
r-v.isc	✓	✓	✓	✓✓	✓
r-v.jinx	✓	✓	✓	✓✓	✓
r-v.linx	✓	✓	✓	✓✓	✓
r-v.kixp					
r-v.saopaulo	✓	✓	✓	✓✓	✓
r-v.sydney	✓	✓	✓	✓✓	✓
r-v.telxatl	✓	✓	✓	✓✓	✓
r-v.wide	✓	✓	✓	✓✓	✓
RIPE RIS					
rrc00	✓	✓	✓	✓✓	✓
rrc01	✓	✓	✓	✓✓	✓
rrc03	✓	✓	✓	✓✓	✓
rrc04	✓	✓	✓	✓✓	✓
rrc05	✓	✓	✓	✓✓	✓
rrc06					
rrc07	✓	✓	✓	✓✓	✓
rrc10	✓	✓	✓	✓✓	✓
rrc11					
rrc12		✓	✓		
rrc13					
rrc14	✓	✓	✓	✓✓	✓
rrc15					

principle, might not be implemented correctly at every IPv6 node. To identify any potential negative effects, we performed a series of short test announcements (as detailed above) before announcing all of the prefixes. We also closely monitored data plane connectivity during these periods and into the first month of the long-term announcement, as discussed next.

In order to validate minimal impact on data plane connectivity, we performed the following: We collected a set of public IPv6 addresses by querying the Alexa top 1M domains [9] for AAAA records. We then categorized these by AS number and covering prefix (i.e., RIR region). We conducted ping tests to the resulting set of twelve thousand hosts, which were spread among diverse ASes and regions, summarized in Table 3.2. *The response rates we saw before and after our announcements were comparable.*

Port Filtering In another approach to validating the representativeness of the captured data, we attempted to detect port blocking between our collection system and sample Internet locations. To this end, we obtained access to a small set of globally-distributed native

Table 3.2: Distribution of the 12,418 IPs used to assess impact of route announcements during testing.

RIR Region	No. of IPs	No. of Unique ASNs
APNIC	1622	603
ARIN	1219	530
LACNIC	159	62
RIPE	9,409	3,654
AFRINIC	9	8
Total	12,418	4,857

IPv6 hosts. These hosts were located in Atlanta (AS3595), Japan (AS2516), South Africa (AS33764), Tanzania (AS37084), and the United Kingdom (AS15830). From these hosts, we were able to actively probe our network telescope and examine the data coming into the collection infrastructure. It should be noted that the server in the United Kingdom was unable to have packets routed to the our blocks. However, this appears to be a filtering policy of only the upstream ISP (TelecityGroup Limited), as all of our routes were observed at the London Internet Exchange (LINX, RIS server `rrc01` in Table 3.1), and AS15830 is a member of LINX.

We began a series of probes of the network telescope prefixes from the remaining four servers. We scanned arbitrarily chosen addresses within the otherwise-unrouted portions of our announcements, such that the scanning packets would be routed to us. We then separated these synthetic packets from organic traffic received by the collector, and discard the scan packets we generated for the rest of the analyses presented here. Upon aggregating the probes, we found *no port-based packet filtering in TCP or UDP* between our four hosts and our collector at Merit Network (AS237). This lies in stark contrast to the experiments conducted by Kreibich, *et al.*, which illustrated widespread port blocking by ISPs for IPv4, as high as 50% for the most commonly blocked ports in IPv4: NetBIOS (TCP/139), SMB (TCP/445), and RPC (TCP/135) [114].

Table 3.3: Packet counts, rates, and protocol breakdown in the complete datasets, by RIR.

RIR	**Dataset A.Complete (24 hr.; with RIPE /12)** **2012-11-12T17:00:00 to 2012-11-13T16:59:59**							**Dataset B.Complete (3 mon.; with RIPE /13 & /14 only)** **2012-12-01T00:00:00 to 2013-02-28T23:59:59**						
	kPackets	**Average (peak)**		**% Protocol Type**				**kPackets**	**Average (peak)**		**% Protocol Type**			
		pkts/sec	**kbits/sec**	**UDP**	**TCP**	**ICMP**	**Other**		**pkts/sec**	**kbits/sec**	**UDP**	**TCP**	**ICMP**	**Other**
APNIC	11,826	85 (149)	310 (552)	48.5	4.8	46.1	0.6	1,345,425	172 (811)	523 (1868)	45.5	20.9	33.2	0.4
ARIN	12,146	88 (517)	299 (1169)	72.7	3.9	23.0	0.4	2,481,476	318 (21185)	418 (28685)	23.3	67.1	9.1	0.5
LACNIC	8,936	69 (130)	210 (370)	50.4	2.1	46.4	1.1	504,848	65 (875)	238 (768)	28.8	4.6	65.6	1.0
RIPE	25,490	175 (8750)	235 (6211)	20.6	41.0	38.4	< 0.1	3	< 0.1	< 0.1	8.3	27.4	64.2	0.1
AFRINIC	298	2 (9)	4 (16)	56.7	7.9	35.2	0.2	20,290	3 (88)	4 (114)	54.4	6.5	38.8	0.3
Overall	58,696	419 pps	1.06 Mbps	41.7	19.9	38.0	0.4	4,352,040	558 pps	1.18 Mbps	30.9	45.4	23.3	0.4

3.1.4 Complications of a Covering Prefix

One interesting difference that emerges in the study of network telescopes that are based on a covering prefix versus traditional ones that are based on unused address space is the impact of routing instability. In a network telescope based on completely *unused* address space, routing instability has no impact, as there are no prefix allocations within the telescope. In the case of a covering prefix, however, the network telescope is effectively the union of all address space (under that prefix) not being otherwise advertised by BGP at any given point in time—whether it is allocated to networks or not. This results in a much more complete telescope but also in complications when the allocated address blocks exhibit instability. In such scenarios, address space can rapidly shift between the network telescope and an allocated and announced address block. As a result, traffic captured due to such instability is much more likely to be composed of otherwise *normal network packets* that happen to be caught by the network telescope due to a (perhaps brief) drop of the more specific route. As such, care must be taken in interpreting and comparing results from network telescopes that do versus ones that do not use a covering prefix announcement. We discuss how we categorized our data with this in mind in Section 3.1.5.

Benefits of a Covering Prefix During the current transition towards greater adoption of IPv6-enabled networks and services, it is critical to understand any routing instability, as it can help to identify and address the misconfigurations, bad practices, or bugs in software or hardware that are the root cause.

In addition to the ability to observe instability, the main advantage of a covering-prefix-based network telescope study is *much better visibility* due to the known clustering of Internet background radiation near active network prefixes. For instance, Bailey, *et al.* and Cooke, *et al.* have shown that, with an IPv4 network telescope, much more data is gathered when it is located near live hosts [17, 40]. Likewise, Harrop *et al.* discuss, in the context of enterprise IPv4 networks, the advantages of *greynets*, which have unused addresses interspersed with live addresses to produce visibility similar to that of a large, contiguous network telescope [86]. Since the IPv6 address space is vast, compared to IPv4, (and, consequently, sparse) this locality advantage is even more necessary to capture a network telescope sample of any meaningful size. Our results bear this out.

As mentioned above, we initially announced each of the five /12 prefixes that have been assigned by IANA to the RIRs. However, after several days RIPE requested that we reduce our 2a00::/12 announcement to 2a04::/14 and 2a08::/13. Although our reduced RIPE announcement still covered 75% of the initial RIPE address space under the /12, the volume of traffic decreased disproportionately from around 300–900 kilobits per second to 0–80 bits per second for RIPE. On most days, no packets arrived for these RIPE prefixes and we collected an aggregate of only 2,635 RIPE packets over the entire course of the three-month period they were announced[1]. These results agree with previous work in IPv4 [40] and in IPv6 [65, 99], showing more packets near used space.

3.1.5 Data Categorization

We sought to segregate the packets we captured according to whether they would be seen by a traditional network telescope methodology or only by a covering prefix, as we suspected differences. This way, we could characterize traditional Internet background radiation traffic separately from what is otherwise potentially valid traffic to instantaneously unreachable destinations that are normally routed. Likewise, focusing on the traditionally collected traffic would allow an apples-to-apples comparison of this IPv6 data with other IPv4 background radiation studies.

We started by aggregating all of the routed BGP prefixes seen by all Route Views monitors. We combined initial RIBs on the first hour of each experiment period with every subsequent update file (a digest of BGP packets) from all Route Views monitors for each dataset. Since we aimed for a *conservative filtering to produce what we considered traditional background radiation data*, we recorded all prefixes ever announced in any BGP message throughout the entire data collection period, no matter how briefly or sparsely. We excluded from consideration prefixes *shorter* than our own announcements (e.g., a sparse announcement of 2000::/3), as those would not affect our packets.

[1] While the traffic dropped by three orders of magnitude right away, as mentioned in section 3.1.2, the two smaller prefixes did have reduced global visibility during the second half of the three-month period as well, which further reduced collected data.

Next, we aggregated the five RIRs' allocation data and built a list of all prefixes which were allocated by the RIRs prior to the end of each of our two data collection periods. The resulting list in each period, along with the routed prefix list discussed above, constituted our *"allocated"* and *"routed"* filters for that period, respectively. The Cartesian product of these two binary filters gives us four categories of packets.

The type of packets that network telescopes have traditionally captured are the ”unallocated and unrouted” category. These are packets destined to prefixes that have not been assigned to any organization by an RIR and which are not normally advertised in the global BGP table—not until the operators of the network telescope announce them. Packets in this category are what have typically been studied when examining background radiation in IPv4 (e.g., the work of Wustrow, *et al.* [175]). Thus, these packets serve as the core data that we use in our characterization of background radiation in IPv6, as discussed in section 3.2. For succinctness, we term this traffic "dark" in the sections that follow, but we use this term interchangeably with "background radiation". We discuss the other three categories of packets in detail in section 3.3.

Table 3.4: Categorization of packets by destination in each dataset. "Allocated" are packets with destinations matching a prefix that was allocated by an RIR; "routed" match a BGP prefix known to Route Views any time during collection. Unique number of destinations as well as total TCP payload in each category is also shown.

Category	**Dataset A**				**Dataset B**			
	Packets	**%**	**Unique Dest.**	**TCP Bytes**	**Packets**	**%**	**Unique Dest.**	**TCP Bytes**
Unallocated, Unrouted ("dark")	2,997,540	5.11	36,855	21,880,029	208,988,570	4.80	1,274,798	2,901,567,271
Unallocated, Routed (UR)	456	0.001	28	0	13,948	< 0.01	848	39,216
Allocated, Unrouted (AU)	35,426,005	60.36	4,784	239,960	2,576,456,636	59.20	85,956	18,857,335
Allocated, Routed (AR)	20,271,633	34.54	22,335	36,435,501	1,566,581,006	36.00	1,580,052	424,971,873
Total (Complete)	58,695,634		64,002	58,555,490	4,352,040,160		2,941,654	3,345,435,695

Table 3.5: Breakdown of address space under our covering prefixes at the end of February 2013, showing percentage allocated and routed. Also shown is size (in /24 subnets) of non-covering IPv4 network telescope prefix we compare some results to. Note that the size of the entire IPv4 address space is 2^{32}, which the square root of the size of a single IPv6 /64 subnet.

RIR	Prefix	Size of Space in /64 Subnets	Alloc.	Routed
APNIC	2400::/12	$2^{52} = 4,503,599,627,370,496$	3.29%	1.31%
ARIN	2600::/12	$2^{52} = 4,503,599,627,370,496$	1.85%	0.20%
LACNIC	2800::/12	$2^{52} = 4,503,599,627,370,496$	6.75%	0.46%
RIPE NCC	2a00::/12	$2^{52} = 4,503,599,627,370,496$	2.66%	2.15%
RIPE NCC	2a08::/13	$2^{51} = 2,251,799,813,685,248$	0.00%	0.00%
RIPE NCC	2a04::/14	$2^{50} = 1,125,899,906,842,624$	0.25%	0.04%
AFRINIC	2c00::/12	$2^{52} = 4,503,599,627,370,496$	0.43%	0.41%
IPv4	35≈/8	A total of 54,784 /24 subnets	None	None

3.1.6 Dataset Description

Table 3.3 summarizes the complete datasets that we captured. We present data from two periods: dataset A (a 24-hour-long capture starting on 12 November 2012), and dataset B (a three-month-long capture starting on 1 December 2012). An outage caused by undetected power failure occurred between 5-9 January 2013; we have no packets from that period.

The covering prefixes we announced for each RIR subsumed varying amounts of address space. Table 3.5 shows the percentage of space under each announced covering prefix that was allocated at any time prior to the last day of the three-month (B) period and that was routed, even briefly, at any time during the period.

Packet and Dataset Categories Table 3.4 shows the breakdown of the two datasets according to the categories explained in section 3.1.5. For each category, we also include the number of unique destinations, which gives a sense of the spatial nature of each target address set, as well as TCP payload bytes. The "unallocated and unrouted" subset is the background radiation data that traditional (non-covering prefix) network telescopes have captured.

As Table 3.4 also shows, due to the covering prefix nature of our route advertisements, we were able to capture a broader spectrum of invalid traffic than would be possible via the traditional network telescope approach. In fact, 95% of the packets we captured were due to our use of a covering prefix. That traffic falls into three categories: unallocated/routed, allocated/unrouted, and allocated/routed. As this traffic is distinct from Internet background radiation as previously studied, we treat it separately and aim to explain and characterize each category in section 3.3.

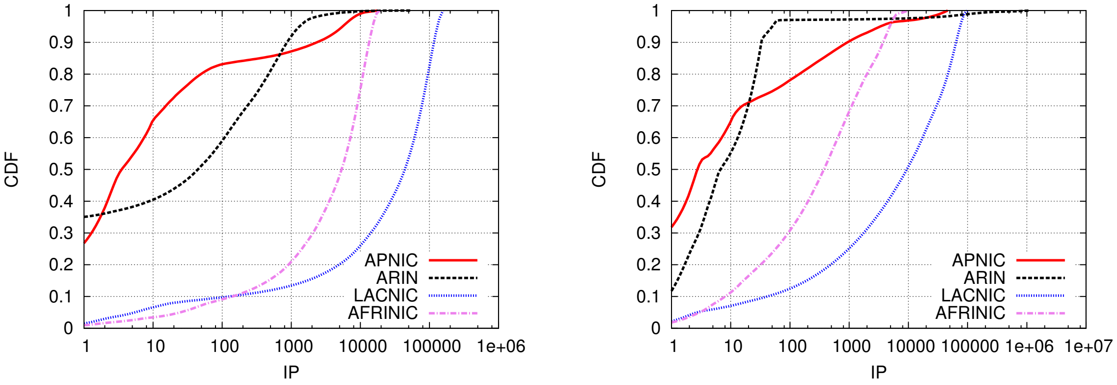
Basic Statistics Before we move to a deep analysis of the "dark" subset of our data, we first provide basic high-level statistics about the overall (unfiltered) captured packets. Table 3.3 shows the volume of all packets per RIR collected during the longer (three-month-long) dataset B time frame (i.e., B.Complete).

An analysis of packet TTL values reveals that the vast majority of received packets (90% captured during time period A and 97% during B) appear to be sent by Windows operating systems (TTL between 64 and 128).

During the 24-hour period selected to be dataset A we received 18.7 GiB in 59M packets. The APNIC, ARIN, LACNIC, and RIPE NCC blocks all received a similar order of magnitude of data, while AFRINIC received two orders less. A similar distribution was observed over the three months that constitute dataset B, except for the loss of nearly all data from RIPE due to withdrawal of 2a00::/12 in favor of two smaller RIPE prefixes. We collected 1,141 GiB via 4.4Bn packets in dataset B. As shown in Table 3.3, we received an aggregate rate of nearly 1.1 Mbps of traffic during period A and nearly 1.2 Mbps during period B. Since RIPE data in B was nearly zero, we refrain from separately analyzing RIPE in dataset B in the sections that follow, except where noted.

Protocols As Table 3.3 also shows, the protocol distribution in the overall data is dominated by no single protocol but is made up of between 20 and 50 percent of each of TCP, UDP, and ICMP, though TCP (with 45% of packets) ranks higher than the other protocols in the complete three-month dataset while UDP is the most common protocol in in the unfiltered 24-hour dataset (42%).

Transition Addresses One question of importance when studying the state of IPv6 deployment is the relative proportion of native IPv6 hosts versus those using IPv6 transition technologies. We took advantage of the large and global nature of our collection to explore



(a) **(b)**

Figure 3.1: Cumulative distribution of source (3.1a) and destination (3.1b) IP addresses in B.Dark, sorted by number of collected packets

the distribution of source addresses that were used for 6to4 and Teredo, two of the most prevalent IPv6 transition technologies. We found that, of the 12.5M unique sources observed in dataset B, 12.4% were using 6to4 prefixes. Teredo was used by another 21.7%. Thus, over 34% of sources seen were transition addresses. Note that, since 53% of sources fell under a single native /36 prefix, removing that outlier would yield 72% transition sources. This is approaching the higher 96% seen in an ad-based experiment conducted by Karir *et al.* [110]. We caution, however, that since our sample is largely based on traffic intended for networks that are misconfigured or unstable, it may not accurately reflect the overall IPv6 client population.

3.2 Background Radiation Results

3.2.1 Background Radiation Data Description

As our first aim is to characterize IPv6 background radiation traffic and compare it to IPv4 background radiation, the core statistics we report in this section focus on the unallocated/unrouted ("dark") category of data. We leave explorations of the other three categories of collected packets to the sections that follow.

3.2.2 Spatial Analysis

In this section, we provide a comparative analysis of spatial features of the dark traffic we observed.

Traffic Volumes and Patterns For the three-month-long background radiation dataset, B.Dark, we observed a total of just 209m packets across all five prefixes. ARIN's traffic dominated the packet count with about 205m packets. We collected about 1m packets in the APNIC dataset. The LACNIC dataset contains 3m packets, and both the AFRINIC and RIPE dataset contain a relatively low number of packets (325k and 2.5k respectively). We refrain from including RIPE in several of the following analyses, due to the negligible number of packets captured after reducing the announcement, as discussed in Section 3.1.4.

Source Distributions Figure 3.1a shows the cumulative distribution of source addresses sorted by packet contribution in B.Dark. In both the ARIN and APNIC datasets, we see a few source addresses account for a significant portion of the traffic. In particular, one source address contributes around 30% of the packets, and 90% of these two regions' traffic is accounted for by 1–2k IPs. The other two regions (AFRINIC and LACNIC) both have higher source address diversity, and it takes more than 10k and 100k unique sources to account for 90% of the packets, respectively.

Destination Distributions Figure 3.1b shows the distribution of destination addresses in B.Dark. Here, we see a slightly greater concentration of destinations and most prominently so for ARIN, where fewer than 30 destination IPs account for 90% of the traffic. In the other three RIRs' datasets, fewer than 10k destinations make up 95% of the packets seen. LACNIC has the most diversity again, with more than 30k unique destinations needed to account for 90% of packets.

3.2.3 Protocol and Port Analysis

Protocols Figure 3.2a shows the protocol breakdown in the darknet datasets. Just as observed in the discussion of the overall data in Section 3.1.6, the protocol volumes in the three-month dataset (B.Dark) are heavily biased toward ICMP. However, in the APNIC dataset, we find that TCP dominates the traffic, at 72%, with another 22% of observed packets being ICMP.

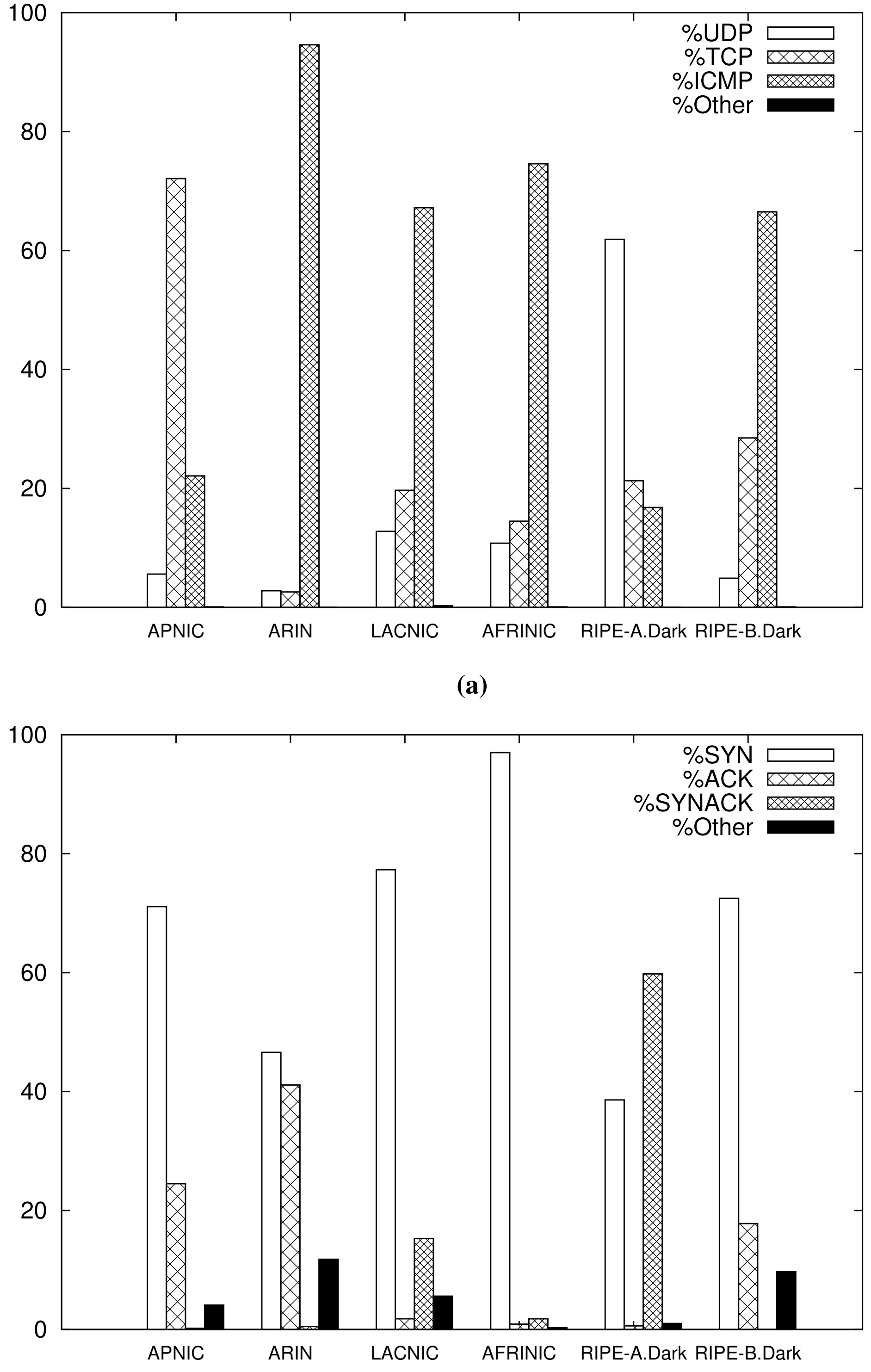
In the one-day dataset with RIPE's network blocks, we see 62% UDP traffic, with TCP and ICMP both contributing about 20%. A surprising 95% of all observed packets in the ARIN dataset were ICMP. We investigate this further in section 3.2.6.1.

These numbers lie in contrast to typical IPv4 network telescope measurements (e.g., [175]), where TCP dominates, UDP contributes about half as much as TCP, and ICMP is present with only single digits of percentage points of volume. Our own comparison IPv4 analysis, described in Section 3.2.5, also bears this out, as we see TCP constituting 82% of packets.

TCP/UDP Ports Table 3.6 presents the top 5 TCP source and destination ports observed in Dataset B.Dark. Source ports 80 and 443 appear in the top 10 or top 20 ports for all datasets. Source port 53 is also the second-most-common port in the ARIN dataset. These suggest that much of the collected traffic is likely to be misdirected responses from DNS and web services. A high number of TCP port 7 packets, used by the Echo protocol to measure round trip times, suggests possible network testing traffic.

Similarly, Table 3.7 shows the top 5 UDP ports observed in our our different datasets. Port 53, in both source and destination fields, clearly dominates the UDP traffic. While only 6% of UDP packets have source port 53, 28% of the bytes received in UDP packets were in packets sent on port 53. Additionally, nearly 85% of received UDP packets were destined for port 53. These constitute DNS responses and queries, respectively. Another notable pattern in UDP is the significant prevalence of traffic both to and from port 123, which is primarily used for Network Time Protocol packets [131]. Further analysis of some of these features is explored as separate case studies in section 3.2.6.

TCP Flags The most interesting flag combinations for network telescope research are SYN and SYN+ACK. The former indicates packets where a connection is being attempted (ultimately unsuccessfully to an unreachable address) such as when a host is scanning ports or IPs to connect to. SYN+ACK indicates the response to a previous connection attempt,



(a)

(b)

Figure 3.2: Breakdown of Protocols (3.2a) and TCP Flags (3.2b) by RIR (from B.Dark unless otherwise noted)

Table 3.6: Top 5 TCP Ports by Packets for Each RIR and Overall in Dataset B.Dark and for RIPE in A.Dark

APNIC		ARIN		LACNIC		AFRINIC		B.Dark Overall, %of TCP Pkts.			
src	dest	src	dest	src	dest	src	dest	src	%	dst	%
56583	80	22	7	445	80	993	39236	22	30.32	7	32.40
49561	443	53	22	135	45682	443	45682	56583	1.59	80	14.44
49559	2001	51211	80	12829	56024	143	24739	445	1.22	22	10.13
49558	445	51208	34521	80	61638	80	26823	49561	0.59	443	1.11
49560	5222	51207	443	49155	29671	5222	52232	53	0.49	34571	0.85

RIPE (A.Dark)			
src	%	dst	%
80	35.62	179	31.10
443	20.45	25	2.18
993	2.73	53	1.03
49166	0.85	80	0.92
5228	0.80	40000	0.91

Table 3.7: Top 5 UDP Ports by Packets for Each RIR and Overall in Dataset B.Dark and for RIPE in A.Dark

APNIC		ARIN		LACNIC		AFRINIC		B.Dark Overall, %of UDP Pkts.			
src	dest	src	dest	src	dest	src	dest	src	%	dst	%
53	53	53	53	12929	53	53	39236	53	5.96	53	84.81
16703	16703	123	123	53	45682	45682	45682	123	1.20	123	1.19
45682	39045	33336	30718	45682	3702	12829	24739	12829	0.61	30718	0.71
12407	37385	54709	33336	123	123	48359	26823	33336	0.31	33336	0.31
54593	45682	500	500	18600	29671	20904	37648	45682	0.14	45682	0.12

RIPE (A.Dark)			
src	%	dst	%
53	10.50	53	88.60
32833	0.07	389	0.50
12589	0.05	40000	0.18
500	0.04	16881	0.04
123	0.03	500	0.04

such as might be seen when an attempt to make a connection is made using a source address that is within the network telescope, and the contacted host replies. When this is done for malicious reasons, it is called backscatter and is often observed during certain classes of DDoS attacks, but this can also be observed in cases of misconfiguration.

Figure 3.2b illustrates the regional differences in the TCP flag combinations in the data we collected. The relatively low volume of SYNACK packets (1.7% in B.Dark) indicates that spoofed-random-source DDoS attacks do not appear to be prevalent in IPv6. However, while low as a percentage of all dark traffic, SYNACKs do constitute a large percentage of the single-day RIPE dataset (59.8%). Our collected TCP packets are overwhelmingly SYN packets; this is most apparent in the AFRINIC dataset, 97% of which are SYN packets. As SYN packets form a plurality, if not the majority, of the TCP packets we see (40.5% in

A.Dark and 52.6% in B.Dark), we can infer that a majority of dark TCP traffic coming to the network telescope is connection attempts to unreachable networks.

3.2.4 Temporal Analysis

Overall, we see no clear trend in the volume of IPv6 background radiation over the three-month period of our study. As Figure 3.3 shows, however, there is a slight decrease in total traffic volume observed by the network telescope in the last two weeks of the B dataset. This is primarily due to a drop in traffic volume to the ARIN region from approximately 30 packets per second to 10. The decreased traffic is related to a reduction in ICMP probing of a small set of ARIN destinations that are heavy recipients of ICMP traffic. We discuss this traffic in section 3.2.6.1.

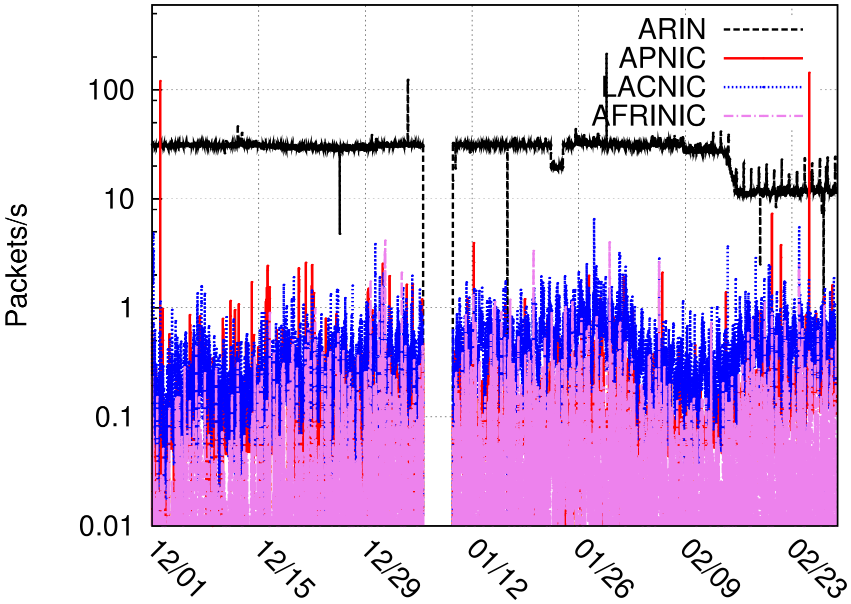


Figure 3.3: Background radiation (i.e., dark) packet rate for the four /12s prefixes we announced over three months (dataset B.Dark). The ARIN prefix received about 30 packets per second, whereas the other regions all saw only between 0.1 and 1.0 pps. The hole in January was caused by a power outage.

Table 3.8: IPv4 and IPv6 protocol breakdown for traffic during the week of February 4, 2013 in the aggregate B.Dark IPv6 dataset and in the IPv4 35.x.y.z network telescope.

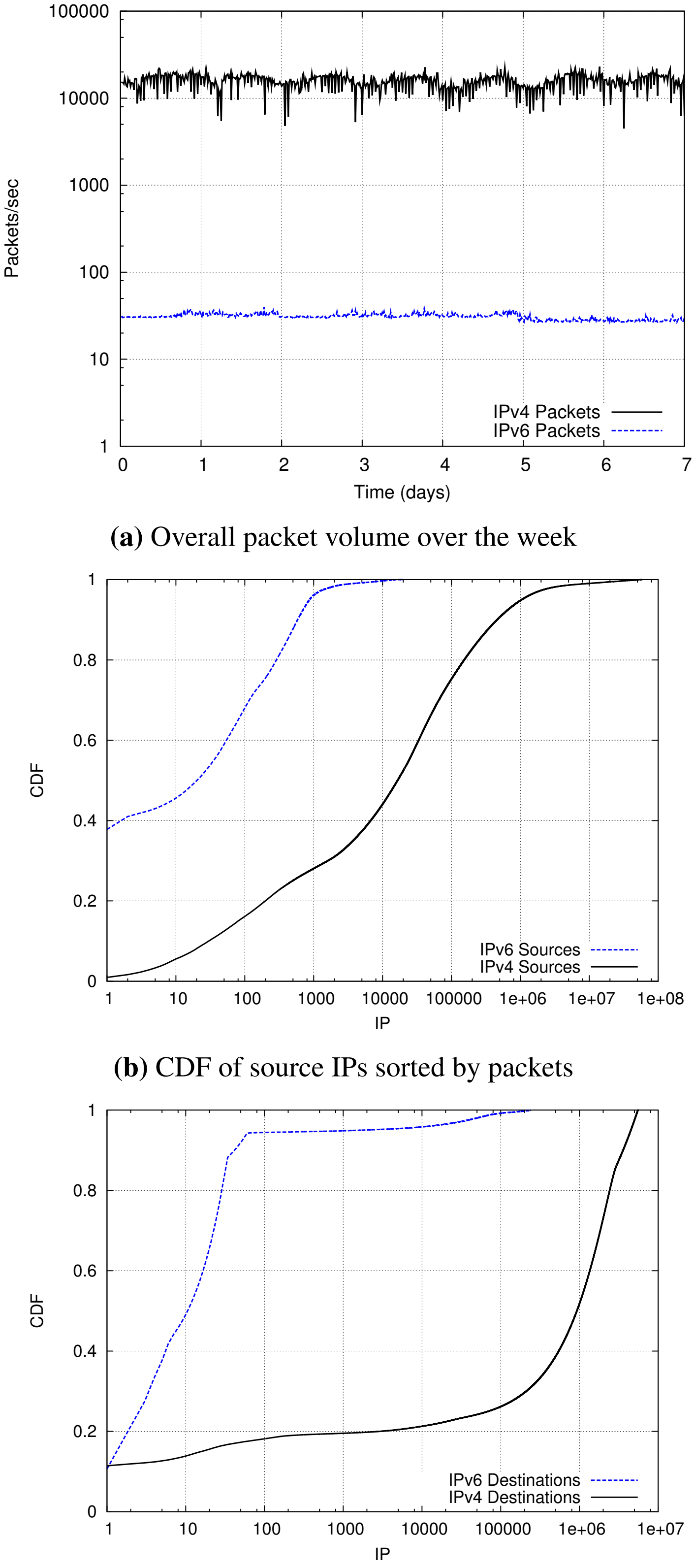
Darknet Packets	% TCP	% UDP	% ICMP	% Other
IPv4 (35.x.y.z)	81.7	15.8	2.3	0.2
IPv6 (All /12)	3.3	2.9	93.8	< 0.1

3.2.5 Comparison to IPv4

Our work follows a series of seminal IPv4 network telescope experiments that characterized the background radiation of unallocated address space in IPv4, as well as its evolution over time (e.g., [140, 175]). Our IPv6 network telescope results suggest several important differences (and some similarities) compared to that body of work. To produce a more recent and valid comparison, we analyzed a single week of IPv4 background radiation captured during the course of our ongoing IPv6 packet capture.

The methodology used to capture this data is identical to that described in Section 3.1.1, with the main differences, aside from the protocol, being that (a) the IPv4 space we monitor is completely *unallocated and unrouted* (by our definition) and thus not a covering prefix; and (b) the size of the address space is considerably smaller: our IPv4 network telescope is composed of address blocks encompassing 13,915,136 total host addresses (≈55k typical subnets). This is equivalent to about 84% of an IPv4 /8, far smaller than our announced IPv6 address space (see Table 3.5). We subset the three-month background radiation IPv6 data (B.Dark), focusing on the same week (beginning 4 February 2013) for both it and the IPv4 data. We next highlight important differences between the two protocols' background radiation.

Traffic Volume The first striking difference between the two background radiation samples is the relative volume of traffic collected. The IPv6 telescope, as Figure 3.4a shows, collected an aggregate packet rate of approximately 30 packets per second. In stark contrast, the IPv4 telescope sustained a rate of around 15,000 pps, a 500-fold difference. The bitrate difference (not shown) is similarly unmistakeable. This disparity is expected, due to the still-low adoption of IPv6 and the dearth of malicious traffic over the new protocol.



(a) Overall packet volume over the week

(b) CDF of source IPs sorted by packets

(c) CDF of destination IPs sorted by packets

Figure 3.4: Comparison of IPv4 and IPv6 background radiation during the week starting on 4 February 2013.

Even the rate of total traffic we received (i.e., with all categories of packets counted), is only about 20-fold higher, at 558 pps. This is in spite of the vastly larger address space covered by our IPv6 announcements as compared to the /8 in IPv4 (as discussed in the previous section).

Spatial Analysis Examining the spatial distributions of the sources and destinations in the two samples, shown in Figures 3.4b and 3.4c, respectively, reveals several differences. In the source distributions, we see the IPv4 source contribution ramp-up is slow, taking more than 100 top sources to account for 20% of packets. In the IPv6 data (top line), however, a single source contributes nearly 40% of the packets; we elaborate on that source in Section 3.2.6.2. The two distributions have similar shape except that the IPv6 line is shifted to the left, as we would expect based on the lower volume of IPv6 traffic, and up, due to the single outlier.

The distributions of the destination IPs, however, suggest a qualitative difference between the two protocols. Namely, the IPv6 sample has many more heavily-hit IPs relative to the total set of destinations, while IPv4 has a small number of heavy hitters. The vast majority of IPv4 destinations only see a relatively low number of packets—e.g., 16.3% of destinations account for just 50% of the IPv4 packets. On the other hand, less than 0.01% (10) of the IPv6 destinations account for 50% of the traffic. We recall that the size of the entire destination address space monitored by the IPv4 network telescope is on the order of 2^{24} (nearly 14M unique destinations), whereas the address space of the IPv6 telescope—detailed in Table 3.5—is on the order of 2^{118} hosts, and only a very small percent of routed space generally outside of our visibility. Thus, the fact that so few hosts contacted the IPv6 space we monitored (and that so few IPs are contacted within the space) is itself a significant difference from IPv4.

Protocol Differences In Table 3.8 we see the protocol breakdown of the IPv4 background radiation compared to that of the aggregated IPv6 B.Dark data for the comparison week. As can be seen, while TCP is the most prevalent protocol in the IPv4 sample (82%), in the IPv6 sample the dominating protocol is, by far, ICMPv6, making up 94% of the packets

Table 3.9: Top 10 TCP Ports with Percentages in the IPv4 Sample and Comparative Ranking in IPv6 Dark Dataset (if Present) for the Week of 2013-02-04.

Source Port	%	v6 Rank	Dest. Port	%	v6 Rank
80	10.07	#47	445	47.75	#4,268
6000	2.83	#27,826	12350	7.14	
30800	0.73		23	4.97	
0	0.70	#5,914	80	4.15	#710
22	0.67	#1	443	4.03	#685
38121	0.54	#20,825	3389	3.91	
4935	0.51		22	2.44	#442
7777	0.49	#25,981	1433	1.68	
6005	0.38		3072	1.03	
21	0.36		1024	1.03	

in the B.dark dataset. This suggests that *there is relatively less pollution traffic in the IPv6 dark space from scanning and backscatter and more from probing, diagnostic, and management traffic.* However, as discussed more in Section 3.2.6.1, a small block of heavily-hit destinations is responsible for the vast majority of ICMP packets in the dark IPv6 dataset; these inevitably skew these results. That said, as IPv6 is at an early stage of deployment, we certainly expect the properties of its traffic to differ from IPv4.

Table 3.10: Top 10 UDP Ports with Percentages in the IPv4 Sample and Comparative Ranking in IPv6 Dark Dataset (if Present) for the Week of 2013-02-04.

Source Port	%	v6 Rank	Dest. Port	%	v6 Rank
19288	1.60	#51,473	10320	35.88	#31,731
39776	1.55	#18,816	5060	4.85	#61,409
17148	1.55	#39,266	47458	2.99	#46,486
58843	1.54	#12,809	53	1.99	#1
17190	1.54	#43,881	137	1.39	
10688	1.52	#48,799	3544	1.24	#37,930
18864	1.51	#1,315	39455	1.17	#2,117
24048	1.48	#25,676	65535	0.58	#62,982
8090	1.48	#60,075	1900	0.44	#58,668
10042	1.42	#59,877	161	0.35	

Port Distribution Tables 3.9 and 3.10 show the top ten source and destination ports for each of TCP and UDP, respectively, in IPv4. Next to each port is that port's placement in the corresponding rankings from the IPv6 data during the same week. We do not separately show all the top ports for this week of IPv6 data, as they closely match the top ports seen in the overall dark data (shown in Tables 3.6 and 3.7). Looking at the top IPv4 ports, we first observe that, unlike in the IPv6 data where UDP port 53 dominates both directions (78% of destination and 13% of source ports during this comparison week), it is not among the top

IPv4 source ports (though it is the fourth destination port). In fact, it ranks as only the 192nd most common source port, accounting for just 0.02% of IPv4 UDP packets. This suggests that relatively little stray DNS traffic is entering the IPv4 network telescope, while this type of traffic is common in our IPv6 data. Indeed, both the prevalence of DNS and ICMP traffic in IPv6 background radiation are congruent with previous findings (e.g., [111]) for overall IPv6 usage, which reported high fractions of these two types of traffic via the new protocol. Other than port 53, we find no coexistent ports in the top 10 lists for UDP in the two samples. For TCP, we also see little similarity, with just port 22 in both protocols' top source port lists and 22 and 80 in both top destination port lists.

Significant absences are TCP/23 (Telnet) and UDP/137 (NetBIOS) from the IPv6 dataset from that week. Despite our earlier discussion of the lack of any port filtering in IPv6 in Section 3.1.3 and the prevalence of these both in IPv4, we saw zero Telnet packets and only a single NetBIOS packet in that week's IPv6 data. In general, there is a markedly different overall distribution of ports between the two datasets, supporting the hypothesis that *there is a qualitative difference in the nature of traffic in IPv4 and IPv6 background radiation—it is not just a matter of volume*.

TCP Flags Lastly, we compare the proportions of TCP flags in the IPv4 dataset to that in IPv6. As discussed earlier in section 3.2.3, the most interesting flag combinations for Internet background radiation research are SYN and SYN+ACK. We tabulated SYN and SYN+ACK packets in the IPv4 dataset and found the two categories to make up 83% and 14%, respectively. The IPv6 percentages in this dataset for these two flag combinations are 25% SYN, and 1% SYN+ACK. These differences suggest that *scanning (which manifests itself as SYN packets to dark space) and backscatter (which manifests as SYN+ACK packets to dark space) are both less prevalent in IPv6 background radiation than in IPv4.*

3.2.6 Darknet Case Studies

3.2.6.1 ICMPv6 Probing/Scanning

As described earlier, we see a significant amount of ICMPv6 traffic in our background radiation dataset. Of particular interest to us was whether we would observe signs of large-scale scanning. We see clear evidence of sequential scanning in a handful of cases, though it was generally limited to smaller subnets rather than randomized scans of the entire address space.

Focusing on just the background radiation subset of the data, we find 16 APNIC, 1,646 ARIN, 9 LACNIC, and 3 AFRINIC addresses sourcing over 1,000 ICMP packets in B.Dark. Of these, we find 249 addresses in ARIN with over 100k ICMP packets and five with over a million. Of those five, one begins with fe80:: (discussed in Section 3.2.6.2), while two are in Akamai Technologies address space.

An interesting specific example is the single ARIN Region IP address 2600:140f:b::b81a:a21f, which is also from address space belonging to Akamai. It generated 2.5M total ICMPv6 packets to 141 unique destinations. Other IP addresses from the Akamai address space sent similarly large amounts of ICMPv6 traffic that was received by our network telescope. In total, we observed over 17M ICMPv6 packets originated by Akamai in B.Dark. We speculate that Akamai probes hosts that provide IPv6 content when making redirection or other content-update decisions.

In one extreme case, we saw fe80::224:38ff:fe7e:af00, a single, link-local ([90]) IP address, send over 71M ICMPv6 packets to only 27 unique destinations, all within the same /120 prefix (i.e., with only the last 8 destination bits varying) under 2607:fc86::/32. A similar number of packets (around 2.6M) were sent to each address, suggesting a repeated automated scan of this small set of destinations. While we do not know the source of these particular packets, it is possible that the misconfigured host was attempting to monitor the availability of a small group of devices, except that the address range of those devices is in neither publicly advertised space nor allocated space. Interestingly, this is not the only source address probing those same destination addresses.

Upon deeper inspection, we find that hosts within the same /120 subnet along with a closely-addressed second /120, both under the same 2607:fc86::/32 block, were probed by an additional 100M packets. This addition brings the total amount of traffic destined for this unallocated block to 192M packets. Thus, *a staggering 92% of our total background radiation packets was ICMPv6 traffic to or from these hosts*. There are over 3,500 unique source IPs that sent traffic to these two /120 groups, which totaled just 66 destination addresses. The sources originate from a wide swath of 378 different /32s, mostly in ARIN space but with some in each region.

We found that the size of these ICMPv6 payloads was either 10 (about 40% of the packets), 1,000 (30%), or 64 (30%) bytes. Sixty-three percent of the packets are echo requests, and the rest are echo replies. This suggests that traffic from hosts addressed in this block is being sent and with solicitation of replies, which we end up capturing. Indeed, an examination of the other datasets reveals some packets sourced from hosts in this block, which we end up capturing because they themselves happen to contact, for example, allocated-but-unrouted addresses. For example, this is the case for four hosts in this range sending around 5k packets in the B.AU dataset and one host sending ten packets in the B.AR dataset.

While we are unable to track down the location of the hosts using this invalid address block, the case study highlights how a single misconfigured network prefix can cause a large volume of pollution—92% of our background radiation packets.

3.2.6.2 Link-local Leakage

We mentioned above that we saw a very large fraction of packets from the IPv6 address fe80::224:38ff:fe7e:af00. This address's packets were the largest single contributor to our background radiation data, accounting for 34% of all received packets.

As addresses beginning with $fe80$ are "link-local" and meant to only be valid on local networks, their presence in our collection indicates a misconfiguration [90]. This address appears to use the EUI-64 encoded MAC address format for the host portion of the address (indicated by the FF:FE in the middle of the last 64 bits). Using this information, we determined that it corresponds to a MAC address whose vendor ID is assigned to Brocade (formerly Foundry Networks), a maker of SAN and network equipment. Assuming the MAC is not spoofed, this may indicate a faulty router. Even though it contributes a large volume of packets to background radiation, as the address is link-local and not allocated, we have no way to determine the operator to contact about possible remediation or root cause determination.

In total, we observe 205 link-local addresses in our background radiation data and over 605 in our overall three-month data, indicating that this is not a single occurrence. Likewise, we found 63 sources in our dark data and 1,678 overall sourcing packets from "unique-local" address space, which is analogous to RFC 1918 private address space and should not be globally routed [91].

These results suggests that current configurations on at least some Internet backbone and edge routers are allowing local IPv6 addresses to be globally visible, which is potentially dangerous, since it can allow a remote host to appear local to another host on the opposite end of the Internet.

3.2.6.3 Worm Activity

One of the traffic artifacts we were interested in was worm activity in IPv6. In particular, even randomly scanning worms should be visible to some degree in our data, if they exist, due to the sheer size and duration of our study. We would expect that even with a single scanning host using a slow scanning rate, such as 10 packets per second, we could expect to observe at least one packet within our four /12 blocks with 99.999% probability in 19.6 minutes [133]. Even though there had been no reports of any IPv6-capable variants of the major worms in IPv4, we decided to look for early signs of worm activity on two ports used by popular worms in IPv4.

We focused on UDP/1434 and TCP/445. These ports are used by Microsoft SQL Server and the Direct Hosted SMB protocol, respectively, and, in IPv4, they are exploited by worms such as SQL Slammer via UDP/1434, and, for example, Conficker and Sasser via TCP/445. The Slammer worm continues to be quite active in IPv4 despite the passage of a decade since the first outbreak [105]. Conficker scanning is also still highly prevalent, ranking it as the second most detected worm in the second half of 2012 [129, 169]. Validating some of this in our own work, the background radiation from the IPv4 network telescope we report on in Section 3.2.5 revealed over 99k UDP 1434 packets over the course of a week. We were able to positively confirm these as Slammer activity by its payload signature [168].

In our analysis of the IPv6 dark data, we noticed some small amounts of traffic on these two ports. However, upon closer inspection the data showed no signs of worm activity on either port. Our dark data contained just 18 packets to UDP port 1434, and it did not appear to be Slammer. The larger complete collected dataset also revealed nothing implicating worm activity on this port. Similarly, though we also observed some activity on port 445 (88k packets, involving a total of 92 unique IPs in the dark data), closer observations revealed that this traffic consisted of conversations between a single pair or a handful of IP address—not a typical scanning pattern. Likewise, expanding the search to the complete set of collected packets showed no indications of any scanning on these ports.

Overall, *we found no evidence of broad scanning nor prevalent malicious traffic in our data, a sharp departure from IPv4.*

3.2.6.4 NTP/BGP Services

Interestingly, we are able to find data destined to critical services such as NTP and BGP in our datasets. We find NTP traffic from over 62k unique IP addresses in the background radiation data. There are just 28 unique source IPs to 62,395 unique destinations among the packets we captured. Of the 28 sources, six are in 6to4 space (2002::/16), while the remaining are from 12 unique prefixes originated by 12 ASes—one Brazilian, one Canadian, one Mexican, and nine U.S., including two large ISPs. In B.Dark, the port used by NTP is the second most common source and destination UDP port, suggesting possible prevalent disruption of NTP activity by IPv6 misconfiguration we observe (although we note that clients often configure several NTP servers for resiliency, possibly prolonging the time to detection of the misconfiguration).

We also find a significant amount of BGP traffic in the background radiation data. In B.Dark, we see BGP packets between 338 unique IPv6 addresses. Examining the IPs of the 150 unique packet sources, we found that ten were clearly invalid, with addresses beginning with 6001::/16, a001::/16, or 1::/16, none of which fall under the global unicast address space (2000::/3); 26 of the sources were out of unrouted space, and the remaining 114 sources were originated from eight unique prefixes advertised by seven unique origin ASes: four US, two German, and one from the Philippines. These included four telecom operators and one very large global Internet search provider. In the one-day dataset, BGP traffic is the top contributor to RIPE's TCP traffic, comprising over 31% of the RIPE TCP packets, and involving eight IPs in six unique /32 blocks. The significance of BGP traffic among background radiation is that it indicates brokenness on the Internet's control plane itself, and 346 routers (if we assume no aliasing) with sources in several ASes shows that the problem is not isolated.

3.3 Beyond Background Radiation

Although we used a covering prefix methodology in order to maximize our visibility, the data that we ended up collecting greatly exceeded the traditional background radiation (dark) data that we initially set out to study. We focused on deeply characterizing the dark data in section 3.2 because it constitutes the first large and long-term glimpse into global background radiation in IPv6, but we now turn to an analysis of this new non-traditional background radiation data, which constitutes the majority ($\approx$95%) of the packets we captured. In the subsections that follow, we examine each of the other three types of packets: allocated/routed (AR), allocated/unrouted (AU), and unallocated/routed (UR); these comprise 35%, 60%, and $<$0.01% of the packets we captured, respectively.

3.3.1 Allocated/Routed (AR) Packets

As previously shown in table 3.4, 34.5% of the packets collected in dataset A and 36.0% of those collected in dataset B were destined to *allocated* and *routed* prefixes. Recall that, because we ended up receiving these packets, it is necessarily the case that a more-specific route to that prefix must not have been globally available at the time of the packet—at least not available between the packet's source and Merit's upstreams (AT&T and Hurricane Electric). Indeed, for several sample days we performed finer-grained analysis comparing the precise time stamp of each packet and the state of the global and local (Merit's) routing tables in order to validate that this was the case—i.e., the longest-matching-prefix routing rule was being followed. Packets only came to us when, at that instant, there was no known more-specific route, not to Merit's upstreams or not globally. In this section we first present a general examination of this allocated/routed (AR) traffic, followed by a description of a routing analysis we conducted to identify characteristics of this subset of IPv6 prefixes.

3.3.1.1 Traffic Characterization

Table 3.4 provides some high-level statistics for our AR data subset, which includes the number of packets, unique destinations, and TCP payload volume of the data in this category. Overall, we observe 4.32B packets in the three-month dataset. Of these, we categorized 1.56B packets as allocated/routed and these were directed to over 1.5M unique destination IP addresses. Below we highlight two of the key features that differentiate this data subset from the traditional background radiation. Recall that, since routes to prefixes in this category are temporarily or regionally available, *this category is more likely to be "normal" IPv6 traffic that is briefly disrupted, rather than traffic that permanently fails, as in the other categories.*

In terms of TCP flags, we find that, overall, SYNACK (76%) dominates SYN (22%) packets in the three-month dataset, B.AR. In the 24-hour collection, A.AR, it is SYN with 76%, and SYNACK at 5%. However, we note that in the A dataset, as shown earlier in Table 3.3, TCP packets are dominated by RIPE, which has a high percentage of SYN (78%) and very low SYNACK (just 2%). Recall that in the dark data, shown in Figure 3.2b, SYN packets generally dominated, i.e., connection attempts were more common. The general dominance here of SYNACK, instead suggests replies to SYN packets from these networks—replies that are unable to find a path back to the source of the antecedent packet due to the lack of a valid route.

TCP source ports are dominated by 443 and 80, at 22% and 14%, with port 5528 at 2% and all others (the next 6 being 25, 993, 995, 587, 110, and 22) below 1% of traffic. This indicates a large number of responses from web servers in the AR traffic. Notable also is the absence of port 53 in the top ten, which was the most common in the dark data. In the opposite direction, TCP destination ports are dominated by port 80 (10%) with other ports

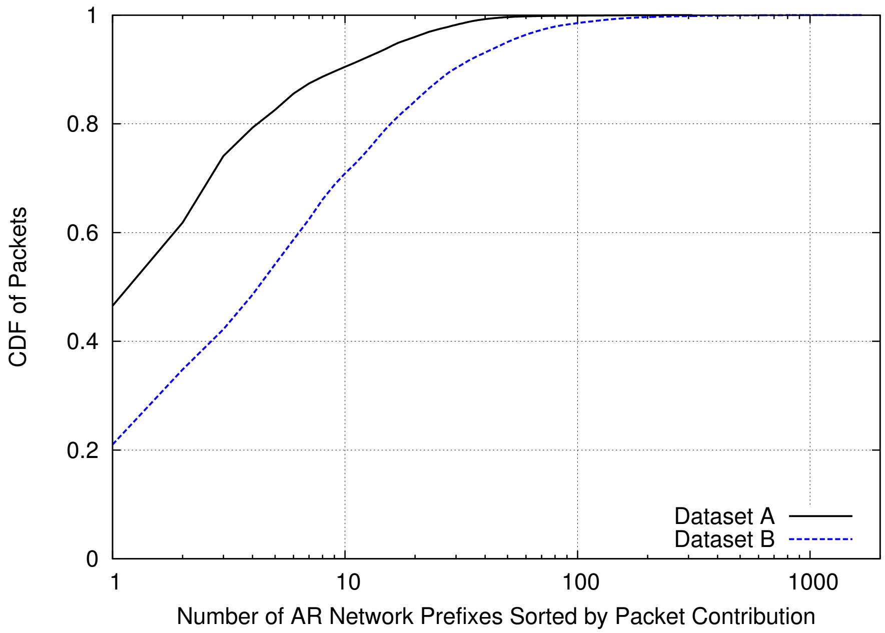


Figure 3.5: Cumulative distribution of contribution of packets by 311 AR nets in dataset A and 1,669 AR nets in dataset B.

in the top 10 all under 1%. In descending order, they are: 113, 179, 22, 25, 51413, 53, 8008, 443, and 11171. This suggests some web traffic, along with client connections to services in AR nets. The presence of BGP (TCP/179 is a destination of 0.2% of packets), just as in the dark data, shows some control plane misconfiguration. In the RIPE data in dataset A.AR, the lists are similar, with 80 and 443 topping the port list in both directions. Overall, the port and protocol AR data, along with the high presence of SYNACK packets, fits the profile of client networks being prominent among AR networks.

3.3.1.2 AR Network Prefix Properties

To better explain this large fraction of our traffic, we sought to more closely characterize the networks in the AR set of destinations. Through our analysis (as described in 3.1.5) we were able to identify 311 AR networks in the dataset A and 1,669 in dataset B. Figure 3.5 shows the cumulative distribution function of these prefixes according to their contribution of packets to the AR data. We note that just one prefix is responsible for over 46% of the packets in A and just one (different) prefix for 21% of the packets in B, and that the top 10 prefixes account for 90% and 71% of the packets, respectively.

Table 3.11: Three properties of AR destination prefixes and their origin ASes as well as averages for the Internet at large (IPv6 and IPv4). All numbers are averages over the dataset, except for withdrawal events, which are averages per 24 hours.

	Dataset A			Dataset B		
	IPv4	**IPv6**	**A.AR**	**IPv4**	**IPv6**	**B.AR**
Route Analysis (over entire dataset)						
Networks	485,233	12,841	311	502,483	13,462	1,669
R.V. Peers	95.22	65.67	28.68	102.75	68.34	65.48
Withdrawals	0.12	0.65	8.22	0.09	0.33	0.55
K-Core Decomposition Analysis (First Day of Dataset)						
Origin ASes	36,559	6,511	247	36,651	6,586	1,018
AS Coreness	2.23	9.87	8.94	2.21	9.23	6.62

We hypothesized that these networks may differ systematically from the overall pool of IPv6 networks. To study this, we examined several properties of these prefixes and their origin autonomous systems (AS). For each prefix seen in the data, we tabulated the number of Route Views peers that had routes to that prefix. This is a measure of *prefix propagation*. We also counted the number of withdrawal events (which we define as the number of withdrawal messages divided by the number of peers) for each prefix seen. This is a measure of *stability*.

Lastly, for each sample day, we examined the *coreness* [12, 83] of all origin ASes. In graph theory, a k-core is a maximal subgraph in which every vertex has at least degree k. The coreness of a vertex v indicates the maximum value of N for which v is in the N-core. As used in studying routing topology, coreness is a measure of AS *connectivity*. The coreness values for each origin AS were obtained via an analysis of BGP table snapshots collected from Route Views and from RIPE RIS on the first day of each dataset period.

For each of the three measures, we tabulated the overall averages for all Internet IPv4 and IPv6 prefixes, separately, as well as the values for just our list of seen AR prefixes in each sample. Table 3.11 summarizes our results.

Propagation As can be seen, there appear to be differences between the pool of AR prefixes and the overall pool of IPv6 prefixes according to our measures. The average count of Route Views collector BGP peers advertising paths to the seen AR prefixes is about half the count of the average IPv6 prefix in the one-day sample (A), though much closer (65.48 versus 68.34) in the three-month sample (B). As with the stability measure discussed next, the difference between the three-month dataset and the one-day dataset likely has to do with the larger aggregated pool of AR prefixes over the three month period. Even a single captured packet destined to a mostly stable prefix due to a brief withdrawal would land it in our AR list. Thus, the longer time period is likely to include more stable routes than period A. We have confidence in these numbers as we also conducted these types of analyses for three single-day subsamples of the B time period and obtained very similar results to the single-day numbers for A presented here. The difference between AR prefix and overall IPv6 prefix peer counts suggests that these *AR prefixes have weaker propagation* in the global BGP routing table.

Stability Next, examining the withdrawal events, we see that the overall pool of IPv6 prefixes sees an average of 0.65 withdrawal events per prefix per day for the single-day dataset, A, and 0.33 for the three-month, B. Here, we see a stark difference, as the AR prefixes exhibit 8.22 withdrawals per day in A and 0.55 withdrawal events in B. Even the latter difference is meaningful, since it indicates that the average AR prefix has 66% more withdrawals per day than the average IPv6 prefix (at 0.33 withdrawals). This suggests that AR prefixes are *much less stable* than average networks.

Through a fine-grained analysis of timestamps, mentioned above, we initially observed that some AR networks ought to have received packets that, instead, we received. We say "ought to" as routes to these networks were available in some part of global BGP, based on Route Views data. The stability and connectivity measures here explain why we received these packets in spite of their intended routes being seen by at least some Route Views peers: the routes must have been unavailable to Merit's providers. Our deeper analysis of Merit's BGP perspective confirmed that Merit's upstreams did *not* have more specific routes for the given weakly-routed prefixes at the instant each packet arrived. *Operators can mitigate lost traffic in this category via route monitoring, such as by consulting Route Views or an alert service.*

Connectivity Coreness is utilized to assess the connectivity and centrality of a network. Intuitively, high coreness indicates a better-connected AS, such as a large ISP. Typical values for all IPv6-enabled ASes range from 0 (stub networks) to 70 (the best-connected ISPs).

The measured coreness of 8.94 in dataset A (versus 9.87 for all IPv6 prefixes) and 6.62 for the three-month sample suggest that *the origin of these unstable and weakly-advertised prefixes is more likely to be a small network*. This is to be expected, since smaller networks might be less likely to be managed as rigorously as large Internet service providers, whose main business is the network.

3.3.1.3 IPv6 Prefix Comparison to IPv4

Compared to the overall pool of advertised IPv6 prefixes, we found that AR prefixes had poorer propagation, were less stable, and were originated by less well-connected autonomous systems. As IPv6 is still at an early deployment phase, we expect IPv6 prefixes, as a whole, to have poor routing. To perform a systematic juxtaposition of IPv4 against IPv6, we repeated the analyses described above to obtain the corresponding IPv4 metrics.

Table 3.11 also depicts these measures applied to the entire pool of IPv4 prefixes. We first observe that the IPv4 networks are, in general, better-connected to Route Views monitors, with average peer counts of 95.22 and 102.75 for the A and B samples, respectively. Examining the average withdrawals, we see that IPv4 prefixes are much more stable, exhibiting just 0.12 (A) and 0.09 (B) withdrawal events per prefix per day. Finally, we see that the average coreness of IPv4 prefixes is much lower than that of IPv6. This means that the average IPv4 prefix is originated by a smaller network than the average IPv6 prefix. This is not surprising, since it has been shown that IPv6 deployment is more prevalent at the core of the Internet than at the edges [56], and, both by definition and by necessity, the core is highly connected.

These findings suggest that, indeed, IPv6 routing is not as mature as IPv4, and that the AR networks are an even less mature subset of the IPv6 pool. This result is intuitive, given the early stage of IPv6 deployment, the fact that it only carries a small fraction ($\approx$0.2%) of Internet traffic [43], that operators generally have much less experience with the new protocol, and that both hardware and software support for the new protocol is still not on par with IPv4 support.

3.3.2 Allocated/Unrouted (AU) Packets

As table 3.4 shows, 60.4% of dataset A and 59.2% of B packets are destined for addresses that match *allocated but unrouted* prefixes. This is interesting, as it suggests traffic to networks that were assigned and configured but which were not intended to be globally reachable—perhaps because they were intended for internal (i.e. “private”) organizational use, in spite of existence of the *unique local* prefix, fc00::/7, analogous to IPv4 RFC 1918 space, for this purpose [91, 148]. Indeed, traffic patterns of the two largest contributors we found in this category, were confirmed examples of this.

Together with the Allocated/Routed (AR) packets, this category of traffic constitutes nearly 95% of packets we saw. This high-level finding, that a majority of collected traffic via a covering prefix network telescope belongs to allocated networks, is consistent with an earlier experiment with just APNIC’s /12 reported by Geoff Huston [99]. Here, we highlight some of the largest contributors we discovered in this subset of the data.

We observed periodic spikes in the overall volume of traffic in the ARIN dataset. These large jumps, resulting in a one or two order of magnitude spike in ARIN traffic, occurred daily at around noon UTC. Upon investigation, we found that they were largely composed of DNS replies. All of these packets had their destinations set to the same value, which we omit here to preserve organizational privacy. In total, we received over 444M packets (over 423M of which were DNS responses) destined to this one IP. It is the top contributor and accounts for over 17% of all packets in the AU dataset. The address is in a network block that has been allocated to a managed hosting company but is not seen in any routing tables visible to any Route Views monitors.

We contacted operators at this organization to inform them of the traffic and learn the cause. It was quickly determined that the source of the traffic was a misconfigured DNS server. The server was assigned a (globally unique) address meant by the organization for internal use, but it was using this address as the source of external packets it sent. Since this address was from a prefix that was not globally advertised in BGP, the replies to that DNS server all ended up being routed, via our covering prefix, to our collector. This misconfigured server was not previously noticed by its operators because it was part of a cluster of several resolvers, which *were* correctly configured. After the company began advertising the network block in which this server was addressed in mid-August 2013, we stopped receiving these packets, and our total ARIN traffic, in bytes, fell by about half.

Another large contributor to the AU dataset is traffic destined to a network block allocated to the R&D unit of a large U.S. wireless phone provider. We received 1.1Bn packets destined to 216 destinations in one prefix allocated to this organization. This constitutes nearly 44% of the packets in the AU dataset. There were over 6M unique IPs sending traffic to these 216 destinations, and nearly 6M were from the same /32 block as the destinations. Most of the sources were from sub-blocks of the /32 that were publicly routed. This strongly suggested that the prefix in question was being used internally but that some routing misconfiguration was causing traffic from other prefixes within that organization to be misdirected out to the Internet instead of toward that internal network. We established contact with the company and they confirmed that this address space is used internally and not meant to be globally reachable. At publication deadline, they were still investigating why these internal hosts were sending their traffic to our collector instead of to the local private network. In general, operators can avoid leaking such internal traffic by either using unique local addressing internally or by placing access lists on edge routers to either block or log such unintended address use.

3.3.3 Unallocated/Routed (UR) Packets

The third and final category, accounting for less than 0.01% of packets in both the A and B time periods, is those whose destination address matching prefixes that were not allocated by an RIR but which did have routes in the BGP routing table at least temporarily during our data collection period. Again, recall that, these prefixes were not globally reachable during the instant we captured a given packet, otherwise it would not be routed to us. We suspected that these unallocated but either partially- or temporarily-routed prefixes may be either the result of RIRs withdrawing allocations while operators continued to use the address space, the result of experimentally-advertised prefixes, or the result of misconfiguration.

In dataset period A, there were two prefixes in this category, 2a02:510::/32 and 2606:8900::/32, and there were also two during period B: 2607:fd20::/32 and 2406:7a00::/31. Each of these four prefixes was unallocated at the end of our data collection windows. However, the first of the four was in allocation for several years prior to our collection, the second was allocated a few days after the collection period, the third was allocated for part of our B collection period then returned, and the last was never allocated (but a prefix one bit longer was).

In each case, these prefixes were known to Route Views by only a single peer router (versus ≈75 for the overall average IPv6 prefix, as discussed in section 3.1.2), suggesting very limited announcements, likely for testing or due to misconfiguration. Because of this, the traffic in this category most closely resembles the unallocated and unrouted (i.e. background radiation) traffic.

Focusing on the packets, we find that ICMPv6 comprised 94.5 and 98.1% of the traffic in this category in datasets A.UR and B.UR, respectively. Upon further examination, we discovered that the vast majority of packets were due to traffic from hosts associated with researchers. In dataset B, for example, 83.5% of packets came from what we confirmed with CAIDA to be their experimental hosts. Another 8.8% of the packets were confirmed by BBN Technologies as hosts that are part of their experiments. A third source, with 2.8% of these packets was confirmed as belonging to another research organization we contacted, in Canada. *In total, at least 95.1% of the packets in the B.UR (and 97.5% in the A.UR) dataset were the result of confirmed researcher activity.*

3.3.4 Near Misses

When we modified our initial RIPE NCC /12 announcement into a /13 and a /14, eliminating the 25% of the address space from which RIPE was making customer allocations, the volume of RIPE traffic we saw dropped disproportionately—by three orders of magnitude. We considered whether this was representative of a general case, i.e., if the majority of network telescope traffic we received was clustered near allocated address space. We discussed most of the answer to this above by showing that, after our allocation and routing analysis, around 95% of the captured traffic was, indeed, destined for allocated prefixes (the

AU and AR data subsets). However, what about traffic to unallocated-and-unrouted space?

We hypothesized that if misconfiguration was a large source of the traffic to unallocated networks, then the target destinations might be addresses that are lexically close to existing, active prefixes (i.e., "fat finger" errors). To examine this systematically, we conducted a Levenshtein distance [119] analysis of each packet's destination in fully-expanded ASCII format, comparing it to all known routed networks (up to the prefix length). An analysis of three day-long samples of destination IPs of packets we captured (all of dataset A, as well as all packets on the first day of each month of dataset B) yielded the somewhat unexpected result that *every destination address had some minimal edit distance of at most* 2. Between 39% and 81% of packets, depending on the sample, were within one hex character change away from an existing routed prefix, and the rest were within two. Since our other analyses showed that such a large percentage of traffic is for allocated networks, this should not be too surprising, as routed prefixes are often part of or near other allocated addresses. However, that this held even for the pure "dark" traffic to unrouted-and-unallocated space was unexpected. While we can't confirm this, it suggests that one explanation for the dark traffic could simply be typos, as a single character change would bring each packet we saw under a legitimate routed prefix.

3.4 Summary

In this chapter, we presented results from deep analysis of a broad observation of Internet background radiation on the IPv6-enabled Internet. To the best of our knowledge, this study, which relies on five large covering prefixes, is the first of its kind in terms of visibility and allows us a deep and broad look into early misconfiguration on the burgeoning IPv6 network. We conclude that broad scale malicious backscatter and malicious scanning (or any at-scale scanning at all) as well as worm activity appear to all be absent on the new network during the three months of our study.

The results for misconfiguration are more concerning. The covering prefix approach provided visibility into routing and configuration errors previously hidden from this type of study—i.e., those affecting allocated prefixes. Nearly 60% of packets we captured were destined to allocated but unrouted destinations, seemingly due to traffic sourced by networks meant for internal use being "leaked" into the Internet; this was the case for the two largest contributors to this category. Another third of packets were destined to allocated space that had, at least briefly or sparsely, advertised routes. We also analyzed BGP updates for these prefixes and found them to be less stable and advertised than the average IPv6 prefix, which is itself less stable and less well-advertised than the average IPv4 prefix. Together, these data suggest that while not necessarily a systemic problem, routing misconfiguration is common on the emerging IPv6 network, and, especially given the BGP update analysis, the quality of routing is worse that in IPv4. While not completely surprising, this result suggests that, at least in the near to medium term, the increasing flow of production traffic on IPv6 may experience a lower quality network than the legacy protocol provides.

CHAPTER 4

IPv6 Network Security Policy

In this chapter, we examine the state of deployed IPv6 network security policy. In previous chapters we showed, first, that the rate of adoption and nature of use in recent years has shifted from past patterns dramatically to a regime of rapid growth and finally to apparent production use of IPv6, and, second, that routing management of the IPv6 network, as reflected in misconfiguration and routing problems visible in a broad network telescope study and in associated BGP analysis, points to areas lagging the incumbent IPv4 network. We were interested in taking a second deep look at one important higher-level aspect of how the protocol is being managed in these early deployments, namely, whether port filtering security policy, as codified in IPv4, is being faithfully applied to the new protocol.

Our focus in this chapter is not new security vulnerabilities that arise from problems in the IPv6 protocol standards or in weaknesses in early implementations of those standards. Rather, we simply seek to study whether the same protections that have been applied on the IPv4 network are being brought online for hosts as they are configured with a second potential entryway for attackers. The same threats that operators are concerned with enough to institute port filtering policy against over IPv4 should also prompt them to defend their networks via the new protocol. Indeed, numerous standards documents, best practice guides, and security talks have urged operators to bring their IPv6 network to parity with IPv4 in this regard (e.g., [5, 31, 32, 66, 75, 112]).

When planning this study, we hypothesized that, in fact, differences between IPv6 and IPv4 security policy would be measurable, likely resulting from several factors related to the lower maturity of the new protocol, including shorter operator experience due to the age of the protocol and legacy management tools that do not support IPv6 (e.g., Netflow version 5 [173]) or have incomplete support [92, 128]. Indeed, anecdotal evidence suggested that security policy incongruity between IPv4 and IPv6 was indeed common, and our goal with this study was to measure the extent of the problem [32, 161]. If network protections are not as robust for IPv6, these dual-stacked hosts are, effectively, lowering their defenses by enabling IPv6—essentially opening up an attack vector that the operators of these systems had already closed for IPv4. As we show in this chapter, based on our measurements of 25K dual-stacked routers and 520K dual-stacked servers on commonly attacked ports, this appears to be the case, and the extent of differences found is large and concerning.

4.1 Methodology

4.1.1 Developing Target Lists

To explore potential policy discrepancies between IPv4 and IPv6 we first must find dual-stack hosts such that we can measure application reachability (i.e., connection success) to the same target via different protocols. As we describe below, we harvest lists of IP addresses and host names to start the process. Given a set of hosts, our strategy is to use DNS names as the basic connective tissue between hostnames, IPv4 addresses, and IPv6 addresses. One reason for this approach is that "DNS is one of the main anchors in an enterprise [IPv6] deployment, since most end hosts decide whether or not to use IPv6 depending on the presence of IPv6 AAAA records..." [31]. Each host H^x in our dataset contains three sets of labels that represent H^x: H^x_N is the set of names, H^x_4 is a set of IPv4 addresses and H^x_6 is a set of IPv6 addresses. After obtaining an initial list of hosts (see below) we then detect common elements across multiple hosts. We compare hosts pair-wise, as follows:

$$s = (H^x_N \cap H^y_N) \cup (H^x_4 \cap H^y_4) \cup (H^x_6 \cap H^y_6) \tag{4.1}$$

When $s \neq \emptyset$ there is overlap between the two hosts and therefore we replace H^x and H^y with a new H^z that represents the merger of the two original hosts, as follows:

$$H_N^z = H_N^x \cup H_N^y \tag{4.2}$$

$$H_4^z = H_4^x \cup H_4^y \tag{4.3}$$

$$H_6^z = H_6^x \cup H_6^y \tag{4.4}$$

Finally, we prune hosts that do not have at least one valid and routable IPv4 and IPv6 address.

There is a possible problem with using DNS as the connective tissue. The labels within a group could actually represent multiple distinct machines and not a single dual-stack machine [23]. As part of our probing we obtain host signatures (e.g., an *ssh* host key). In roughly 3% of the hosts we detect multiple signatures across different host labels. We believe this is a strong indication that our process is predominantly identifying dual-stack hosts, and we discuss this point further in section 4.2.

There is also a possible issue with using DNS names as the source of targets for our probes in the first place. Specifically, because we are starting from hosts that have names in DNS (more precisely, that have associated sets of A, AAAA, and PTR records), we will naturally exclude those hosts without DNS presence or without complete record sets. We acknowledge that this may bias our results. However, we believe this bias would result in an underestimate of the core policy misalignment finding. First, without both A and AAAA records, server hosts could not be reached by most clients (except when addresses are directly hard-coded in applications, which is uncommon). Routers could still be functional without these records, but ease of management and configuration dictates that interfaces are often named. Further, for PTR records, maintaining these is a known best practice [21], and, in the case of some servers, may be necessary to operate (e.g., for sending mail [120]). Thus, because maintaining all three record types is generally a sign of correctly operating an Internet-facing service interface (i.e., public router or server), we believe results for these targets will likely produce a conservative picture of actual security policy misalignment in the overall dual-stacked server and router population, as the larger population will include less properly-operated hosts.

As we note above, we start the process of identifying dual-stack hosts using two different lists of hosts, as follows.

Router List: Our first focus is on Internet routers, which form the core of the Internet and hence the correct application of security measures is crucial. An attacker that can compromise a router might be able to redirect traffic for man-in-the-middle attacks, cause a complete or targeted network outage, or adjust filtering rules on the router that otherwise serve to protect the network from other attacks.

Our router dataset is derived from router interface IP addresses found in Internet-wide *traceroute* data taken by CAIDA's Ark measurement platform [26]. Each system in the Ark platform infers the forward IP path with *traceroute* measurements to random IPv4 addresses in every routed /24 prefix, as well as to the first (::1) and a random address in every routed IPv6 prefix. We extracted the source IP addresses of ICMP hop-limit responses (intermediate routers) archived during December 2014 and performed reverse DNS lookups to obtain names where available. This forms the initial list of routers, which we then refine using the procedure above.

Server List: Our second list consists of servers, which make specific services widely available over the network. While not as central to the system as routers, the security posture of these systems can be of significant importance. We initialize our server list with the Rapid7 DNS ANY datasets from the `scans.io` repository [158]. This dataset represents DNS ANY queries for a set of hostnames gathered from (*i*) more than 200 DNS TLD zone files (from the popular .com and .net to emerging TLDs such as .farm and .toys); (*ii*) reverse DNS names of IPs detected in HTTP scans, SSL certificate scans, and IPv4-wide reverse-DNS lookups; and (*iii*) `<a>` tags in HTTP responses to HTTP scans. The resulting names are filtered in an attempt to remove all hostnames that appear associated with static or dynamic customer-premises IPs (e.g., ip-192-168-0-1.example.com), which leaves mostly hostnames that do not appear to be automatically-generated for consumer Internet endpoints[1]. Due to the provenance of the names and addresses in the server list, including this applied filter, although it could include some dual-stacked clients, we believe that these are in the minority. To validate this, we examine what fraction of the IPv4 addresses

[1] The exact filter we use is available [146]

associated with each host is part of known consumer network blocks, using the SpamHaus Policy Block List (PBL) [164]. We find that 97% of the hosts are not in PBL ranges, as expected. The raw DNS dataset represents approximately 1.4B name to address mappings. After first culling the list to just A and AAAA records, as sketched above, we then detect common names and addresses to reduce the set to 2.4M addresses associated with 950K dual-stacked hosts.

4.1.2 Probing

We perform active probing to assess security posture differences between IPv4 and IPv6. When probing routers we use services that are some combination of (*i*) likely to be running on routers (e.g., SSH), (*ii*) crucial to router operation (e.g., BGP) and/or (*iii*) problematic when leveraged by an attacker (e.g., NTP [45]). Thus, the exposure of all of these ports generally increases the attack surface of routers. We probe these services on all routers in our dataset[2]:

> ICMP echo, SSH (TCP/22), Telnet (TCP/23), HTTP (TCP/80), BGP (TCP/179),
> HTTPS (TCP/443), DNS (UDP/53), NTP (UDP/123), SNMPv2 (UDP/161)

Similarly, after developing the list of dual-stack servers we perform active probing, but with a different set of application types that are more apropos for servers than routers, as follows:

> ICMP Echo, FTP (TCP/21), SSH (TCP/22), Telnet (TCP/23), HTTP (TCP/80),
> HTTPS (TCP/443), SMB (TCP/445), MySQL (TCP/3306), RDP (TCP/3389),
> DNS (UDP/53), NTP (UDP/123), SNMPv2 (UDP/161)

To select these, we consulted literature on prevalence of scanning [59], prevalence of port blocking [114], as well as application DDoS amplification susceptibility [152]. We wanted to minimize probing ports of lower deployment or interest, as well as constrain our set to a

[2] After initial experiments, several applications were showing minuscule dual-stack response rates (generally a tenth of a percent or less). To focus on more prevalent applications, we dropped these from study. They included, for routers: TFTP, and for servers: IPMI, MS-SQL, NetBIOS, SSDP, and VNC. We also excluded SNMPv1, as results closely matched SNMPv2.

small number so as to minimize load on targets. Ultimately, the potential impact of breach was the most important factor for inclusion.

We use two probing methods to collect the data we use in the remainder of the chapter. The "basic" probing consists of single probe packets to each service with both IPv4 and IPv6. These are done using *scamper* [122], which is a parallelized bulk probing tool that supports various types of probes, including *pings* over ICMP, TCP, and UDP, for both IPv4 and IPv6. We did not use zmap because it does not support IPv6 [60]. For ICMP this is an echo request, for TCP it is a SYN segment, and for UDP this is an application-specific request (e.g., DNS A query for "www.google.com", NTP version query, or SNMP query for the sysName.0 MIB using the default public community string). We probe roughly every two weeks starting in mid-January 2015 and mid-February 2015 for the routers and servers, respectively, through July 2015. We found little difference between these data collections and therefore focus on the router collection from February 19 2015, which we denote $\mathcal{R}_B$ and the server collection from April 10, 2015, denoted $\mathcal{S}_B$.

Our second probing strategy is based on *traceroute* style measurements using a variety of probe traffic types (again using *scamper*). Because there is no implementation of traceroute that considers application-responses (traditional traceroute deliberately chooses high-numbered, unused ports to solicit ICMP port unreachable error messages) we extended scamper's traceroute implementation to record UDP application-level responses. We configured *scamper* to probe the ports listed above using each IPv4 and IPv6 address of every host. To limit the burden we place on our measurement targets we tested one port at a time and in random order. Our goal in doing so was to remove the possibility that we would trigger rate limiting by probing the host too quickly and thus raise the possibility that a host that was initially responsive would become unresponsive, conflating a rate-induced outage with a policy to discard specific types of packets. We configured scamper to probe at 5000 packets per second; meaning the router list took approximately eight hours to measure while the server measurements took approximately 22 hours. Scamper was configured to send a single packet with each TTL (or hop limit) value and to wait for 5 sec for a response to each query. We paused for at least 1 sec between measurements to a given host. Therefore, despite our relatively high probing rate, we spread the load across a set of targets so that we had little impact on individual hosts measured.

We also collected a number of these *traceroute* datasets. Again, our analysis shows similar results across time and therefore we concentrate on the router dataset collected on June 5 2015, which we denote $\mathcal{R}_T$, and the server dataset collected on July 10 2015, which we denote $\mathcal{S}_T$.

Table 4.1: Dataset summary.

	Probe Date		**Addresses**		**Hosts**	
Dataset	**(2015)**	**Names**	**IPv4**	**IPv6**	**Total**	**Suitable**
$\mathcal{R}_B$	Feb 19th	38K	41K	41K	35K	25K
$\mathcal{R}_T$	Jun 5th	38K	41K	41K	35K	25K
$\mathcal{S}_B$	Apr 10th	8.3M	1.0M	1.4M	947K	520K
$\mathcal{S}_T$	Jul 10th	8.5M	1.0M	1.4M	951K	533K

Table 4.1 describes the datasets we use in the remainder of the chapter. The "suitable" column represents the hosts we analyze and is the set of hosts which respond to ICMP echo requests in both IPv4 and IPv6. We only measure policy congruity on suitable (responsive) hosts to avoid mistaking a host that is unreachable at all from one where application policy controls are enforced. While this test is not foolproof, we know these hosts are responsive to both IPv4 and IPv6. We could be excluding hosts that apply different ICMP policies to IPv4 and IPv6, as well as those that filter all ICMP requests. However, we believe the set of hosts we leverage is large enough to give us broad insight into the policy differences between IPv4 and IPv6 across the Internet.

The hosts we measured are spread across the network, and encompass 58% of the dual-stack ASes observed in public BGP tables (all available Route Views [166] and RIPE RIS [150] RIB files) for midnight February 1, 2015. The $\mathcal{R}_T$ target list contains hosts from over 2K routed prefixes, 1K autonomous systems (ASes) and 70 countries. The $\mathcal{S}_T$ target list, on the other hand, contains hosts from over 15K routed prefixes, 5K ASes and 133 countries. Unsurprisingly, while we leverage a breadth of targets, the set is also skewed. In the $\mathcal{R}_T$ list we find that 19 ASes that belong to the ten most-represented network operators in our list account for half the hosts. Similarly, we find that the ten ASes belonging to ten large hosting and context provides make up half the hosts represented in the $\mathcal{S}_T$ list.

4.1.3 Ethical Considerations of Probing

Research involving the active measurement of networks potentially creates ethical issues as both the conduct of such research and the disclosure of results thereof may result in harm to a variety of stakeholders including, but not limited to: research institutions, service providers, network operators, and end users. We take note that, while the security community has not reached consensus on standards for such research, existing published work in the field [60], as well as broad ethical guidelines [19] provide a roadmap for how one may minimize the potential for such harms. For example, in the conduct of this research we: (*i*) signaled the benign intent of work through WHOIS, DNS, and by providing research details on a website on every probe IP address; (*ii*) significantly rate limited the scanners to minimize impact; (*iii*) limited ourselves to regular TCP/UDP connection attempts followed by RFC-compliant protocol handshakes with responsive hosts that never attempt to exploit vulnerabilities, guess passwords, or change device configurations; (*iv*) we respect opt out requests and seed our opt out list with previous opt out requests provided to other researchers [60]. In mitigating disclosure harms, we carefully avoid providing target lists in the published result, and notify, where feasible, the most egregious networks prior to publication, so that they may correct unintended policy differences.

4.1.4 Result Interpretation

One final methodological task involves interpreting the results of the probes. First we must decide if a probe succeeds or fails. We define success as reception of (*i*) an ICMP echo reply message in response to an ICMP echo request, (*ii*) a TCP SYN+ACK in response to a TCP SYN and (*iii*) a UDP response to a UDP request. We consider anything else—including no response—as connection failure (e.g., ICMP unreachables, a non-SYN+ACK TCP packet). Once we have a decision for probes within some H_4^x, we make a final IPv4 determination based on majority vote across all IPv4 addresses when there are multiple. Likewise for the IPv6 addresses H_6^x.

Lastly, a minor note on terminology: the versions of IP we study (at OSI layer 3) as well as the applications we probe (at layers 5-7) can all be called *protocols*. To avoid confusion, however, we reserve that term for the IP version under study and instead use the term *application* to denote the protocol/application at the higher layers (e.g., SSH, NTP, etc.) for which we are measuring connectivity.

4.2 Calibration

A core assumption of our work is that all the labels we find for some host H^x point to the same host. This is not of only theoretic concern, but previous work shows that DNS names mapping to IPv4 and IPv6 addresses do not always point at a dual-stack host, but instead to multiple hosts [23]. Since our goal is to understand the security posture of dual-stack hosts, we first calibrate our method for aggregating labels into hosts. To test our assumption we seek to collect a set of application-level information for each host in the $\mathcal{R}_B$ and $\mathcal{S}_B$ host lists in both IPv4 and IPv6, as follows.

HTTP: We send each host that responds to TCP port 80 connections a HEAD request and extract the server version string (including operating system) from responses (e.g., “Apache/2.2.22 (Debian)”). While we don’t exclude any, the three most frequently returned strings are indistinct as version and OS are not provided (e.g. “Apache” (39%), “nginx” (23%) and “cloudflare-nginx” (6%)). The next 20 most frequently returned strings are more specific and, thus, provide a stronger fingerprint for matching.

HTTPS: We are able to collect an extensive set of information about each host responding on TCP port 443 probes, using both the *openssl* client and *NMAP* with the ssl-enum-ciphers.nse script [4], including: (*i*) the supported cipher suites (for all except SSLv2-only hosts); (*ii*) the supported SSL/TLS protocol subset of {SSLv2, SSLv3, TLSv1, TLSv1.1, and TLSv1.2}; (*iii*) the actually negotiated protocol between client and server; and (*iv*) the server’s certificate fingerprint.

SNMP: For SNMP, we retrieve two MIBs—*sysDescr.0* and *sysName.0*—via SNMP version 2c *get* requests to the *public* community. Responses include the OS (including version) and an administrator-set system name. We require responses to both to obtain an identifying fingerprint.

NTP: For each host responding to UDP port 123 queries we issue the *version* command

Table 4.2: Alias validation via application signatures. A majority (96% of $\mathcal{R}_B$ with data and 97% of $\mathcal{S}_B$) of hosts with signature data matched fingerprints among all host address members.

	$\mathcal{R}_B$ **List**		$\mathcal{S}_B$ **List**	
Application	Hosts	Same Sig	Hosts	Same Sig
http	269 (1.1%)	97.0%	235,575 (46.2%)	99.2%
https	183 (0.8%)	96.7%	96,468 (18.9%)	94.2%
snmp2c	12 (0.1%)	100%	41 (0.0%)	95.1%
ntp	843 (3.6%)	97.0%	3,462 (0.7%)	99.1%
ssh	603 (2.6%)	96.7%	218,100 (42.8%)	98.9%
mysql	–	–	1,055 (0.2%)	99.5%
Overall	1576 (6.7%)	96.4%	303,111 (59.4%)	97.1%

using the *ntpq* tool. This typically provides a semi-structured string that contains: *version, processor, system, stratum, precision, refid, reftime, frequency, status,* and *associd* among other fields that we do not further utilize as they are less common or naturally vary across queries, so are not useful for fingerprinting.

SSH: We use the *ssh-keyscan* utility (part of the OpenSSH-clients tools) to obtain the SSH server version, the key length of the server's encryption key, and the fingerprint of the key.

MySQL: For hosts with open MySQL ports (TCP/3306) we send two newline characters which causes most servers to print an identifying banner, which we harvest (stripping control and unprintable characters). As we note above, we do not probe the MySQL port on routers, and, therefore, this information is only available for the server host list.

For each host x on our host lists we collect the above information via probing to every IP address in H_4^x and H_6^x. We then check for consistent behavior across all applications that respond on all addresses. When we find consistency across all IP addresses for H^x we conclude that the host is highly likely to be a single dual-stack host. Even if this conclusion is wrong, we believe the identical configuration indicates the operator's intention is to provide the same service across IPv4 and IPv6 and therefore policy differences are important to illuminate.

As an upper bound on our ability to match fingerprints, we first assess the general openness of our target hosts to providing the information we used for this consistency checking. If we collapse our overall probe results for each host across all applications tested and both IPv4 and IPv6 (i.e., a very coarse-grained analysis of the results we discuss in the sections to come), we find that we can access at least one of the above signature-providing services via at least one of a given host's IP addresses for only 44% and 76% of the hosts in $\mathcal{R}_B$ and $\mathcal{S}_B$, respectively. In other words, our technique will have no data at all to assess match consistency in over half the routers and for nearly one-quarter of the servers. In fact, since we need at least one IPv4 and one IPv6 address in order to do this validation, the number of hosts we can actually compare signatures for across the IP versions is even lower (7% and 59%, respectively, as discussed below). However, we stress that we are calibrating our *technique* of aggregating sets of labels (i.e., results for IPs grouped via their associated A, AAAA, and PTR records) into hosts and not each individual assessment. As such, the sample, though biased toward more open hosts, seems sufficiently large to reasonably represent such name-based aggregation.

Table 4.2 shows the results of our consistency probing. For each host with signature data from at least one of these signature-providing applications open on all associated addresses, which probed, we find a high level of consistency—96% for routers and 97% for servers. This roughly agrees with previous work that shows that, while it is not exceedingly rare for hostnames with both IPv4 and IPv6 addresses to represent different machines, 93% of the time they in fact do represent the same system [23]. For servers, our results cover nearly 60% of the hosts in the $\mathcal{S}_B$ host list. On the other hand, we can make a signature-based comparison for only roughly 7% of the routers in the $\mathcal{R}_B$ list. While the coverage in $\mathcal{R}_B$ is low, we note that, since routers have less general openness than servers, we expect our consistency check for routers to be useful in fewer cases. Even with the low coverage we believe the high consistency rate validates our methodology for aggregating labels into hosts.

The small fraction of hosts for which a signature match between the IPv6 and IPv4 addresses fails suggests operators using a separate server for IPv6 than for IPv4 behind the hostname. We can speculate as to various explanations for such a configuration, including an obvious one of matching resources to load where the IPv4 address points to a load balancer, as most traffic is via IPv4, while the IPv6 points to a single specific server that supports IPv6. We do not exclude the hosts without matching signatures nor those failing matches, mainly for the reason that we contend that *the semantics of a single hostname are a single service*. As such, whether it is deployed on one physical machine, configured identically, or neither, Internet users and application routers do not expect that the services available via a hostname differ based on the network protocol used to reach them, just as they don't expect DNS-based load balancing with multiple A records for a name to provide different service depending on the address their host happens to select from the several available in DNS. In other words, while the calibration in this section is useful for understanding the underlying population of machines that our hosts represent, the policy misalignment we find is orthogonal to and no less troubling whatever these signature matching results for any name show.

4.3 Baseline Policy Discrepancy

Overall, 26% of routers and 26% of servers we tested were reachable in IPv6 for at least one tested application that was not reachable in IPv4; five of eight tested applications are more open over IPv6 for the routers, and six of eleven tested applications are more open over IPv6 for the servers. While 18% of routers and 17% of servers we tested were reachable in IPv4 for at least one application that was not reachable in IPv6, the applications involved can have default configurations that do not listen in IPv6. The policy discrepancy landscape overall is profoundly varied; a staggering 44% of routers and 43% of servers had different application reachability (i.e., connection success) depending on version of IP used. At a high level, this suggests a significant difference in the set of services that dual-stacked hosts effectively make available (intentionally or not) over one version of IP versus the other.

4.3.1 Router Application Openness Results

Figure 4.1 shows the protocol discrepancy we observed between IPv4 and IPv6 for the routers we probed. For each application, we show the percentage open over IPv4 and/or

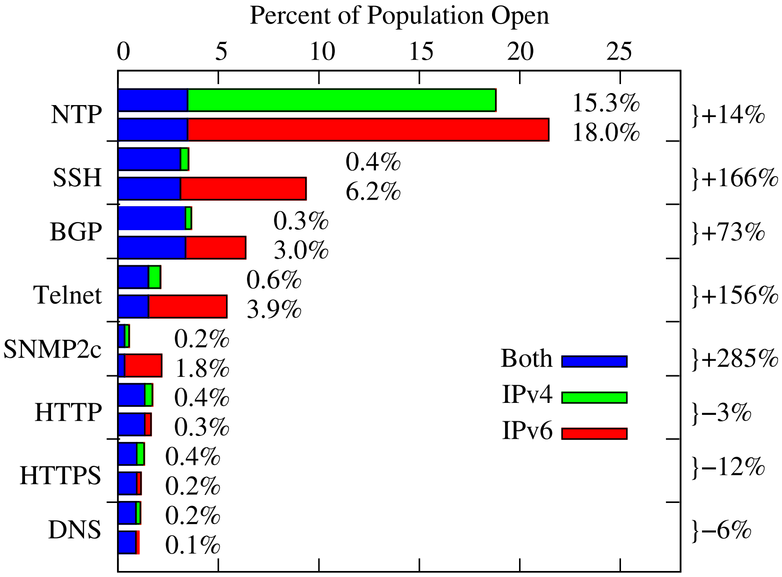


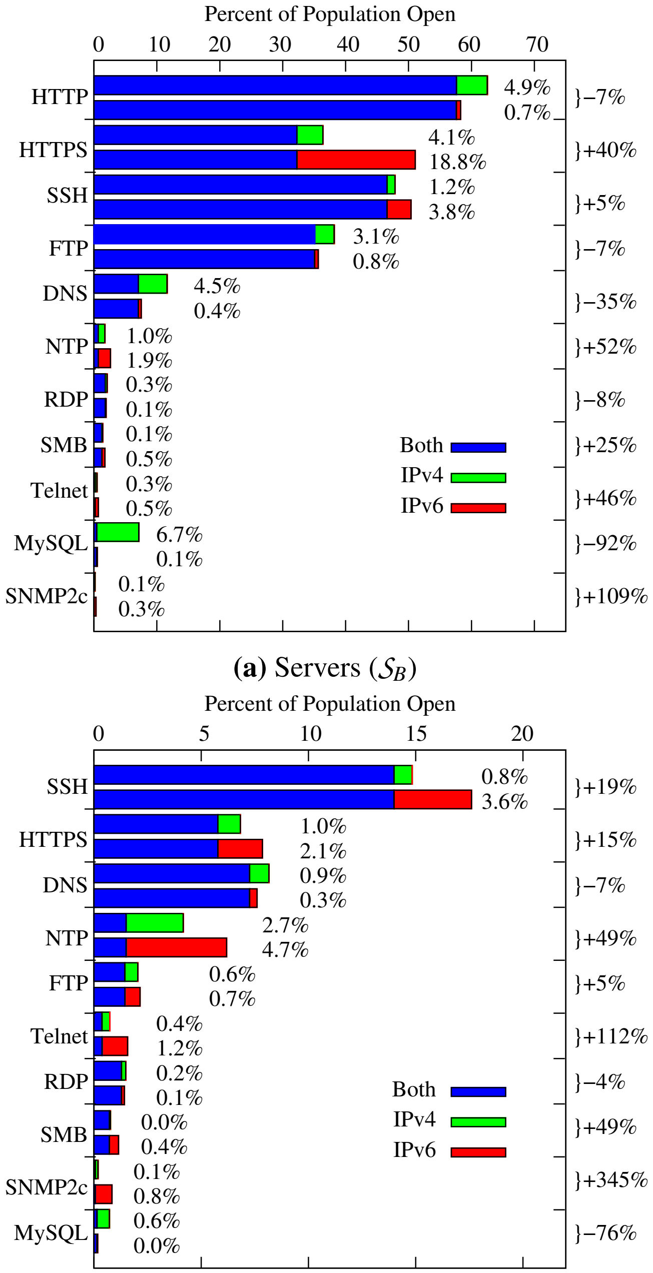
Figure 4.1: Percentage of 25K dual-stack routers ($\mathcal{R}_B$) responsive to ping that were open via IPv4 and/or IPv6 for each application tested. For each application, the green bar corresponds to reachability (connection success) over only IPv4, the red bar only IPv6, and the blue bar reachability over both. Beside each bar we report the percentage of hosts tested that were only reachable by IPv4 or IPv6, and beside each application is the percentage difference in reachability over IPv6 compared to IPv4. High value applications including SSH, Telnet, and BGP were more reachable in IPv6.

IPv6, the percentage open over just IPv4 or IPv6, and the percentage difference in openness of IPv6 over IPv4. Particularly troubling is the observation that the three most open protocols in IPv6 are high-value: SSH, BGP, and Telnet; these three protocols were 166%, 73%, and 156% more open in IPv6 than IPv4, respectively. We next discuss each application result and comment on its possible impact.

NTP: Among the applications, NTP is most open overall, but discrepancy between the two protocols is relatively moderate at 14% more openness for IPv6. The fact that NTP is the most reachable application in this dataset is not totally unexpected, given that this application is commonly enabled by default on some network devices (e.g., [1]) and left open. A surprising finding is that a relatively large percentage of the routers only respond via one protocol or the other relative to those that respond on both. This suggests some peculiarity in default router NTP configurations. While access to NTP is not a critical risk, it has been leveraged for large-scale distributed denial-of-service (DDoS) attacks in the past, and lagging IPv6 protection may signal less attention paid to blocking its traffic over IPv6 than has been deployed for IPv4 [45]. Further, we found that the NTP version command we used can leak the device vendor and version in many cases, which may be helpful to attackers targeting specific vulnerabilities.

SSH: The second most open application we see is SSH, and SSH also has the second largest discrepancy between the two protocols, with IPv6 being more than twice as open; 166% more hosts allow connecting over IPv6 than IPv4. As SSH is a management application allowing control over the device, this is a troubling finding. If exploited via brute-force password attempts, harvested passwords used by administrators on other compromised sites or hosts, or via software vulnerabilities, SSH access could lead to stealthy and large-scale attacks. These might include, for example, redirecting traffic for specific websites, email, or DNS queries to attackers, and facilitating other various forms of man-in-the-middle attacks. Further, since routers are specialized systems with typically proprietary operating systems and less general-purpose computing power, they may be less likely than servers to be bolstered with protections against a range of SSH-based attacks—e.g., by limiting password tries, enforcing SSH key-only logins and sending failed logins to syslog for alerting.

BGP: The third most open application is BGP, which we would expect to be running on



(a) Servers ($\mathcal{S}_B$)

(b) Servers ($\mathcal{S}_B$) unresponsive over port 80

Figure 4.2: Percentage of 520K dual-stack servers responsive to ping that were open via IPv4 and/or IPv6 for each application tested (figure 4.2a) and that of 137K (37% of all) servers that were not responsive to HTTP (figure 4.2b). Seven of the eleven applications tested are more open in IPv6, including the security-critical SSH, SNMP, SMB, and Telnet services. The subset of servers that are not HTTP servers are more open than the general server population, which may be due to web load balancers on the http servers skewing results for overall population.

routers, but not to necessarily be open for anyone to connect. An open BGP port on routers leaves them potentially more susceptible to various TCP-based attacks, such as SYN floods, and blind in-window attacks [123]. The fact that 73% more hosts completed the TCP handshake over IPv6 than IPv4, suggests, at the very least, that some additional protection, likely via an access control list, has been set up on these devices for IPv4 but not for IPv6. Hence, the deployed security policy on these routers for IPv6 contradicts their IPv4 policy. As routers constitute the backbone of the Internet, and BGP is the protocol by which Internet routers communicate where to send traffic, vulnerabilities in BGP pose a serious threat. If attackers can disrupt routes via TCP-based attacks, or, worse, combine the open port with BGP or router-specific vulnerabilities to hijack routes, they would be able to redirect traffic, enabling wholesale man-in-the middle or denial of service attacks.

Telnet: The fourth most open application is Telnet. We were surprised to discover so many routers accept global TCP connections to Telnet at all (9% of the dataset over any IP version), given the fact that this application has been replaced by SSH as a primary management interface for routers, in large part due to its inherent insecurity. This insecurity stems mostly from the fact that Telnet sends traffic unencrypted and that, unlike SSH, it also has no means of validating the identity of the server that a client connects to (which an SSH client can do by checking the fingerprint of the key that the server provides during connection). Moreover, beyond server authentication, there is no key-based authentication for clients in Telnet either; so, all connections involve sending a user name and password over the wire (in clear text)—to a sever whose identity can not be verified prior to doing so—making the application traffic susceptible to being eavesdropped for password harvesting, for example, or other attacks. Router Telnet sessions have even been targeted by nation states to capture the configuration of routers, leading to deeper network breach [70]. As with SSH, the danger of weak passwords that can be brute-forced and the possibility of shared passwords across sites allowing compromised credentials to be used to gain broader access, mean that the security posture of these devices is degraded simply by having Telnet exposed. As there are again more than double—156% more—IPv6-open routers with Telnet exposed, the deployment of IPv6 here has markedly reduced security in this sample of routers.

SNMP: We attempted SNMPv2c requests over UDP/161 for the sysName.0 MIB using the common default *public* community string. Three percent of the routers responded with data. We did not attempt to use the common *private* community to alter configuration on the system, nor did we collect any data from the device. However, when we did our follow-on probes for signature matching described in § 4.2, we additionally performed SNMP gets for the sysDescr.0 MIB, which allowed us to aggregate operating systems and versions, the large majority of which reported being Cisco. While the read-only public community may itself not necessarily pose a catastrophic risk to the device, it may be used to leak version information about the device, find weaknesses in configuration, or gather information about connected devices; all of which can be useful reconnaissance for attackers, especially when paired with published vulnerabilities. Furthermore, the fact that these devices expose SNMP for nearly four times—285% more hosts—over IPv6 than IPv4 suggests that many operators took steps to block this management application over IPv4. If these operators are relying on access-lists, firewalls, or other port filtering for protection of SNMP but keeping the default community strings in place, it is likely that this population of routers could be reconfigured using the private community over IPv6, a much more direct and immediate threat than that posed by the read-only probe we attempted for ethical reasons, as such, we consider this a serious vulnerability.

HTTP and HTTPS: The two web application protocols were not very common on the routers in our dataset, and these were the first to break the pattern of greater IPv6 openness, with each slightly more closed for IPv6 than IPv4 (-3% and -12%, respectively). For routers with web-based management enabled, this means security is probably no worse under IPv6 than under IPv4. There were, however, a small handful of hosts where access was only allowed over IPv6 (78 routers for HTTP and 51 HTTPS), suggesting, perhaps, at least some cases where IPv4 access was blocked but similar blocks were not put in place for IPv6. As embedded web-based management applications are notorious for vulnerabilities, this capability is rarely used by professional router operators. Having an unknown web-based attack vector over IPv6 enabled, even for this small number of hosts is problematic. Though, fortunately, the scale here is small.

DNS: Like HTTP and HTTPS, DNS was less open for IPv6 compared to IPv4, though, again, a small handful of routers (35) only responded to DNS over IPv6 and not IPv4. Aside from application-specific vulnerabilities (e.g., BIND CVEs) that might impact the device if DNS is exposed, the other notable security implications of having DNS open to the public when policy would dictate otherwise has to do with either leaking internal-only DNS records or facilitating DDoS attacks based on DNS. Overall, however, policy for the two IP versions was most similar for DNS.

Overall, the baseline protocol differences we found in this population are troubling. The fact that more than a quarter of routers had at least one application accessible over IPv6 that was closed over IPv4, including some high-value application ports for attackers, means that the routers in our sample are generally more vulnerable under IPv6 than IPv4 (at least on the tested common applications). Since network operators are at the forefront of understanding and deploying IPv6, this is somewhat surprising. We conjecture that network hardware may be subject to less security audits and scrutiny than servers are, although it is also possible that as router operators usually deploy IPv6 (naturally) before the server operators that rely on it do, they may be doing so under either greater time pressure or with fewer existing institutional tools and processes for assuring consistent security policy on routers.

4.3.2 Server Application Openness Results

Figure 4.2 shows the protocol discrepancy we observed between IPv4 and IPv6 for the servers we probed (dataset $\mathcal{S}_B$). As with the router set in figure 4.1, the general pattern is for a more open security policy in IPv6, with HTTPS, SSH, NTP, Telnet, SMB, and SNMP more open. The overall discrepancy we find in the server list between IPv4 and IPv6 is much smaller than in the router data, relatively speaking. However, the sample size is twenty times larger. Thus, in absolute terms, even the smaller differences between IP version found in this dataset translate to thousands of potentially inadvertently exposed systems.

Because characteristics of the server dataset are heavily influenced by the overwhelming presence of HTTP servers, we examine the server dataset in two dimensions: all responsive servers (figure 4.2a) and the 191K (37%) server hosts that did not respond on port 80 for both IPv4 and IPv6 (figure 4.2b). Other than for NTP—which is nearly flat, going from 52% to 49% more open in IPv6, and HTTPS—which drops from 40% to 15% more open in IPv6, mostly due to the elimination of a single large hosting provider's servers—the fraction of hosts for which IPv6 is more open than IPv4 increases for every tested application in this non-HTTP subset. For instance, SSH's openness in IPv6 relative to IPv4 increases from 5% to 19% (4x), Telnet jumps from 46% to 112% (2.5x), SMB from 25% to 49% (2x), and SNMP from 109% to 345% (3x). These results suggest that dual-stacked non-web servers generally have more policy discrepancy and, thus, apparently more IPv6 vulnerability than the overall dual-stacked server population suggests. As with the router list, we next discuss each application result and comment on its possible impact. Our intuition behind examining non-HTTP-responsive servers separately stems from the fact that we believe these servers are less likely to be behind load balancers or IPv6 gateways (e.g., as offered by CloudFlare [2]). Since these load balancers and gateways generally do not forward non-web traffic to the actual server behind them and since they may terminate the IPv6 connection (in the case of gateways, that is, in fact, their function), they are much less likely to show IPv6 capability on non-web ports. Thus, looking at the non-web subset of servers may be more indicative of the *typical* configuration of the servers actually providing the content or service.

HTTP and HTTPS: HTTP was found to be less open on IPv6 than IPv4 by 7%, but there were 3.5K servers not reachable over IPv4 that were reachable over IPv6. Since it is unlikely that dual-stacked public websites would purposefully allow only access via IPv6, it is possible some of these servers are hosting confidential or otherwise content not intended for public view. With respect to HTTPS, we did find a large percentage of servers (19%) only reachable over IPv6. Digging deeper into this peculiar group, we found that 94% of these IPv6-only HTTPS servers (92k hosts) belong to a single large European hosting provider. Of the hosts operated by this provider, 99% have HTTP open on both IPv4 and IPv6, while HTTPS is only served for IPv6. We contacted this provider but did not receive a response and, thus, have no explanation as to the intention behind this configuration.

SSH and Telnet: Both remote terminal applications were more open for IPv6, at 5% and

46% (respectively) in the overall server set, and 19% and 112% more open for IPv6 in the non-webserver set. Although the policy mismatch percentages are more modest than for routers, in absolute terms 20k servers were only reachable on SSH via IPv6 (versus only 6.5K that were reachable by IPv4). In addition, the non-webserver set shows a more worrying openness pattern, perhaps as a result of these systems having a more varied role, or our probes not being dropped by intermediate gateways. For Telnet, 2.5k were only reachable over IPv6 and 1.4k only over IPv4. This means that 23K servers could be vulnerable to brute-force password or server vulnerability exploits that were protected via IPv4. Digging a bit deeper at cross-application groups, we were curious if the IPv6-open SSH servers were more likely to also be open on Telnet, as it is used similarly to SSH and may also be neglected to be blocked by the same operators. Indeed, it appears that a disproportionate 7.3% of these SSH servers were also open to responding over IPv6 on Telnet (versus 0.5% in the overall sample that had Telnet open for only IPv6, regardless of SSH status). As SSH brute-force scanning is highly prevalent in IPv4 [106], it is reasonable to assume that such attacks over IPv6 are on the horizon. While random address scanning may not be common in IPv6 (though, see § 4.7), once a hostname for a dual-stacked server is discovered somewhere, brute-force password guessing against that server over IPv6 is feasible. Since these 20 thousand servers are running SSH but have blocked it on IPv4, they may be less likely to utilize other SSH security measures (such as disabling the root account, or disabling password-based login in favor of key-based authentication).

SMB and RDP: The Server Message Block (SMB) application layer protocol is generally used by Microsoft Windows Clients and Servers for file and printer sharing, as well as serving as an inter-process communication layer. Over the years, it has been an attack vector for numerous vulnerabilities, and has been exploited by worms, including Sasser and Conficker [142]. It is often on the Internet Storm Center's list of top-10 most scanned ports [154]. As such, this port is often treated as internal-only by operators and is commonly the subject of filtering policy [155]. The remote desktop protocol (RDP) is also built into Windows servers and clients and allows remote console access to the systems. While this application does not have as deep a history of exploits as SMB, it does provide management access to Windows systems; thus, as with SSH and Telnet, if it is exposed to connections from the public network, the potential exists for brute-force or other exploits that can lead to system compromise. In our analysis, SMB was found to be open on 25% more hosts via IPv6, exposing a total of 2.4K hosts that have it blocked on IPv4 in the overall server population. In the non-webserver population, 49% more hosts were reachable only via IPv6. While RDP is less open on IPv6 than on IPv4, we did see nearly 700 servers with this port open for IPv6 where it was closed on IPv4. As both of these ports are attractive targets for attackers, these several thousand servers that allow connections over IPv6, where likely the policy intention was no access, are a concerning discovery.

DNS and NTP: Open DNS resolvers are problematic for two reasons. First, open resolvers can be susceptible to cache poisoning attacks [108, 159]. These, in turn, leave the users of subverted resolvers vulnerable to being re-directed to nefarious services. Second, open resolvers are susceptible to being leveraged in reflection and amplification DDoS attacks (e.g., [124]). The DNS port on servers is less open via IPv6 than IPv4. Again, however, a small fraction of servers, numbering 2.3K, were found reachable via only IPv6. We also found 52% more servers allowing NTP queries via IPv6 compared to IPv4. This means roughly 10K additional servers—that return system and version information—can be used as DDoS amplifiers [45] or for reconnaissance to gather version and system information about the servers. While weaker threats, both DNS and NTP have had vulnerabilities reported as well as been used to attack others in DDoS campaigns, and so, IPv6 security on both should be brought in line with IPv4.

FTP: We found FTP to be slightly less open (7%) in IPv6 than IPv4 in the overall server population tested, though more open (5%) in the non-webserver population. For IPv6, there were a small number (3k hosts, 0.8%) only allowing FTP connections over IPv6 (and 3.1% only on IPv4). Interestingly, FTP's prevalence in the webserver set is much more closely correlated with being on an HTTP server. For the fraction that were open in IPv6 where IPv4 was blocked, these could represent a back door to web content, including source code to websites.

SNMP: Although the absolute numbers were low for SNMP among servers, we found 109% more of them (1.6K) to respond over IPv6 than IPv4; in the non-webserver population, a staggering 345% more systems were open over IPv6 where the IPv4 application was blocked. This may be a source of reconnaissance for attackers or may indicate that the default read/write *private* community is also open on these servers (which, again, for ethical reasons, we did not test). As such, it is concerning that an additional almost two thousand servers may be probed (and possibly manipulated) over IPv6 via SNMP.

MySQL: Finally, we probed servers for the MySQL server port, and found that only 0.5% supported IPv6 at all. MySQL prior to version 5.5.3, released in mid-2010, did not support IPv6. Current versions of the database support IPv6, but IPv6 was not enabled by default, even on dual-stacked hosts, until version 5.6.6—first released in mid-2012 [134]. In fact, when we analyzed the MySQL minor version strings returned by 32K servers that responded to our banner grab, as described in § 4.2, 26% were running versions that did not even support IPv6, 66% were running versions with IPv6 disabled by default, and just 8% were running versions where IPv6 was supported and enabled by default. The fraction of servers responding to MySQL that responded on both protocols or IPv6 was just 0.5%. In absolute numbers, nearly 600 servers responded on IPv6 only, while 2.2K responded on both protocols and 35K responded on IPv4 only. Similar to FTP access, MySQL access is correlated with presence on a webserver, suggesting a reliance on a database system that is needlessly exposed to the Internet. In fact, since databases are typically run as back-end services to web sites or internally in organizations, the fairly high number of globally reachable servers was surprising, and the several hundred apparently reachable by IPv6 only, though relatively few, is concerning.

Overall, the server dataset showed smaller discrepancy percentage-wise between IPv4 and IPv6 port filtering policy for the applications we tested than we found in the router probes. However, as we noted, there were some high-value applications that were more open, and, due to the substantially larger population, the raw numbers of servers open on IPv6 only for many applications is of concern. In many cases, brute-force attacks are enabled by this discrepancy, and in other cases, known vulnerabilities in software may be exposed on thousands of dual-stack servers whose operators may believe that they have no exposure to these threats due to their IPv4 filtering.

4.4 Policy Uniformity

4.4.1 Network Response Uniformity

The results we present above show a difference between IPv6 policy and the *intended* policy, as indicated by the IPv4 policy. We next turn to understanding whether this discrepancy is a symptom of an organization's overall security posture or due to small scale misconfiguration that deviates from the organization's intended policy. Therefore in this section we aggregate results for each organization—at both routed prefix and autonomous system (AS) granularities—and assess policy uniformity.

We aggregate hosts in our $\mathcal{R}_T$ and $\mathcal{S}_T$ datasets by routed prefix and AS based on BGP table data collected by RouteViews and the RIPE RIS BGP collector on February 1, 2015. We find that the IPv4 and IPv6 addresses for a host are in the same AS in 94% ($\mathcal{R}_T$) and 95% ($\mathcal{S}_T$) of the cases. Therefore, for simplicity we label the hosts with their IPv4 routed prefix and AS number. Further, we label each host and service with the IP-level protocol(s) that allow a connection. Hosts with multiple IPs are labeled with the majority outcome. When a given service is unreachable via both versions of IP we exclude it from further analysis because we cannot determine whether the service is not reachable due to policy or simply not running and therefore these cases provide no insight into policy. For each service on each host we are left with one of three labels: "4" for services that are only reachable via IPv4, "6" for services that are only reachable via IPv6 and "B" for services that are reachable via both IPv4 and IPv6. Given these labels we define the uniformity for each service within the organization—defined by routed prefix or AS—as the fraction of hosts with the most common label for the service. For example, consider an organization with five devices running DNS, three of which are labeled "B", one labeled "4" and one labeled "6". The uniformity is therefore 60%. Table 4.3 shows the mean and median number of devices we detect in each organization for our datasets.

To put our uniformity results in perspective, we compare with a "pseudo network" which is made up of a random selection of hosts—regardless of network boundary—of the same size as the median organization size given in table 4.3. We compute uniformity

Table 4.3: Number of devices within an organization.

Dataset	Aggregation	Mean	Median
Router	Routed Prefixes	20	5
	Autonomous Systems	40	5
Server	Routed Prefixes	52	6
	Autonomous Systems	133	8

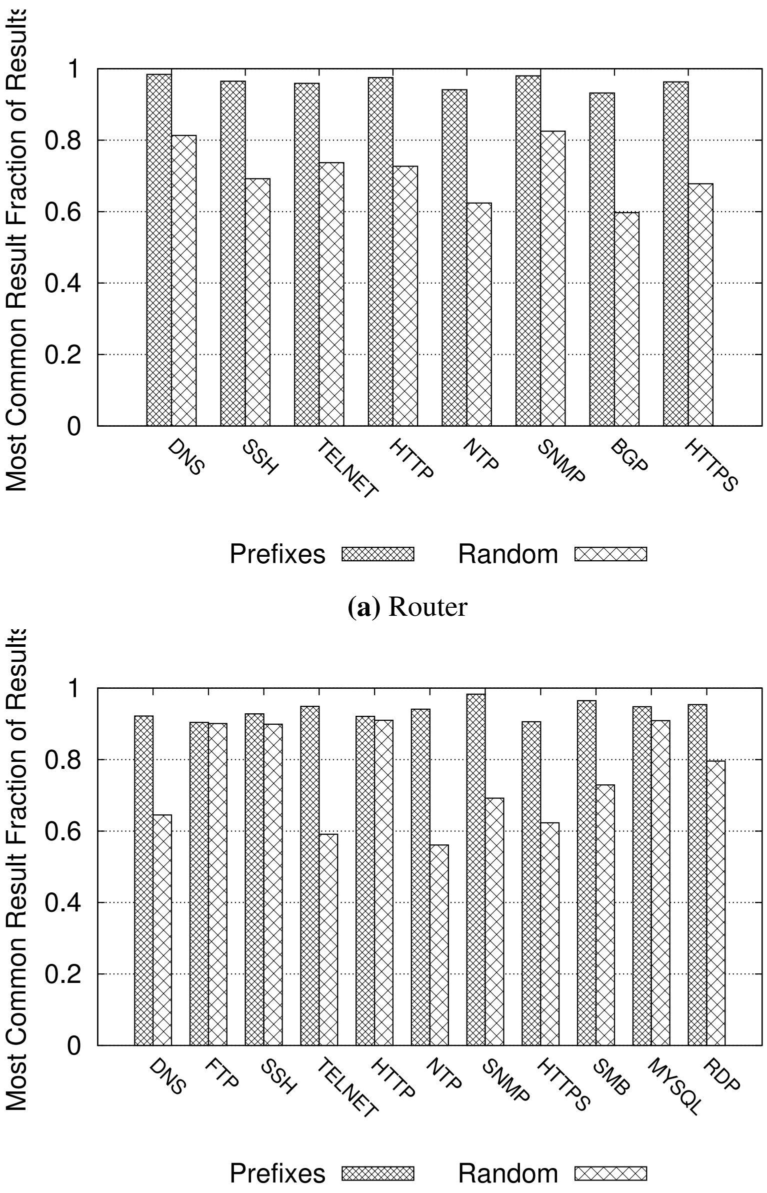
across the randomly chosen pseudo network just as we describe above for hosts within a well-defined network boundaries (routed prefix or AS). For each application we calculate the mean uniformity across 1,000 random pseudo networks.

Figure 4.3 shows the mean uniformity results for both routed prefixes and random pseudo networks for both datasets. First, we find at least 90% mean consistency within organizations across applications. This indicates that the disparity we detect between IPv4 and IPv6 policy is not driven by one-off misconfigurations, but is in fact a systematic difference in policy deployment.

Additionally, the figure shows—across datasets and applications— higher uniformity within actual organizations than within randomly selected pseudo networks. This strengthens our conclusion that we are detecting in-situ policy differences and are not being lead astray by small, but broad misconfigurations. Also, we elide the results for organizations defined by AS for clarity, however note: (*i*) the uniformity is generally lower for AS-based organizations than routed prefixes due to the increased aggregation across difference administrative domains, and (*ii*) just as with routed prefixes the uniformity is greater for actual organizations than for pseudo networks.

4.4.2 Intra-protocol uniformity

We next tackle an issue related to organization-level uniformity: host-level uniformity. That is, how uniform are individual hosts for the same version of IP across addresses? This question is important for two reasons. First, if policy differs for hosts across different addresses via the *same* protocol, it may not be surprising that there are differences between IPv4 and IPv6. Further, non-uniformity at the address level could indicate ad-hoc policy applied at individual machines as opposed to systematic policy applied at the organizational

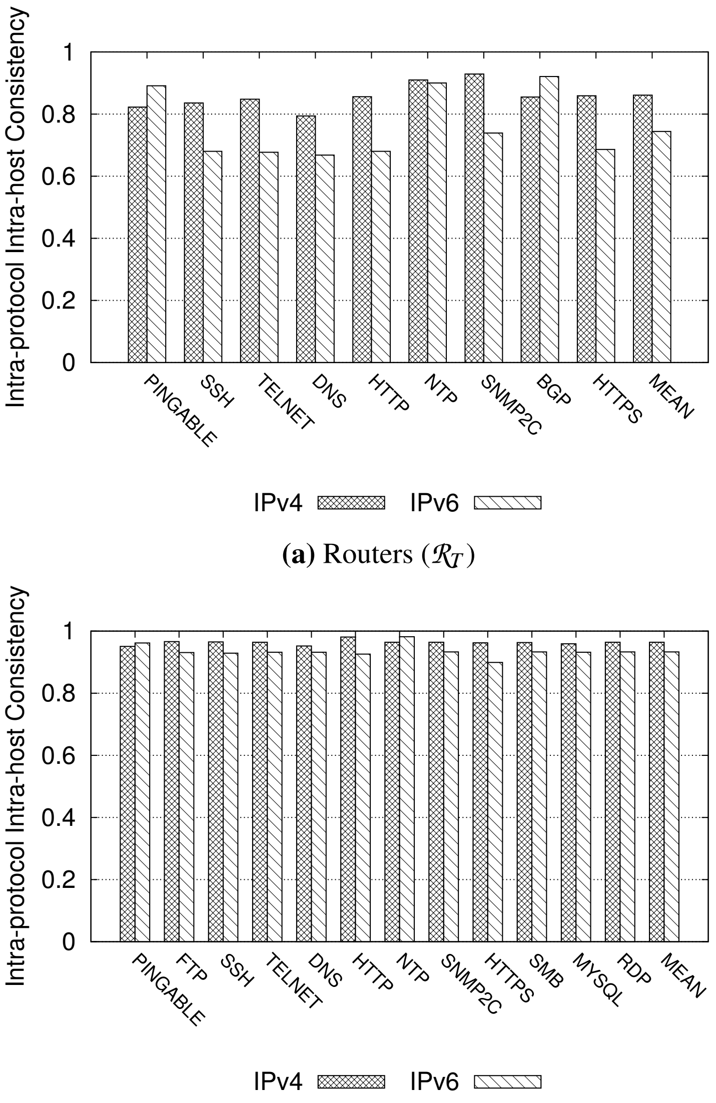


(a) Router

(b) Server

Figure 4.3: Average organization uniformity for server router ($\mathcal{R}_T$) and server ($\mathcal{S}_T$) dataset compared to the average pseudo-network of same median host count (each randomly selected from population of host results). The uniformity is more consistent within network boundaries than within random groupings.

border. Second, intra-protocol policy uniformity speaks to the maturity of security controls and on average, is useful in comparing the difference in maturity between protocols. For example, if we find IPv4 to be more consistent than IPv6, this is an indication that security controls for IPv4 are more mature, tested and robust than for IPv6.



(a) Routers ($\mathcal{R}_T$)

(b) Servers ($\mathcal{S}_T$)

Figure 4.4: Average intra-IP version uniformity within hosts having more than one IP of the given version. We see that results are more consistent for IPv4 than IPv6, and more in the server ($\mathcal{S}_T$) dataset than the router ($\mathcal{R}_T$) dataset. Also, we show that the fraction of addresses that are ICMP-pingable when multiple addresses are associated with the same host is higher for IPv6 than IPv4.

We calculate the uniformity across each host and IP version and present the mean uniformity across hosts in figure 4.4. In addition to per-application results, we also show two additional sets of bars: (*i*) the overall mean across all applications and (*ii*) the mean uniformity for ICMP ping. The plots first show that the host-level uniformity is higher for servers (90–95%) than for routers (70–90%). One possible reason for this discrepancy is router IP addresses identify individual interfaces, which have different tasks (i.e., peering with different networks). Therefore, it may be natural to find different policy applied to different interfaces. On the other hand, servers do not have the same sort of natural per-IP division of labor and therefore show higher uniformity across addresses.

Also, for the $\mathcal{S}_T$ dataset we find approximate parity between IPv4 and IPv6 in terms of uniformity across applications. This is in contrast to the $\mathcal{R}_T$ dataset where we generally find higher uniformity in IPv4 compared to IPv6. There are two exceptions where IPv6 is more uniform than IPv4: BGP and ICMP. While we cannot readily explain BGP's disparity, the ICMP difference may stem ICMP being less strictly filtered in IPv6 due to a deployment requirement for IPv6 (e.g., [66]).

4.5 Blocking Enforcement

Having established that myriad filtering occurs, we next turn our attention to the mechanisms employed to block traffic and where those mechanisms are implemented. To measure this, we use traceroute probes, as described in § 4.1.2, for both routers ($\mathcal{R}_T$) and servers ($\mathcal{S}_T$). For each application, and each address associated with each host, we first determine whether the application is open or closed. For each closed application we determine the enforcement mechanism. As we note in the last section, we do find cases where policy differs within the same IP protocol. In these cases, we label the host based on the majority enforcement mechanism. We classify each attempt, as follows.

- ***Open***: In this case, the target host responds favorably (i.e., with a TCP SYN+ACK or a UDP response).

- ***Passive:Target***: In this case, the target host silently drops the SYN or UDP request. We detect this by observing that the last responding host within the traceroute is the hop immediately prior to the target host (as established by ICMP-based traceroutes and/or traceroutes involving other applications).

- ***Passive:Other***: In this case, we find a hop in the path prior to the hop before the target host is the last hop to respond to the traceroute. Therefore, we conclude that a firewall is silently dropping the traffic before arriving at the target host.

- ***Active:Target***: In this case, the target host actively responds to our SYN or UDP request with an error indicating the service is not available (e.g., TCP reset or ICMP error message).

- ***Active:Other***: In this case, a device on the path towards the target host issues an active ICMP error or TCP reset that indicates the service is not available.

Note that firewalls typically simply drop undesired traffic silently without generating error traffic (i.e., fall in the "Passive:Other" category). Closed ports on hosts are more prone to generating an active error message (i.e., "Active:Target"). Thus, the breakdown between our various categories can shed light on firewall, access control list, or other similar policy enforcing device in the path to the target.

Table 4.4: Connection failure mode distribution differences. We observe that connection failures are more frequently active for IPv6 than for IPv4 in both datasets, suggesting fewer silently-dropping policy devices in IPv6.

	Router ($\mathcal{R}_T$)		**Server ($\mathcal{S}_T$)**	
Failure Mode	**Mean IPv4**	**Mean IPv6**	**Mean IPv4**	**Mean IPv6**
Open	4.17	6.04	18.57	18.89
Passive:Target	43.50	27.15	36.06	31.17
Passive:Other	10.12	15.82	16.31	14.20
Active:Target	30.93	36.14	22.82	27.61
Active:Other	3.55	6.94	2.09	2.79

4.5.1 Typical Connection Failure Modes

Table 4.4 shows the average breakdown across applications of our dataset into the categories we give above for IPv4 and IPv6. We first find that across dataset and IP version host-based policy enforcement accounts for the majority of the cases where traffic is filtered (i.e., the ":Target" categories). Additionally, silent dropping by the network accounts for 10–15% of the filtering cases, while active errors are relatively rare for the network to generate (2–7% of the cases). However, there are also differences between IPv4 and IPv6. For instance, IPv4 shows Passive:Target is more prevalent than Active:Target in the router dataset, but IPv6 shows the opposite. Further, an active error message is more likely in IPv6 than in IPv4—perhaps showing that traditional border firewalls are silently discarding unwanted IPv4 traffic and not yet dealing similarly with IPv6 traffic. At a high level these results show that even when both protocols implement the same high-order policy to block some service, they are not always doing so in the same manner.

While we do not show individual applications' connection failure distributions, one interesting outlier to mention in the router dataset is NTP. We find NTP is five times more likely (24% versus 5%) to respond with an active error in IPv6 than in IPv4. Given the widespread IPv4 NTP DDoS spike and subsequent operator mitigations reported in recent years [45], we might expect silent dropping of NTP to be a prevalent security posture. However, our results suggest that while this is a reasonable expectation in IPv4, sadly this mitigation is not as extensive in IPv6.

4.5.2 Connection failure Locations

A second aspect of policy enforcement is the *location* of the filtering. In the last subsection we started to address this by detecting whether policy is applied on the target host or by some on-path network element prior to the target host. In this section we analyze those cases where policy is being enforced by the network and not the host. To address this question, we analyze where in the path between our measurement probe and the target hosts the policy is being applied. For each failing traceroute probe not ascribed to the target host, we extract the difference in the hop count between where we know that target host to be—as established via successful ICMP echo and open application responses—and the final response from the non-target. This response could be either an active error message (Active:Other) or a normal traceroute (TTL expired) probe response in the case of a silent drop (Passive:Other). Figure 4.5 shows the fraction of these responses at each hop distance prior to the target host.

First, we find that the differences between IPv4 and IPv6 drop distance distributions are generally small ($< 6\%$) for servers at each hop distance. Further, for servers the lion's share of drops for both IP protocols happen two hops away from the target—with 49% of

IPv6 drops and 55% of IPv4 drops. This suggests that, when policy is applied along the path to a server and not at the server itself, it is likely to be applied at the same point for both protocols.

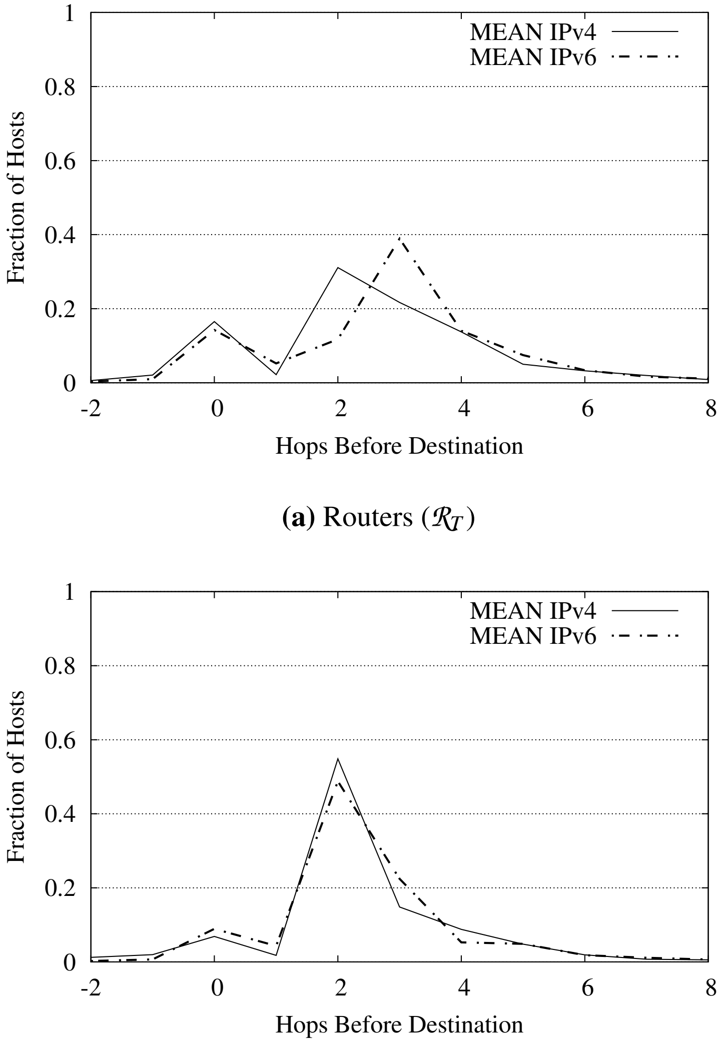
For routers, the difference in distance distribution between IPv4 and IPv6 connection failures was greater (up to 20%). IPv6 drops are most likely to happen at a distance of three hops away. IPv4 on the other hand is most likely to see drops two hops from the target host. The distribution at earlier and later hops shows rough parity between IPv4 and IPv6. In sum, although the differences between IPv4 and IPv6 enforcement location are not stark in general, we did find some differences which, when combined with the connection failure mode distributions we show above, lead us to conclude that the deployment of policy enforcement mechanisms, both in number, kind, and to a lesser extent, location, differs measurably between the two protocols.

4.6 Case Studies / Validation

Table 4.5: Validation summary. Twelve of 16 contacted operators of various types responded, and each indicated that the discrepancy was unintentional. Ten took steps to remediate.

Operator	Host-App Pairs w/Only IPv6 Open	Response
Global CDN 1	3	✓
Tier1 ISP 1	498	
Global Transit Pro. 1	201	✓
Large Hosting Pro. 1	<800	
Large University 1	5	✓
Large University 2	6	✓
Large University 3	989	✓
National ISP 1	4757	✓
National ISP 2	89	
Research/Ed. ISP 1	1	✓
Research/Ed. ISP 2	523	✓
Research/Ed. ISP 3	77	✓
Research/Ed. ISP 4	17	✓
Small Hosting Pro. 1	17	✓
Small ISP 1	12	
Small Transit Pro. 1	2	✓
	Total Contacted	16
	Total Responded	12
	Total Confirmed	12

We solicited validation on our methods and our findings from 16 networks for which we had contacts. These networks were varied in their types, ranging from access, transit,



(a) Routers ($\mathcal{R}_T$)

(b) Servers ($\mathcal{S}_T$)

Figure 4.5: Fraction of hosts (mean across all applications in dataset) where failure response (*Passive:Other* or *Active:Other*) originated the given number of hops prior to target.

university, content, and hosting networks, and varied in their location footprints, ranging from Asia, Europe, Oceania, and North America. For each network, we emailed our raw findings in text format, listing IPv4/IPv6/name tuples with associated information on which ports were apparently blocked in IPv4 but were not blocked in IPv6. We received responses from twelve networks, summarized in Table 4.5. In every case, we received a confirmation of our hypothesis that the underlying cause was an oversight on consistent application of security policy. In addition, ten of the twelve responding networks immediately worked to establish a congruent policy in IPv6.

When we followed up with individual operators, we found that policy was typically being applied on the individual devices. One operator had used IPv4-specific examples for how to harden the control plane of a router, without adding additional configuration to accomplish the same in IPv6. Another operator had an organization-wide standard security policy for IPv6 that was found to not be applied to a single device; this device was installed as a IPv4-only system, and had IPv6 later added. The organization had been working to ensure their IPv6 posture was on par with their IPv4 posture, though the firewall configuration tool their system administrators had been using does not have an IPv6 option, leaving a lot more manual work for them. Similarly a large transit provider confirmed that they did not intend for external ssh and telnet access for their routers in IPv6. They deployed configuration on the routers to prevent external access in IPv6, but were not able to deploy the same configuration on customer routers that used their address space to number the interconnection links with their customers. We found most transit providers do not block packets headed towards their customer's interfaces.

Shortly before this chapter went to press, we took additional steps to send emails to abuse contacts for 396 remaining autonomous systems not associated with the 16 operators above, whose routers were also found with open IPv6 access in $\mathcal{R}_T$, as we deemed the threat to routers to be of greatest urgency.

4.7 Scanning Feasibility

As we have shown, IPv6 often provides access to application ports that are unreachable via IPv4. This in turn provides attackers with a path to vulnerable services. However, an attacker must first find these hosts and services before attempting to exploit vulnerabilities. Within IPv4, the most straightforward method is to scan the entire address space for vulnerable services. Current scanning techniques allow a single host connected to a fast network to scan the entire IPv4 address space in less than one hour [60]. Scanning the IPv6 address space in this fashion would take on the order of 10^{22} years. The task is prohibitively expensive, even considering parallelizing the work and assuming massive network capacity improvements. Alternatively, an attacker could leverage the sorts of DNS and traceroute data we start with to form a hit list for scanning. Although this is useful to obtain a sample that is suitable for understanding the general policy posture of the IPv6 network, it is far from comprehensive.

While scanning each IPv6 address is impracticable, some researchers note that the feasibility of targeted IPv6 scanning depends on the device addressing strategy within each block [34]. When operators concentrate devices in a contiguous sub-block of a routed prefix, attackers can concentrate on the sub-block and ignore everything outside—potentially putting the task of comprehensive scanning of devices within reach. Random address assignment within a routed prefix may at first appear as security-through-obscurity, but the strategy actually determines whether IPv6 brute-force scanning is practically possible. As an example, 2008 work by Malone showed that significant fractions of the host ID portion of IPv6 addresses were derived from the MAC address using the EUI-64 mechanism [126]. This addressing strategy effectively reduces the search space for an attacker from 64 bits to 48 bits—and even further down to 24 bits if the Ethernet card vendor is known or can be guessed. Further, Malone notes a prevalence of low-integer host IDs. While EUI-64 is less common in today's networks, we are still interested in whether current address assignment strategies impact an organization's security posture.

Therefore, we next turn to using the addresses found in our $\mathcal{R}_T$ and $\mathcal{S}_T$ datasets to understand the addressing practices of operators. We first note that the high-order network ID portion of routed IPv6 networks is obviously advertised in BGP—often with a prefix of /48 or longer. These are available in public routing table repositories and serve to significantly winnow the scanning space an attacker would have to cover for a comprehensive scan. After the prefix, the natural next question is whether the middle (subnet) portion of the IPv6 address—typically 16 bits—is random. In our target address data, we find that 47% of the router and 45% of the server subnets use only the lower 8 bits. Additionally, 8% of router subnets and 19% for servers use a reverse-low pattern, where the high-order four bits are used and the remainder of the bits are zero, resulting in 15 possible subnets. Thus, just 270 possibilities account for 55% of router and 64% of server subnets. Scanning this small fraction (0.4%) of the theoretical 65,536 possible subnets would identify the majority of used networks.

Table 4.6: IPv6 Interface Identifier (IID) types for all IPv6 addresses for hosts in the $\mathcal{R}_T$ and $\mathcal{S}_T$ datasets, including 30K router and 968K server IPv6 addreses. For each dataset, we show the percentage in that category and a cumulative total. We find that 89% of router and 37% of server addresses are within very low ranges, allowing discovery within seconds on a subnet. Recall that half of the bit space (i.e., 32 bits) is a minute fraction of the address space—$2^{32}/2^{64}$ or 1/4,294,967,296

		Router		**Server**	
IID Bits Used	IID Value Range	%	Cum. %	%	Cum. %
1	<= 0x0001	23.74	23.74	5.83	5.83
4	<= 0x000F	37.89	61.63	5.94	11.77
8	<= 0x00FF	6.87	68.49	4.76	16.53
16	<= 0xFFFF	11.00	79.50	5.50	22.03
32	<= 0xFFFF FFFF	9.81	89.31	14.50	36.53
EUI-64	Middle == 0xFFFE	0.92	90.23	4.92	41.45
Other	Not in Above	9.77	100.00	58.55	100.00

Finally, we turn to the host ID portion—low-order 64 bits—of each address. In table 4.6 we classify each address into one of several allocation ranges based on use of a decreasing number of leading zeros or use of the EUI-64 scheme. We find nearly a quarter of routers and 6% of servers use the value of 1 as the host id, and that scanning just the lower quarter—16 bits—of the theoretical host ID space will identify 80% of the open routers and 22% of the open servers. The address assignments are therefore extremely concentrated and an attacker could get significant coverage at a miniscule fraction of the cost of a full scan ($2^{16}/2^{64}$ or just 4×10^{-15} of addresses). Further, we find EUI-64-derived addresses in just under 1% of the routers and nearly 5% of servers. We also find that the 24-bit vendor ID portion of the host ID shows eight vendors account for 46% of the routers and 69% of the servers. This reduces the search space to the low-order 24 bits, which, even with random assignment, is tractable to scan.[3] A 1 Gbps (1.4 Mpps) scanner [60], could scan all of the categories in table 4.6 except for "Other" and including only the top eight most common EUI-64 vendor IDs on any given subnet in 53 minutes. In our dataset this would identify 90% of routers and 40% of servers at a minuscule fraction of the cost of scanning a full IPv6 64-bit address block at that rate (418 thousand years). Given these numbers and the

[3] MAC addresses are often assigned sequentially as the network cards are manufactured. Thus, they are not uniformly random, and one can expect to find less entropy within large organizations in EUI-64-derived IP addresses [77].

addressing schemes we saw, brute force scanning for servers and routers, while not exhaustive or foolproof, is still largely feasible for enumerating the majority of IPv6 hosts on a subnet. With prefixes easily identifiable and most subnets using just one of 270 values, we conclude that scanning is still a viable way to identify large fractions of hosts within networks, even if complete scanning of the IPv6 address space is impracticable. Thus, our main findings reporting greater openness in IPv6 may be exploitable not just by hostname but also via brute force scans, especially if they target a single network prefix. One word of caution for researchers interested in applying scanning as a technique for brute-force measurement in IPv6: there is a known severe denial of service condition that can be triggered in many older or improperly configured IPv6 routers due to memory exhaustion from incomplete neighbor discovery entries (see e.g., [5]).

4.8 Discussion

Our experiments probing 25K routers and 520K servers on commonly attacked ports showed that, for both routers and servers, 26% of the hosts were more open for IPv6 than for IPv4 on at least one tested application port (versus 18% and 17%, respectively, that were more open for IPv4). For routers, the average application was open for 84% more hosts via IPv6 than IPv4, including SSH, which was reachable via IPv6 for a staggering 166% more routers than over IPv4. For servers, this number was a lower but still significant 12%, which notably included SSH (5% more open on IPv6), and Telnet (46%). The numbers were even higher for the 37% of servers that did not support HTTP (and, thus were less likely to be behind load balancers or gateways). Among those 191K servers, 49% more servers were open for IPv6 than IPv4 on the often-attacked server message block (SMB) protocol, 112% for Telnet, and 343% for SNMP. Lastly, deeper probing we conducted using traceroute-style probes also showed that, even when both protocols blocked an application, the manner in which policy is deployed (i.e., discrete firewall or host firewall) also differs between IPv4 and IPv6.

To review, the high-level findings in this study are the following:

- **IPv6 is more open than IPv4.** A given IPv6 port is nearly always more open than the same port/protocol is in IPv4. In particular, routers are twice as reachable over IPv6 for SSH, Telnet, SNMP, and BGP. While openness on IPv6 is not as severe for servers, we still find thousands of hosts open that are only open over IPv6.

- **When policy variances exist, they tend to exist network-wide.** Our analyses of differences between IPv4 and IPv6 policy show these policy differences tend to be consistent within autonomous systems and routed prefixes (e.g., for 78% of routers in the average prefix, their IPv4/IPv6 policy differences are consistent across the entire routed prefix).

- **Not only does policy differ, but policy mechanism as well.** We classify the type of connectivity failure through the addition of traceroute measurements and show that differences exist not only in how policy for various services are handled, but in how IP protocol versions are as well (e.g., 9% more routers respond *actively* over IPv6 when ports are closed, indicating fewer policy devices silently dropping than on IPv4).

- **Existing IPv6 open services are easily discovered through scanning.** Our analysis of host addressing patterns shows that, for a given subnet, most routers and a quarter of servers could be discovered by their IPv6 address within 10 seconds using state of the art port scanners.

Even when IPv6 was less open, there were hundreds or thousands of hosts for many applications that were only reachable via IPv6. While we can speculate that hosts which only support a service on IPv4 have yet to configure IPv6, it is more difficult to imagine plausible scenarios where a service is not intended to be available on IPv4 but is intentionally made so on IPv6. This is the reason we are concerned even when, for applications where there are relatively more IPv4-reachable hosts, we still find hundreds or thousands only accessible over IPv6. Although the lack of IPv6 connectivity may be an adoption problem, it is not a security problem; whereas, each of the hosts that do have a service reachable over IPv6 only, even if they are a minority of the hosts, could be exposed to an unexpected attack vector—a back door waiting for an IPv6-savvy attacker to come along and knock on it.

We note that the risk due to these services being reachable—where intended policy appears to be that they are not—is likely exacerbated by the lack of maturity of IPv6 tools and processes. For instance, older Netflow version 5 systems [173], which are essential elements of aggregating, transmitting, and storing network traffic data for network many operators, do not support IPv6 (the newer Cisco Netflow v9 and IETF standard IPFIX do), requiring both sources of flow information and sinks to be updated to have visibility into IPv6 traffic. Aside from Netflow, anecdotal evidence suggests some large organizations, including service providers, run various homegrown or legacy network management software that simply does not yet support IPv6.

4.9 Summary

In this chapter we examined the network filtering policy of a large sample of globally-distributed dual-stacked routers and servers. We found that, in an alarming number of cases, these important network hosts were exposing services to potential attack via IPv6 where the same application ports were blocked over IPv4. For thousands of measured routers and over 130,000 servers, attackers could attempt brute-force password guessing, launch TCP denial-of-service attacks, or use UDP to power reflected distributed denial-of-service attacks over IPv6, where the same attack vector on the host had been effectively blocked over IPv4. Our established contacts with a dozen operators confirmed that these lapses in security were not intentional and lead to immediate remediation in most of the twelve cases. However, much disparity still remains, and the broad set of hosts affected indicates a global, systemic failure to properly secure new hosts that are being dual-stack enabled. We fear that this is leading to a period where the rapid rise of basic IPv6 capability is surpassing the needed security controls, putting previously secure hosts, or—via vulnerable routers—even entire networks, in a weaker security state due to their added IPv6 capability

CHAPTER 5

Related Work

Our studies each build on a burgeoning body of research into core networking and network security in IPv6, as well as much interesting analogous work in IPv4. In this chapter, we highlight closely related work to each of the three studies that constitute this thesis.

5.1 IPv6 Adoption

There is much work in the literature that offers valuable data and insight into the IPv6 adoption process from various perspectives, though these typically focus on a deep analysis of just one or two aspects of the protocol's deployment, and usually from only one perspective (i.e., data source). Several studies characterize IPv6 traffic from the perspective of one or more ISPs (e.g., [111,156,160]) and 6to4 relays (e.g., [87,157]). On June 8, 2011, the Internet Society sponsored "IPv6 World Day" [102] and several pieces of work explore this event explicitly (e.g., [156]). Other work examines IPv6 adoption from the perspective of the World Wide Web (e.g., [38, 138]). Additionally, a variety of contributions explore the technical, economic, and social factors that influence adoption (e.g., [82,97]). Finally, much previous work focuses on topology measurements and performance in IPv6 and their relationships to IPv4 (e.g., [33, 56, 72, 141, 170, 180, 181]). In contrast to much of these studies, our adoption work sacrifices depth for breadth in order to form a longitudinal big picture of IPv6 adoption.

Our DNS packet analysis in § 2.3 extends work by researchers at Verisign [139, 171].

The key distinction of our contribution here over this previous work is that we examine DNS queries via both IPv4 and IPv6 traffic, we focus on IPv6 adoption, and the data presented is more recent; the earlier Verisign work contains longitudinal IPv4-only traffic analysis, though performed at greater detail.

Claffy [37] discusses IPv6 evolution and observes that “we lack not only a comprehensive picture of IPv6 deployment, but also consensus on how to measure its growth, and what to do about it.” Our study is in part a response to this call, offering a possible way forward. Closest to our IPv6 adoption work in both spirit and substance is Karpilovsky *et al.* [111], who provide a snapshot of IPv6 adoption from three main perspectives (allocation data, routing data, and traffic from a tier-1 ISP). In comparison, our work broadens the traffic perspective to a large sample of global tier-1 ISPs and nearly 100 tier-2/regional ISPs (260 providers in total), includes large samples of .com and .net TLD data, and juxtaposes these datasets with seven additional (mostly public) datasets.

5.2 IPv6 Network Telescopes

Related work to the study of IPv6 network telescopes and background radiation can be divided into the following broad categories: early IPv6 network telescope measurements; IPv4 background radiation analysis; and Internet background radiation collection considerations. In the following sections we highlight prior research under each of these three umbrellas.

5.2.1 Early IPv6 Network Telescope Measurements

The earliest known study to collect IPv6 background radiation advertised an small /48 prefix (the typical size of single organization’s IPv6 address allocation) for a month in 2006, only collecting 12 packets over that time period [65]. More recent studies have announced a single covering /12 prefix [52, 99]. Both of these works served as an initial inspiration for our own experiment. In the shorter term study, Deccio *et al.* collected two weeks of data and reported an average of 74 packets/second of pollution traffic [52]. In the longer term study, Huston *et al.* collected data for 115 days and presented analysis of observed traffic [99]. They reported a negligible increase in average pollution traffic rate during that period. That study also presented a detailed analysis of traffic address destinations. Our results confirm a major finding of that work, which was that the vast majority ($\approx$95%) of captured traffic was for *allocated* networks.

The results we report complement these earlier efforts by scaling up the basic approach in order to better understand global and regional trends. In addition, we also adopt a more rigorous approach by quantifying not only the BGP-control-plane reachability of our prefix announcements, but also data-plane-traffic reachability. Further, we attempted to verify that our global-scale experiment did not negatively impact actual IPv6 traffic on the Internet. Lastly, we present a more thorough analysis of the collected data, providing greater confidence about the generalizability of our results to overall IPv6 pollution.

5.2.2 IPv4 Background Radiation Analysis

Other related work has attempted to characterize and quantify IPv4 pollution traffic. Wustrow *et al.* analyzed a five-year sample of data collected passively from an unallocated IPv4 /8 block in addition to weeklong captures of three newly-allocated unused /8 prefixes [175]. Their temporal and spatial analysis revealed that background pollution traffic increased four-fold over the course of their sample and that pollution traffic was increasingly dominated by traffic resulting from misconfiguration and other errors. In contrast to their IPv4 findings, our work showed no discernible malicious activity indicators (i.e., neither scanning nor significant backscatter) in IPv6.

In work by Pang *et al.*, Yegneswaran *et al.*, and Cooke *et al.*, the authors utilize active responders in order to gain additional insight into the sources of pollution traffic [41, 140, 176]. Bailey *et al.* discuss the advantages of an active-responder-based network telescope monitor [17]. We did not use active responders in our study in an effort to ensure that we did not directly affect valid IPv6 traffic during this critical transition period; given that we were advertising a *covering prefix*, and, thus, receiving connection attempts often from and to active address space, such active response could have interfered with legitimate host traffic.

Glatz *et al.* and Brownlee *et al.* examine sources of Internet pollution and attempt to create classification schemes for this traffic based on various parameters, such as source address, protocol, and inter-arrival times. They present data that quantifies the amount of such traffic on a given network, as well as its properties [25, 73].

Internet pollution traffic has also been analyzed to provide insight into large-scale scanning activities [48], Internet censorship [113], and even large-scale physical events such as outages from earthquakes and hurricanes [7, 49]. Barford *et al.* use similar data to find the location of malicious hosts [20].

5.2.3 Background Radiation Collection Considerations

Similar techniques to the ones used in our study have previously been discussed in the work of Bailey *et al.*, which describes the use of honeypots with network telescope monitoring [18]. They analyzed a distributed IPv4 network telescope with over 60 smaller network telescopes and 17 million routable addresses to determine the difficulties in implementing such a hybrid monitoring system.

The size and spatial location of a network telescope are also an important factor in determining the amount of pollution traffic it receives. Moore *et al.* describe the effect the size of a network telescope has on the types of security events witnessed, such as worm spreading, scanning, and distributed denial of service (DDoS) attacks. Similarly, Cooke *et al.* examine Internet background radiation data over ten distributed monitors and study how location affects the collected data [40, 133]. These various studies all conclude that location, visibility, route announcement propagation, and filtering all have the potential to affect the observed traffic. Based on these conclusions, we paid particular attention to the data-collection details when designing our study.

5.3 IPv6 Security Policy

Standards documents and deployment guides (e.g., [31, 66, 75, 112]) have been urging network operators to apply firewall rules and access control lists for IPv6 in parity with IPv4 as part of their deployment of the new protocol. Instead, security researchers as well as RFC authors have lamented that in practice: “networks tend to overlook IPv6 security controls: [often] there is no parity in the security controls [between] IPv6 and IPv4” [5], and “in new IPv6 deployments it has been common to see IPv6 traffic enabled but none of the typical access control mechanisms enabled for IPv6” [32]. Beyond the quotes, we were not aware of any data that would shed light on the extent of real-world deployment of security filtering for IPv6. A desire to measure and raise awareness about these fundamental security control disparities in IPv6 was the motivation for our work. To our knowledge, ours is the first such Internet-scale study of deployed IPv6 security policies.

The IPv6 protocol was standardized nearly twenty years ago, before many lessons had been learned about Internet security. On top of that, IPv6 introduces many changes and features that go far beyond increased address space. As such, issues with the design and implementation of IPv6 have come to the fore as IPv6 is given more scrutiny by early adopters and researchers. For instance, Ullrich *et al.*, aggregated a taxonomy of 36 known design and implementation weaknesses in IPv6 [165]. Several of these problems (e.g., fragmentation header related, hop-by-hop header) have been discussed for some time by practitioners and non-academic security researchers (e.g., [16]), and there have been a number of updates to the original IPv6 specifications in recent years as a result (e.g., [8, 74, 76]). However, hardware changes to support the updated specifications in policy enforcement devices are taking time for vendors to implement and operators to deploy (e.g., [5]), leaving some networks vulnerable. A carefully controlled study of such vulnerabilities would be interesting future work. Our study, however, focused on characterizing the apparent *misalignment* of network security policy between IPv4 and IPv6, as measured by relatively reachable application ports. This shines light on the vulnerability that IPv6 poses as a *path* to exploit *upper layer applications*, rather than exploring weaknesses in IPv6 itself. As such, our work is largely orthogonal to such research and standards changes.

There has been a good deal of interest in characterizing the size of the open or vulnerable IPv4 host population for various ports, with several recent studies related to the topic of large-scale IPv4 application discovery (e.g., [59, 152]. To our knowledge, ours is the first work to attempt an Internet-scale characterization and comparison of commonly vulnerable or high-value open applications on IPv6 versus IPv4. As a further difference, rather than blanketing the address space with service probes, our work focuses on hosts that are more intentionally accessible by way of having hostnames associated with their addresses via A, AAAA, and PTR records in DNS. As recent studies (e.g., [44, 56]) and data (e.g., [78]) have shown a surge in IPv6 deployment, security weaknesses related to the rise of IPv6 are naturally of interest to the network security community. Relatedly, ours is the first large-scale study of the degree to which dual-stacked hosts provide the same services across both protocols, a metric pertinent to the study of IPv6 adoption itself.

IPv6 host addressing schemes deployed in the wild were last studied at scale by Malone in 2008 [126], though much has changed since that study, including several orders of magnitude more IPv6 deployment and the phasing out of EUI-64-based host identifiers by common operating systems. However, the results from Malone's September 2007 traceroute data, which is likely dominated by routers, shows 80-90% of host IDs there using just the lower 8 bits, suggesting some improvement in desirable randomness of HIDs in the intervening eight years (we saw 68%). Methods for discovering IPv6 hosts for scanning via leveraging secondary information sources such as the DNS (earlier discussed by Bellovin *et al.* [22], then in RFC 5157 [34], and more recently in an IETF draft [77]) have been successfully applied to IPv6 client discovery in recent years (e.g., [145]). Our analysis of the host IDs used by servers and routers in particular showed secondary sources were not necessary for identifying large fractions of these high value hosts given today's address allocation patterns and modern scanners. This should help dispel the myth that simple scanning on IPv6 is futile, and it somewhat heightens the risk associated with our core application openness findings in the IPv6 network security policy study.

CHAPTER 6

Conclusion

This thesis sought to shed light on the state of adoption, usage, routing stability, and security of the Internet Protocol, version 6. Through three global-scale measurement studies of IPv6, we have discovered several properties of the emerging network as currently deployed.

The measurements of man-made systems, especially ones as dynamic as the Internet, naturally have a short shelf life. However, we contend that the IPv6 network we examined is important enough and its global-scale measurement is challenging enough that the detailed studies we present are of lasting value to the network research community. First, they serve as a multi-dimensional, at-scale snapshot of the state of this large complex system during a key period in its evolution. Second, they describe both new and existing measurement methodologies that can be successfully applied to the IPv6 network now and in the future. Third, the studies, by pointing out either areas of slower adoption (e.g., naming, content) or systemic problems (e.g., route leakage, security policy disparity) with IPv6, can inform the *design and planning* of subsequent large-scale Internet upgrades. Relatedly, our work can help highlight patterns in the large-scale transition of federated complex systems, which may serve to inform not just the design but also the *study* of such systems in the future.

In the following sections, we summarize our key contributions and high-level insights from the studies in this thesis, discuss recent adoption progress, and propose areas of future work.

6.1 Summary of Contributions

This thesis made several contributions to the empirical understanding of the IPv6 Internet.

- **A broad longitudinal measurement of IPv6 adoption.** We conducted the longest and broadest-scale IPv6 adoption measurement study to date, exploring twelve metrics of adoption across a compendium of ten data sets, several spanning ten years of progress.

- **A holistic understanding of adoption.** Aside from being broad and longitudinal, our IPv6 measurements afford a *systemic* view of IPv6 adoption. By examining the adoption progress from multiple perspectives at multiple layers of the system, we were able to observe patterns in the protocol's deployment that are only evident when simultaneously examining many aspects of the network. Our study showed, across multiple datasets, a qualitative shift in the nature of IPv6 use as well as a marked jump in the pace of its adoption, heralding a maturing of the new protocol into a first-class content-carrying component of the Internet.

- **An exploration of the covering prefix methodology in network telescopes.** Our IPv6 background radiation chapter described the first academic IPv6 study and the largest published use of a covering prefix methodology for capturing Internet background radiation. This perspective on background radiation allowed us to (*i*) measure the routing stability of the new protocol, (*ii*) capture and study more typical IPv6 traffic, and (*iii*) detect misconfiguration types not otherwise visible.

- **A broad, deep measurement of IPv6 Internet Background Radiation.** Using the covering prefix methodology, we conducted the broadest IPv6 network telescope study to date. This broad perspective allowed us to detect differences among global regions, to closely examine IPv6 routing instability at scale, and to draw conclusions about the early state of global IPv6 network scanning and worm traffic. We found that IPv6 routing is not as mature as IPv4 routing in terms of stability and that basic management plane misconfiguration is common on the new network.

- **A broad measurement of global IPv6 network security policy.** We probed network security policy in a globally-distributed sample of 25K routers and 520K servers. We found substantial vulnerabilities in this population, including most egregiously on dual-stacked routers, which allowed TCP connections to Telnet, SSH, and BGP ports, at rates substantially higher than over IPv4. These findings suggest a weaker deployed IPv6 security policy relative to IPv4.

6.2 Insights and Takeaways

The global state of IPv6 deployment is something so large, multifaceted, and complex as to be akin to the state of global climate. As such, no handful of studies can do more than illuminate a small fraction of the phenomenon. Further, as this is the first core upgrade of a man-made system this immense, one involving this many distributed artifacts, the lessons learned here may not be fully generalizable to other similar systems or even future upgrades to the Internet itself. With these caveats, in this section we take a step up the ladder of abstraction and underline the higher-order lessons from our studies that we believe are of value for both designers and students of future Internet-scale technology upgrades.

First, we have learned that the whole is greater than the sum of its parts. In the spirit of the psychology concept of "gestalt," we find that examining multiple layers of protocol adoption and examining several of these from multiple vantage points sheds light on the phenomena that the same individual studies conducted in isolation would not be able to. The juxtaposition of multiple measurements of IPv6 adoption from various angles has allowed us to: (*i*) detect an ordering (phased adoption) pattern; (*ii*) find that our view of the *level* of adoption is greatly determined by the aspect we measure; (*iii*) discover that, while the obvious metric of raw traffic is low, this by itself is very misleading, since underlying readiness for IPv6 is much higher and recent traffic growth in this metric has been non-linear; and, most interestingly, (*iv*) detect a dramatic qualitative shift in the nature of IPv6 use in recent years—one signaling a true coming of age. Given this insight, a lesson we draw is that measurements of the state of such massive scale systems should be taken at multiple layers of the system or levels of abstraction; any single published metric, in isolation, should be interpreted with extreme caution.

Second, we observed a phased adoption pattern, wherein lower-layer or prerequisite aspects of the protocol were generally being adopted at higher rates or earlier than upper-layer aspects. For instance, our metric for network address allocation, a requisite step for operating an IPv6 network, ended 2014 at a level of around 0.5 relative to IPv4, whereas routing topology richness was only at around 0.02, and actual traffic, which is a measurement of protocol adoption that naturally follows at a later stage than address allocation and routing, was at just 0.006. To some extent, such ordering is expected. However, the magnitude of the differences we found was surprising. Beyond just resulting in drastically different perceptions of adoption depending on the layer measured, it suggests that the underlying *latent capacity* for IPv6 is much broader than the actual traffic numbers would suggest, which means that protocol use could spike dramatically over a short period of time if circumstances align. For instance, another IPv6 flag day or launch day that targets content providers, much as the summer 2012 IPv6 launch [103] did, could result in a sudden discontinuity in IPv6 usage. It appears that, beyond the prerequisite and actual usage layers, the even higher layers of network operation and security seem to follow the inverse lagging trend. This is critical to understand, as it has serious repercussions on the reliability of the overall network as more of it enables IPv6 as the preferred path in dual-stack hosts.

A possible lesson here is that core protocol upgrades may lead to a degradation of the system between the time that basic functions are deployed and the time that parity with the older protocol is achieved in terms of proper routing management and security. This point is important enough to reiterate. Two findings in our studies stick out: first, IPv6 adoption has been accelerating to the point that the network is already being used for production and at an exponentially growing rate; and, second, maturity level at the higher-level functions (routing management and port security) has put the emerging dual-stack IPv6 network in a degraded state of reliability and security. If network availability and security remain weaker in IPv6, conceivably, the medium term might bring a spate of IPv6-based attacks, since the network has grown large enough to become a target and broad enough to expose a significant number of operators but is not yet mature enough to defend itself on par with the IPv4-only network.

That also brings us to the third high-level insight, which is the notion of regressions. In upgrading to IPv6, we've seen that, as is common with software upgrades, there are what can be termed *network regressions* that creep up. Problems that had already been solved in IPv4 rear their heads again in IPv6. In addition to poorer routing and basic network security not being as well-deployed, which our studies have shown, other researchers have uncovered other regressions, including, for example: source routing [61]; smurf attacks (directed broadcast via all-node multicast) [135]; ND cache exhaustion (the analogue of ARP table exhaustion, with many orders of magnitude more punch) [71]; default gateway spoofing via router advertisement (RA) [35]; and, basic firewalling becoming difficult again because of the IPv6 header design and fragmentation [5]. This is not an comprehensive list. The broader lesson here is that such network regressions must be kept in mind when designing and deploying protocols. While there is no magic bullet, in practice this will likely mean that new protocol designs must more carefully integrate the teachings from failures of the past.

The fourth insight and another important future design issue has to do with backwards compatibility. It is often lamented that a key design problem of IPv6 has been the fact that the new protocol was not compatible with IPv4, and, effectively, required a completely new stack [27, 98, 127]. This design choice has had numerous consequences, including raising the cost and barrier to adoption [98], leading to code that had been hardened by years of deployment (i.e., the IPv4 stack) to be paired with less mature code, and requiring a large learning curve by operators to deploy [161]. While our work did not focus on these repercussions of the IPv6 design, the problems we found in basic security as well as misconfiguration issues add some fuel to the argument for backward compatibility. If the design of the new version of IP had been such that it could leverage more existing IPv4 components and infrastructure, such as falling under existing firewall rules (e.g., if addresses in the new protocol were integrated addresses of the older type) the set of new processes, tools and knowledge needed to deploy IPv6 may have been reduced. In that scenario, perhaps some of the problems our studies identified (as well as some of the resistance to adoption) may have been ameliorated. It is difficult to reason about what could have been, but certainly there are many familiar examples of computing technology where backwards compatibility was a design goal of a successor to some system or protocol (e.g., new Microsoft Office software versions being able to open saved files from older versions; USB 3.0 physical ports and chipsets supporting older cables and devices; and Blu-Ray players supporting DVDs, etc.). In each of these cases, systems of the new version entered the market as hardware/software was replaced, without discarding the user investments made in the previous versions of these technologies. We can not be certain that this helped the adoption of the newer technology, but it seems highly plausible. The lesson here is that incremental backwards-compatible upgrades are advised for future Internet upgrades, as they may reduce technology adoption friction.

Fifth, while the Internet is global, actual adoption of technologies is local. Thus, we expected to see substantial differences between global regions due to disparities in the motivations and incentives for adoption. However, in addition to the level of adoption differing for a given metric across regions, we also found that the relative ranking of global regions varies by the metric examined. Some regions are further ahead in address routing for example, while lagging other regions in traffic. Because of this finding, we have some evidence that policy, which is regional, at the various levels (e.g., address allocation rules [15, 149]) may impact adoption to differing extents. This is, effectively, a kind of natural experiment on adoption incentives. The lesson here is that policy-based interventions may have measurable impact on the deployment of IPv6, beyond the organic demand for the protocol that the Internet market creates; policy profoundly impacts technology.

Finally, on the human factors front, we have seen that the new protocol requires considerable effort in order to bring network operators' and researchers' expertise to the same level as their IPv4 knowledge [85, 94, 161, 163]. The skills gap likely stems from the fact that the older protocol is a familiar daily part of network operation, whereas the new one, especially prior to 2011, has been fairly exotic in most parts of the Internet. Although we leave collection of data illuminating the reasons for adoption rates to future work, we conjecture that this lack of familiarity itself is hindering both basic adoption and deployment of requisite management functions. Thus, the lesson here is that better operator education may grease the wheels for smoother for future complex technology deployments.

6.2.1 Advice for Future Internet Upgrades

To inform future Internet upgrades, here we summarize our advice for easing and expediting complex large-scale technology deployments:

1. No single measurement should be relied on in isolation when evaluating the state of such a large, federated, and complex system.

2. Network regressions must be kept in mind when designing and deploying protocols. New protocol designs must carefully integrate the teachings from failures of the past.

3. Future upgrades must be backwards compatible. Systems running the new protocol should accept data (packets) from the older version, and, ideally, interact with older-version systems transparently.

4. If the new version is preferred by multi-version systems, it should be at least as dependable as the older version; otherwise, depending on the fallback mechanisms, the entire system's reliability may be degraded.

5. Address allocation, route admittance, and other socio-economic policies governing the technology should be carefully crafted to encourage desired behavior.

6. The operators, engineers, and researchers whose work impacts the quality of the new technology's functioning should be well-educated about new protocol prior to deployment.

6.3 Adoption Progress Update

We concluded our IPv6 adoption study at the start of 2014, but the pace of deployment is swift and IPv6 has continued to grow. There has been progress on among both the content provider and the content consumer networks and devices.

6.3.1 Content-Side Progress

In September 2015, for example, Facebook, the world's third most popular website [10], revealed in a talk that 10% of its worldwide users were accessing the site over IPv6 (up from 1% in mid-2012), and use in the U.S. is higher, at around 23%. On mobile, in particular, a third of U.S. Facebook users and 45% of the high-end (4G/LTE) users connect over IPv6 [153]. Interestingly, in the same talk, Facebook revealed that an internal study showed that its users over IPv6 experience 15% faster connections than over IPv4, likely due to fewer middleboxes. Facebook has also moved most of its internal traffic off of IPv4 to IPv6 and aims to complete the transition to *single-stack IPv6-only* within the next two years [177]. Another major player on the content side, Google, who operate the world's top most visited website [10], has continued to see IPv6 growth, with early November 2015 data showing 9% of Google's users arriving to the site over IPv6 [78]. While these are very promising numbers, we note that not all content providers are as far along in IPv6 support as these examples would suggest. Our continuing bimonthly data collection (see § 2.5) shows that, at the end of October 2015 just 5% of the top 10,000 Alexa websites are reachable by IPv6.

6.3.2 Client-Side Progress

On the content consumer side, we are also seeing progress. In 2015, Apple announced that its iOS 9 and OS X 10.11 (El Capitan) operating systems released in the fall of 2015 would revamp the Happy Eyeballs algorithm [174], which is employed to decide between IPv4 and IPv6 on dual-stacked hosts, so that it more often favors IPv6. The result is expected to significantly increase clients choosing IPv6 on dual-stacked Apple devices, from approximately 50% of the time before the change to 99% after [30]. Globally, recent data from continuous tests at APNIC show that, as of November 2015, nearly 5% of Internet users are IPv6-capable, up from just 1.5% two years prior and 3% in November 2014 [13]. The growth has varied significantly by country, however. In the U.S. in particular, growth has been robust in the last several years, with the U.S. now in third place globally, having 27% of users IPv6-capable. This is likely due to some of the largest ISPs rolling out the protocol. In mid-2015, for example, Comcast, the largest American ISP [117], reported that the majority of its customers have IPv6 connectivity and that 10% of Internet traffic they see is IPv6 [6].

6.4 Future Work

The proliferation of IPv6 is still at an early stage, and the number of aspects of adoption, as well as the forces that shape it, are many. Our experiments have only begun to scratch the surface in understanding a small number of the many facets of IPv6 deployment and security. Much additional work remains.

For instance, on the study of raw support for IPv6, our work did not include analysis of the devices and software that form the substrate on which the network runs. For one, the state of IPv6 support in hardware is an area warranting study. Although many vendors claim IPv6 support, anecdotal evidence suggests that this support is sometimes a weaker version of what the devices support for IPv4 On the software side, key functions of running modern networks, including services like DHCP and DNS, as well as management tools like network monitoring systems must be made to support IPv6. The extent of this readiness is an open question and deserves a systematic look.

Our application of the covering prefix methodology in network telescopes has not been applied extensively in IPv4. It would be useful to conduct studies using this method in order to discover the prevalence of network outages caused by brief route advertisement failures as well as to capture meaningful IPv4 background radiation, now that completely unused large blocks of address space are increasingly rare. While our permission to advertise the covering prefixes in IPv6 expired after our study, it would be useful to maintain such broad network telescopes and sensors on the IPv6 network for long-term study of the phenomena these methods illuminate, as well as to provide early warning for broad scanning in IPv6.

On the pure network security front, the IPv6 capabilities of both hardware and software is also a concern. Our work did not explore the various systemic security problems with IPv6 policy devices including the nuances related to correct IPv6 header parsing by hosts and policy devices alike, both of which had been reported by security practitioners in recent years (e.g., [16]. More broad and systematic measurement studies as well as proposals for solutions to some of these protocol challenges by the network research community are needed in this space. Our treatment of IPv6 security has been at the level of measuring fundamental properties (availability and blocking policy) of existing deployments. But, we did not examine the security flaws of either the protocol specifications or the implementation of those specifications in deployed systems. More systematic work in this space is needed.

We would also like to know what factors have been motivating or hindering adoption, including non-technical factors such as economic, social, and behavioral ones. Understanding these issues more thoroughly, especially with support from empirical studies of operator readiness and attitudes, could contribute to a smoother adoption path for future global Internet updates.

6.5 Closing Thought: A Sea Change

We are witnessing a monumental change in the core technology of the Internet. Since the time it evolved from ARPANET and NSFNET roots into its modern form, it has arguably not faced a core technological change this significant. Deceptively, on the surface the IPv6 transition may seem to be little more than adding bits to the address space. There is growing awareness, however, outside of the Internet engineering and standards communities, of the fundamental differences that IPv6 brings. These, in turn, have lead to challenges in reliably and securely operating IPv6 networks—including, for example, due to something as basic as address presentation differences and address agility:

> In an IPv4 network, it is easy to correlate multiple logs, for example to find events related to a specific IPv4 address. A simple Unix grep command was enough to scan through multiple text-based files and extract all lines relevant to a specific IPv4 address.
>
> In an IPv6 network, this is slightly more difficult because different character strings can express the same IPv6 address. Therefore, the simple Unix grep command cannot be used. Moreover, an IPv6 node can have multiple IPv6

> addresses...
>
> In order to do correlation in IPv6-related logs, it is advised to have all logs with canonical IPv6 addresses. Then, the neighbor cache current (or historical) data set must be searched to find the data-link layer address of the IPv6 address. Then, the current and historical neighbor cache data sets must be searched for all IPv6 addresses associated to this data-link layer address: this is the search set. The last step is to search in all log files (containing only IPv6 address in canonical format) for any IPv6 addresses in the search set. [32]

We highlight the above quote from an Internet Engineering Task Force draft to reiterate that the deployment of IPv6 has implications far exceeding merely a *larger* address space. Changes the protocol introduces, both stark and subtle, mean that the Internet operations and research community has much work to do before the quality of the IPv6 network measures up to the bar set by the modern IPv4 Internet.

As we hope this thesis has helped show, IPv6 is not just more bits; it is a sea change. Fortunately, at least for researchers, this means a great deal of interesting work yet to be done.

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Cisco Nexus 7000 Series NX-OS System Management Configuration Guide, Release 5.x. http://www.cisco.com/c/en/us/td/docs/switches/datacenter/sw/5_x/nx-os/system_management/configuration/guide/sm_nx_os_cg/sm_3ntp.html#wp1107779.

[2] CloudFlare : IPv6 Gateway Feature. https://www.cloudflare.com/ipv6.

[3] IPv6 versus IPv4 Header Image. https://ls-a.org/lib/exe/detail.php?id=school%3A2b_chapter_7_notes&media=school:ipv4-vs-ipv6-header.png. Licensed for free use via GNU FDL 1.3.

[4] NMAP : ssl-enum-ciphers. https://nmap.org//nsedoc/scripts/ssl-enum-ciphers.html.

[5] Security Assessments of IPv6 Networks and Firewalls. http://www.si6networks.com/presentations/ipv6kongress/mhfg-ipv6-kongress-ipv6-security-assessment.pdf, June 2013. Slides of Presentation at IPv6 Kongress.

[6] The Benefits of Deploying IPv6 Only. https://www.youtube.com/watch?v=Efjd0c41g0s, June 2015. Video of a talk given at NANOG 64, May 21–June 3, San Francisco, California.

[7] Emile Aben, Alistair King, Karyn Benson, Young Hyun, Alberto Dainotti, and KC Claffy. Lessons Learned by "Measuring" the Internet During/After the Sandy Storm. In *In Proceedings of FCC Workshop on Network Resiliency 2013*, February 2013.

[8] J. Abley, P. Savola, and G. Neville-Neil. RFC 5095: Deprecation of Type 0 Routing Headers in IPv6, 2007.

[9] Alexa. Alexa Top 1M Sites. http://www.alexa.com/topsites.

[10] Alexa. Alexa Top 500 Sites. http://www.alexa.com/topsites, March 2016.

[11] Mark Allman. IPv6 DNS & Reachability Dataset. http://www.icir.org/mallman/data/ipv6-dns-data.tar.gz.

[12] J. Ignacio Alvarez-Hamelin, Luca Dall'Asta, Alain Barrat, and Alessandro Vespignani. k-core Decomposition: a Tool for the Visualization of Large Scale Networks. *CoRR*, 2005.

[13] APNIC. IPv6 Measurement Maps. `http://stats.labs.apnic.net/ipv6`.

[14] APNIC Pty. Ltd. APNIC's IPv4 Pool Usage, 2012. `http://www.apnic.net`.

[15] ARIN. ARIN Number Resource Policy Manual. `https://www.arin.net/policy/nrpm.html`, 2016.

[16] Antonios Atlasis. Attacking IPv6 Implementation Using Fragmentation. `https://media.blackhat.com/ad-12/Atlasis/bh-ad-12-security-impacts-atlasis-wp.pdf`, 2012.

[17] Michael Bailey, Evan Cooke, Farnam Jahanian, Andrew Myrick, and Sushant Sinha. Practical Darknet Measurement. In *Proceedings of the 40th Annual Conference on Information Sciences and Systems*, CISS'06, 2006.

[18] Michael Bailey, Evan Cooke, Farnam Jahanian, Niels Provos, Karl Rosaen, and David Watson. Data Reduction for the Scalable Automated Analysis of Distributed Darknet Traffic. *Proceedings of the ACM Internet Measurement Conference*, 2005.

[19] Michael Bailey, David Dittrich, Erin Kenneally, and Doug Maughan. The Menlo Report. *IEEE Security & Privacy*, 10(2):71–75, 2012.

[20] Paul Barford, Rob Nowak, Rebecca Willett, and Vinod Yegneswaran. Toward a Model for Source Address of Internet Background Radiation. In *Proceedings of the Conference on Passive and Active Network Measurement*, PAM'06, 2006.

[21] D. Barr. RFC 1912: Common DNS Operational and Configuration Errors, 1996.

[22] Steven M Bellovin, Bill Cheswick, and Angelos Keromytis. Worm Propagation Strategies in an IPv6 Internet. *LOGIN: The USENIX Magazine*, 2006.

[23] Robert Beverly and Arthur Berger. Server Siblings: Identifying Shared IPv4/IPv6 Infrastructure via Active Fingerprinting. In *Proceedings of the Passive and Active Measurement Conference*, PAM'15, 2015.

[24] S. Bradner and A. Mankin. RFC 1752: The Recommendation for the IP Next Generation Protocol, 1995.

[25] Nevil Brownlee. One-Way Traffic Monitoring with iatmon. In *Proceedings of the Conference on Passive and Active Network Measurement*, PAM'12, 2012.

[26] CAIDA. Archipelago (Ark) Measurement Infrastructure. `http://www.caida.org/projects/ark/`.

[27] Graeme Caldwell. Why Is the Transition To IPv6 Taking So Long? `http://teamarin.net/2014/08/13/transition-ipv6-taking-long/`, August 2014.

[28] George Chalhoub. Answer to: Are There Enough IPv6 Addresses for Every Atom on the Surface of the Earth?, June 2014. http://skeptics.stackexchange.com/a/22502/7131.

[29] David R. Cheriton and Gritter Mark. TRIAD: A Scalable Deployable NAT-based Internet Architecture. http://gregorio.stanford.edu/papers/triad/triad.html, March 2000.

[30] Richard Chirgwin. Apple Snuggles Closer to IPv6. http://www.theregister.co.uk/2015/07/12/apple_snuggles_closer_to_ipv6/, July 2015.

[31] K. Chittimaneni, T. Chown, L. Howard, V. Kuarsingh, Y. Pouffary, and E. Vyncke. RFC 7381: Enterprise IPv6 Deployment Guidelines, 2014.

[32] Kiran Chittimaneni, Merike Kaeo, and Eric Vyncke. Operational Security Considerations for IPv6 Networks. March 2015.

[33] Kenjiro Cho, Matthew Luckie, and Bradley Huffaker. Identifying IPv6 Network Problems in the Dual-stack World. In *Proceedings of the ACM SIGCOMM Workshop on Network Troubleshooting: Research, Theory and Operations Practice Meet Malfunctioning Reality*, NetT '04, 2004.

[34] T. Chown. RFC 5157: IPv6 Implications for Network Scanning, 2008.

[35] T. Chown and S. Venaas. RFC 6104: Rogue IPv6 Router Advertisement Problem Statement, 2011.

[36] Cisco. The Zettabyte Era—Trends and Analysis. http://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/VNI_Hyperconnectivity_WP.html, May 2015.

[37] kc claffy. Tracking IPv6 Evolution: Data We Have and Data We Need. *SIGCOMM Computer Communication Review*, 2011.

[38] Lorenzo Colitti, Steinar H. Gunderson, Erik Kline, and Tiziana Refice. Evaluating IPv6 Adoption in the Internet. In *Proceedings of the Conference on Passive and Active Network Measurement*. 2010.

[39] Louis Columbus. Where Internet of Things Initiatives Are Driving Revenue Now, July 2015. http://www.forbes.com/sites/louiscolumbus/2015/07/28/where-internet-of-things-initiatives-are-driving-revenue-now/#2b136fbdd671.

[40] Evan Cooke, Michael Bailey, Z. Morley Mao, David Watson, and Farnam Jahanian. Toward Understanding Distributed Blackhole Placement. In *Proceedings of the ACM Workshop on Rapid Malcode*, WORM'04, Oct 2004.

[41] Evan Cooke, Michael Bailey, David Watson, Farnam Jahanian, and Jose Nazario. The Internet Motion Sensor: A Distributed Global Scoped Internet Threat Monitoring System. Technical Report CSE-TR-491-04, University of Michigan, Electrical Engineering and Computer Science, 2004.

[42] John Curran. ARIN IPv4 Free Pool Reaches Zero, September 2015. https://www.arin.net/announcements/2015/20150924.html.

[43] Jakub Czyz, Mark Allman, Jing Zhang, Scott Iekel-Johnson, Eric Osterweil, and Michael Bailey. Measuring IPv6 Adoption. ICSI Technical Report TR-13-004, August 2013.

[44] Jakub Czyz, Mark Allman, Jing Zhang, Scott Iekel-Johnson, Eric Osterweil, and Michael Bailey. Measuring IPv6 Adoption. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM'14, 2014.

[45] Jakub Czyz, Michael Kallitsis, Manaf Gharaibeh, Christos Papadopoulos, Michael Bailey, and Manish Karir. Taming the 800 Pound Gorilla: The Rise and Decline of NTP DDoS Attacks. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'14, 2014.

[46] Jakub Czyz, Kyle Lady, Sam G Miller, Michael Bailey, Michael Kallitsis, and Manish Karir. Understanding IPv6 Internet Background Radiation. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'13, 2013.

[47] Jakub Czyz, Matthew Luckie, Mark Allman, and Michael Bailey. Dont Forget to Lock the Back Door! A Characterization of IPv6 Network Security Policy. In *Proceedings of the Network and Distributed System Security Symposium*, NDSS'16, San Diego, CA, February 2016.

[48] A. Dainotti, A. King, K. Claffy, F. Papale, and A. Pescapè. Analysis of a "/0" Stealth Scan from a Botnet. In *Proceedings of the ACM SIGCOMM conference on Internet Measurement*, IMC'12, Nov 2012.

[49] Alberto Dainotti, Claudio Squarcella, Emile Aben, kc. claffy, Marco Chiesa, Michele Russo, and Antonio Pescapé. Analysis of Country-wide Internet Outages Caused by Censorship. In *Proceedings of the ACM SIGCOMM conference on Internet Measurement*, IMC'11, 2011.

[50] Joe Davies. Domain Name System Client Behavior in Windows Vista. `http://technet.microsoft.com/en-us/library/bb727035.aspx`, 2006.

[51] G. Van de Velde, T. Hain, R. Droms, B. Carpenter, and E. Klein. RFC 4864: Local Network Protection for IPv6, 2007.

[52] Casey Deccio. Turning Down the Lights: Darknet Deployment Lessons Learned. Technical report, Sandia National Laboratories, 2012.

[53] S. Deering and R. Hinden. RFC 1883: Internet Protocol, Version 6 (IPv6) Specification, 1995.

[54] S. Deering and R. Hinden. RFC 2460: Internet Protocol, Version 6 (IPv6) Specification, 1998.

[55] L. Delgrossi and L. Berger. RFC 1819: Internet Stream Protocol Version 2 (ST2) Protocol Specification - Version ST2+, 1995.

[56] A. Dhamdhere, M. Luckie, B. Huffaker, K. Claffy, A. Elmokashfi, and E. Aben. Measuring the Deployment of IPv6: Topology, Routing and Performance. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'12, 2012.

[57] R. Droms. RFC 1531: Dynamic Host Configuration Protocol, 1993.

[58] Monica Dunahee and Harlan Lebo. World Internet Project Report–6th Edition. http://www.digitalcenter.org/wp-content/uploads/2013/06/2015-World-Internet-Report.pdf, 2015.

[59] Zakir Durumeric, Michael Bailey, and J Alex Halderman. An Internet-wide View of Internet-wide Scanning. In *Proceedings of the USENIX Security Symposium*, SEC'14, 2014.

[60] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. ZMap: Fast Internet-wide Scanning and Its Security Applications. In *The 22nd USENIX Security Symposium*, SEC'13, 2013.

[61] Jake Edge. IPv6 Source Routing: History Repeats Itself. https://lwn.net/Articles/232781/, May 2007.

[62] C. Feather. RFC 3977: Network News Transfer Protocol (NNTP), 2006.

[63] Greg Ferro. Scheduling the IPocalypse, November 2010. http://etherealmind.com/scheduling-ipocalypse/.

[64] Kelly Fiveash. DeSENSORtised: Why the 'Internet of Things' will FAIL without IPv6. http://www.theregister.co.uk/2014/04/24/ipv6_iot/, April 2014.

[65] Matthew Ford, Jonathan Stevens, and John Ronan. Initial Results from an IPv6 Darknet. In *International Conference on Internet Surveillance and Protection*, 2006.

[66] Sheila Frankel, Richard Graveman, John Pearce, and Mark Rooks. Guidelines for the Secure Deployment of IPv6. *NIST Special Publication*, 800-119, 2010.

[67] Mark Frauenfelder. The Great IP Crunch of 2010. http://www.cnn.com/TECH/computing/9909/21/ip.crunch.idg/index.html, 1999.

[68] V. Fuller, T. Li, J. Yu, and K. Varadhan. Supernetting: an Address Assignment and Aggregation Strategy, 1992.

[69] V. Fuller, T. Li, J. Yu, and K. Varadhan. RFC 1519: Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy, 1993.

[70] Sean Gallagher. NSA Hacker in Residence Dishes on How to Hunt System Admins, March 2014. http://arstechnica.com/security/2014/03/nsa-hacker-in-residence-dishes-on-how-to-hunt-system-admins/.

[71] I. Gashinsky, J. Jaeggli, and W. Kumari. RFC 6583: Operational Neighbor Discovery Problems, March 2012.

[72] Vasileios Giotsas and Shi Zhou. Detecting and Assessing the Hybrid IPv4/IPv6 AS Relationships. *SIGCOMM Computer Communication Review*, 2011.

[73] Eduard Glatz and Xenofontas Dimitropoulos. Classifying Internet One-Way Traffic. pages 417–418, 2012.

[74] F. Gont. RFC 6946 : Processing of IPv6 ”Atomic” Fragments, 2013.

[75] F. Gont and W. Liu. RFC 7123: Security Implications of IPv6 on IPv4 Networks, 2014.

[76] F. Gont, V. Manral, and R. Bonica. RFC 7112 : Implications of Oversized IPv6 Header Chains, 2014.

[77] F. Gont and T.Chown. Network Reconnaissance in IPv6 Networks. Internet-Draft draft-ietf-opsec-ipv6-host-scanning-07, IETF Secretariat, 2015.

[78] Google. IPv6 Statistics. http://www.google.com/intl/en/ipv6/statistics/.

[79] Enrico Gregori, Alessandro Improta, Luciano Lenzini, Lorenzo Rossi, and Luca Sani. On the Incompleteness of the AS-level Graph: a Novel Methodology for BGP Route Collector Placement. In *Proceedings of the 12th ACM Conference on Internet Measurement*, IMC'12, 2012.

[80] Chris Grundemann. IPv4 Is the sky falling? http://chrisgrundemann.com/index.php/2011/ipv4-sky-falling/, January 2011.

[81] Chris Grundemann. IPv6 Security Myth 2: IPv6 Has Security Designed In. http://www.internetsociety.org/deploy360/blog/2015/01/ipv6-security-myth-2-ipv6-has-security-designed-in/, January 2015.

[82] Roch Guérin and Kartik Hosanagar. Fostering IPv6 Migration Through Network Quality Differentials. *SIGCOMM Computer Communication Review*, 2010.

[83] Gonca Gürsun, Natali Ruchansky, Evimaria Terzi, and Mark Crovella. Inferring Visibility: Who's (not) Talking to Whom? *SIGCOMM Comput. Commun. Rev.*, 2012.

[84] Scott Gurvey. Who's afraid of IPv6? You shouldn't be! http://newsroom.cisco.com/feature-content?articleId=1746147, February 2016.

[85] Matt Hamblen. IPv6 Requires Learning Curve for Network Admins, August 2007.

[86] Warren Harrop and Grenville Armitage. Defining and Evaluating Greynets (Sparse Darknets). In *Proceedings of the The IEEE Conference on Local Computer Networks 30th Anniversary*, LCN '05, 2005.

[87] Y Hei and K Yamazaki. Traffic analysis and worldwide operation of open 6to4 relays for IPv6 deployment. In *Proceedings of the International Symposium on Applications and the Internet*, 2004.

[88] Stacey Higginbotham. The internet is going private. Its also grown to 138 Tbps of capacity, April 2014. `https://gigaom.com/2014/04/23/the-internet-is-going-private-its-also-grown-to-138-tbps-of-capacity/`.

[89] R. Hinden and S. Deering. RFC 1884: IP Version 6 Addressing Architecture, 1995.

[90] R. Hinden and S. Deering. RFC 4291: IP Version 6 Addressing Architecture, 2006.

[91] R. Hinden and B. Haberman. RFC 4193: Unique Local IPv6 Unicast Addresses, 2005.

[92] Scott Hogg. IPv6 Network Management. `http://www.networkworld.com/article/2225031/cisco-subnet/ipv6-network-management.html`, July 2013.

[93] Scott Hogg. Mobile Devices and BYOD are Driving IPv6 Adoption. `http://www.networkworld.com/article/2224844/cisco-subnet/mobile-devices-and-byod-are-driving-ipv6-adoption.html`, June 2013.

[94] Edward Horley. The IPv6 Skills Crisis. `http://www.informationweek.com/strategic-cio/team-building-and-staffing/the-ipv6-skills-crisis/a/d-id/1297481`, July 2014.

[95] Lee Howard. The Cost of IPv6. NANOG 57. Presented at the 57st North American Network Operators Grooup Meeting in Orlando, Florida, 2013.

[96] S. Huitema. RFC 4380: Teredo: Tunneling IPv6 over UDP through Network Address Translations (NATs), 2006.

[97] Geoff Huston. The Case for IPv6: Extinction, Evolution or Revolution? `http://www.potaroo.net/presentations/2006-06-23-ipv6-evolution.pdf`, 2006.

[98] Geoff Huston. IPv6 Transitional Uncertainties, September 2011. `http://www.circleid.com/posts/ipv6_transitional_uncertainties/`.

[99] Geoff Huston. IPv6 Background Radiation. Technical report, 2012. Slides of a talk given at DUST 2012 – The 1st International Workshop on Darkspace and UnSolicited Traffic Analysis, May 14–15, San Diego, California.

[100] Geoff Huston. Measuring IPv6 - Country by Country. https://labs.ripe.net/Members/gih/measuring-ipv6-country-by-country, June 2012.

[101] IANA. IPv6 Addresses for the Root Servers. http://www.iana.org/reports/2008/root-aaaa-announcement.html, January 2008.

[102] Internet Society. World IPv6 Day. http://www.worldipv6day.org/, 2011.

[103] Internet Society. World IPv6 Launch. http://www.worldipv6launch.org, June 2012.

[104] Simona Jankowski, James Covello, Heather Bellini, Joe Ritchie, and Daniela Cost. The Internet of Things: Making sense of the next mega-trend. http://www.goldmansachs.com/our-thinking/outlook/internet-of-things/iot-report.pdf, 2014.

[105] Jeff Jarmoc. SQL Slammer 10 years later. http://www.secureworks.com/resources/blog/research/general-sql-slammer-10-years-later/, 2013.

[106] Mobin Javed and Vern Paxson. Detecting Stealthy, Distributed SSH Brute-forcing. In *Proceedings of the ACM SIGSAC Conference on Computer & Communications Security*, SIGSAC'13, 2013.

[107] Richard Jimmerson. IPv4 is depleted. Now what? http://teamarin.net/category/ipv4-depletion/, January 2016. [Online; accessed 28-March-2016].

[108] D. Kaminsky. Black Ops 2008: It's the End of the Cache As We Know It. *Black Hat USA*, 2008.

[109] P. Kampanakis. Implementation Guidelines for parsing IPv6 Extension Headers. August 2014.

[110] Manish Karir, Geoff Huston, George Michaelson, and Michael Bailey. Understanding IPv6 Populations in the Wild. In *Proceedings of the Conference on Passive and Active Network Measurement*, PAM'13. 2013.

[111] Elliott Karpilovsky, Alexandre Gerber, Dan Pei, Jennifer Rexford, and Aman Shaikh. Quantifying the Extent of IPv6 Deployment. In *Proceedings of the International Conference on Passive and Active Network Measurement*, PAM'09, 2009.

[112] C. M. Keliiaa and V. N. McLane. Cyberspace Modernization: An Internet Protocol Planning Advisory. *SANDIA Report*, SAND2014-5032, July 2014.

[113] Alistair King. Syria disappears from the Internet. http://blog.caida.org/best_available_data/2012/12/05/syria-disappears-from-the-internet/, 2012.

[114] Christian Kreibich, Nicholas Weaver, Boris Nechaev, and Vern Paxson. Netalyzr: Illuminating the Edge Network. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'10, 2010.

[115] Craig Labovitz, Scott Iekel-Johnson, Danny McPherson, Jon Oberheide, and Farnam Jahanian. Internet Inter-Domain Traffic. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM'10, 2010.

[116] Mike Leber. Global IPv6 Deployment Progress Report. `http://bgp.he.net/ipv6-progress-report.cgi`, January 2014.

[117] Inc. Leichtman Research Group. 2.6 Million Added Broadband from Top Cable and Telephone Companies in 2013, March 2014. `http://www.leichtmanresearch.com/press/031714release.html`.

[118] Barry M. Leiner, Vinton G. Cerf, David D. Clark, Robert E. Kahn, Leonard Kleinrock, Daniel C. Lynch, Jon Postel, Larry G. Roberts, and Stephen Wolff. Brief History of the Internet. `http://www.internetsociety.org/internet/what-internet/history-internet/brief-history-internet`, October 2012.

[119] V. I. Levenshtein. Binary Codes Capable of Correcting Deletions, Insertions and Reversals. *Sov. Phys. Dokl.*, 1966.

[120] G. Lindberg. Anti-Spam Recommendations for SMTP MTAs. RFC 2505 (Best Current Practice), 1999.

[121] Steve Lohr. A New 'Law' for the Mobile Computing Era. `http://bits.blogs.nytimes.com/2012/01/11/a-new-law-for-the-mobile-computing-era/`, January 2012.

[122] Matthew Luckie. Scamper: a Scalable and Extensible Packet Prober for Active Measurement of the Internet. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'10, 2010.

[123] Matthew Luckie, Robert Beverly, Tiange Wu, Mark Allman, and k claffy. Resilience of Deployed TCP to Blind Attacks. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'15, 2015.

[124] Douglas MacFarland, Craig Shue, and Andrew Kalafut. Characterizing Optimal DNS Amplification Attacks and Effective Mitigation. In *Proceedings of the Passive and Active Measurement Conference*, PAM'15, 2015.

[125] Priya Mahadevan, Dmitri Krioukov, Marina Fomenkov, Xenofontas Dimitropoulos, k c claffy, and Amin Vahdat. The Internet AS-level Topology: Three Data Sources and One Definitive Metric. *SIGCOMM Computer Communication Review*, 2006.

[126] David Malone. Observations of IPv6 Addresses. In *Proceedings of the Passive and Active Measurement Conference*, PAM'08. 2008.

[127] Caroly Marsan. Biggest Mistake for IPv6: It's Not Backwards Compatible. `http://www.networkworld.com/news/2009/032509-ipv6-mistake.html`, 2009.

[128] Carolyn Duffy Marsan. IPv6 management tools lacking. `http://www.networkworld.com/article/2291003/lan-wan/ipv6-management-tools-lacking.html`, June 2007.

[129] Microsoft. Microsoft Security Intelligence Report - July–December 2012, 2013.

[130] Randy Milgrom. How the Net Was Won. `http://dme.engin.umich.edu/internet/`, 2015.

[131] D. Mills, J. Martin, J. Burbank, and W. Kasch. RFC 5905: Network Time Protocol Version 4: Protocol and Algorithms Specification, 2010.

[132] P. Mockapetris and K. J. Dunlap. Development of the Domain Name System. In *Proceedings of the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications*, SIGCOMM'88, 1988.

[133] David Moore, Colleen Shannon, Geoffrey M. Voelker, and Stefan Savage. Network telescopes. Technical Report CS2004-0795, UC San Diego, 2004.

[134] MySQL IPv6 Support. `https://dev.mysql.com/doc/refman/5.5/en/ipv6-server-config.html`.

[135] Santosh P Naidu and Amulya Patcha. IPv6: Threats Posed by Multicast Packets, Extension Headers and Their Counter Measures. *International Journal of Computer Science and Network Security (IJCSNS)*, 14(10), 2014.

[136] T. Narten, G. Huston, and L. Roberts. RFC 6177: IPv6 Address Assignment to End Sites, 2011.

[137] T. Narten, E. Nordmark, W. Simpson, and H. Soliman. RFC 4861: Neighbor Discovery for IP version 6 (IPv6), 2007.

[138] Mehdi Nikkhah, Roch Guérin, Yiu Lee, and Richard Woundy. Assessing IPv6 Through Web Access: a Measurement Study and its Findings. In *Proceedings of the COnference on emerging Networking EXperiments and Technologies*, CONEXT'11, 2011.

[139] Eric Osterweil, Danny McPherson, Steve DiBenedetto, Christos Papadopoulos, and Daniel Massey. Behavior of DNS' Top Talkers, a .com/.net View. In *Proceedings of the Conference on Passive and Active Network Measurement*, PAM'12, 2012.

[140] Ruoming Pang, Vinod Yegneswaran, Paul Barford, Vern Paxson, and Larry Peterson. Characteristics of Internet Background Radiation. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'04, 2004.

[141] David Plonka and Paul Barford. Assessing Performance of Internet Services on IPv6. In *Proceedings of the IEEE Symposium on Computers and Communications*, ISCC'13, 2013.

[142] P. Porras, H. Saidi, and V. Yegneswaran. An Analysis of Conficker's Logic and Rendezvous Points. Technical report, SRI International, March 2009.

[143] J. Postel. RFC 760: DoD Standard Internet Protocol, 1980.

[144] Aiko Pras. Measuring the Most Complex System Ever Built, May 2013. https://www.utwente.nl/en/news/!/2013/5/164642/measuring-the-most-complex-system-ever-built.

[145] Hosnieh Rafiee, Christoph Mueller, Lukas Niemeier, Jannik Streek, Christoph Sterz, and Christoph Meinel. A Flexible Framework for Detecting IPv6 Vulnerabilities. In *Proceedings of the 6th International Conference on Security of Information and Networks*, SIN '13, 2013.

[146] Rapid7. DNS consumer hostname filtering code, 2015. https://github.com/rapid7/dap/blob/master/lib/dap/filter/names.rb#L98.

[147] Kasu Venkat Reddy. IPv6 Transition for Mobile Operators. https://www.apricot.net/apricot2011/media/Kasu_Venkat_Reddy_IPv6_Mobile_Operators_2011.pdf, February 2011.

[148] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear. RFC 1918: Address Allocation for Private Internets, 1996.

[149] RIPE. IPv4 Address Allocation and Assignment Policies for the RIPE NCC Service Region. https://www.ripe.net/publications/docs/ripe-649, 2015.

[150] RIPE NCC. Routing Information Service (RIS). http://www.ripe.net/ris/.

[151] Karen Rose, Scott Eldridge, and Lyman Chapin. The Internet of Things: An Overview. https://www.internetsociety.org/sites/default/files/ISOC-IoT-Overview-20151014_0.pdf, October 2015.

[152] C. Rossow. Amplification Hell: Revisiting Network Protocols for DDoS Abuse. In *Proceedings of the Network and Distributed System Security Symposium*, NDSS'14, 2014.

[153] Paul Saab. IPv6 is Here and You're Hurting Your Users. https://www.youtube.com/watch?v=_7rcAIbvzVY, September 2015.

[154] Top Ten Reports. https://isc.sans.edu//top10.html, 2015.

[155] Matthew Sargent, Jakub Czyz, Mark Allman, and Michael Bailey. On The Power and Limitations of Detecting Network Filtering via Passive Observation. In *Proceedings of the Passive and Active Measurement Conference*, PAM'15, 2015.

[156] Nadi Sarrar, Gregor Maier, Bernhard Ager, Robin Sommer, and Steve Uhlig. Investigating IPv6 Traffic - What Happened at the World IPv6 Day? In *Proceedings of the International Conference on Passive and Active Network Measurement*, PAM'12, 2012.

[157] Pekka Savola. Observations of IPv6 Traffic on a 6to4 Relay. *SIGCOMM Computer Communication Review*, 2005.

[158] Scans.io: Rapid7. DNS Records (ANY) Datasets, 2015. https://scans.io/study/sonar.fdns.

[159] Kyle Schomp, Tom Callahan, Michael Rabinovich, and Mark Allman. Assessing DNS Vulnerability to Record Injection. In *Proceedings of the Passive and Active Measurement Conference*, PAM'14, 2014.

[160] Wenchao Shen, Yanjiao Chen, Qianli Zhang, Yang Chen, Beixing Deng, Xing Li, and Guohan Lv. Observations of IPv6 traffic. In *Proceedings of the International Colloquium on Computing, Communication, Control, and Management*, 2009.

[161] Bruce Sinclair. Biggest risks in IPv6 security today. http://www.networkworld.com/article/2171504/tech-primers/biggest-risks-in-ipv6-security-today.html, November 2013.

[162] P. Srisuresh and M. Holdrege. RFC 2663: IP Network Address Translator (NAT) Terminology and Considerations, 1999.

[163] Mukom Akong Tamon. Why IPv6 Deployment is Slow in Africa and What to Do About It, October 2015. http://www.circleid.com/posts/20151018_why_ipv6_deployment_is_slow_in_africa_what_to_do_about_it/.

[164] The Spamhaus Project - PBL. http://www.spamhaus.org/pbl/.

[165] J Ullrich, K Krombholz, H Hobel, A Dabrowski, and E Weippl. IPv6 Security: Attacks and Countermeasures in a Nutshell. In *Proceedings of the USENIX Workshop on Offensive Technologies*, WOOT'14, 2014.

[166] University of Oregon. Route Views Project. http://www.routeviews.org/.

[167] Iljitsch van Beijnum. A Decade's Worth of IPv4 Addresses, January 2010. http://arstechnica.com/tech-policy/2010/01/dont-publish-the-decade-in-ipv4-addresses/.

[168] VigilantMinds. MS-SQL Slammer Signature. http://seclists.org/snort/2003/q1/871, January 2003.

[169] Scott Walls and Manish Karir. Internet Pollution - Part 2. Presented at the 51st North American Network Operators Grooup (NANOG51) Meeting in Miami, Florida, 2011.

[170] Yi Wang, Shaozhi Ye, and Xing Li. Understanding Current IPv6 Performance: a Measurement Study. In *10th IEEE Symposium on Computers and Communications*, 2005.

[171] Duane Wessels, Matt Larson, and Allison Mankin. Analysis of Query Traffic to .com/.net Name Servers. http://www.apricot2013.net/__data/assets/pdf_file/0006/58884/130226-com-net-query-analysis-for-apricot-2013_1361840547.pdf, 2013. Slides of a talk given at APRICOT, Feb. 19–Mar. 1, Singapore.

[172] Wikipedia. History of the Internet—Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=History_of_the_Internet&oldid=691442136, 2015. [Online; accessed 23-November-2015].

[173] Wikipedia. Netflow—wikipedia, the free encyclopedia, 2016. [Online; accessed 28-March-2016].

[174] D. Wing and A. Yourtchenko. RFC 6555: Happy Eyeballs: Success with Dual-Stack Hosts, 2012.

[175] Eric Wustrow, Manish Karir, Michael Bailey, Farnam Jahanian, and Geoff Huston. Internet Background Radiation Revisited. In *Proceedings of the ACM SIGCOMM Conference on Internet Measurement*, IMC'10, 2010.

[176] Vinod Yegneswaran, Paul Barford, and Dave Plonka. On the Design and Use of Internet Sinks for Network Abuse Monitoring. In *Proceedings of the Symposium on Recent Advances in Intrusion Detection*, RAID'04, 2004.

[177] Dan York. Facebook Moving To An IPv6-Only Internal Network. http://www.internetsociety.org/deploy360/blog/2014/06/facebook-moving-to-an-ipv6-only-internal-network/, June 2014.

[178] Sebastian Zander, Lachlan L.H. Andrew, Grenville Armitage, Geoff Huston, and George Michaelson. Mitigating Sampling Error When Measuring Internet Client IPv6 Capabilities. In *Proceedings of the ACM SIGCOMM Internet Measurement Conference*, IMC'12, 2012.

[179] Beichuan Zhang, Raymond Liu, Daniel Massey, and Lixia Zhang. Collecting the Internet AS-level Topology. *SIGCOMM Computer Communication Review*, 2005.

[180] Guoqiang Zhang, Bruno Quoitin, and Shi Zhou. Phase Changes in the Evolution of the IPv4 and IPv6 AS-Level Internet Topologies. *Computer Communications*, 34(5), April 2011.

[181] Xiaoming Zhou and Piet Van Mieghem. Hopcount and e2e Delay: IPv6 Versus IPv4. In *Proceedings of the Conference on Passive and Active Network Measurement*, PAM'05, 2005.

[182] Sebastien Ziegler, Peter Kirstein, Latif Ladid, Antonio Skarmeta, and Antonio Jara. The Case for IPv6 as an Enabler of the Internet of Things. http://iot.ieee.org/newsletter/july-2015/the-case-for-ipv6-as-an-enabler-of-the-internet-of-things.html, July 2015.