Census and Survey of the Visible Internet (extended)

USC/ISI Technical Report ISI-TR-2008-649

John Heidemann[1,2], Yuri Pradkin[1], Ramesh Govindan[2], Christos Papadopoulos[3],
Genevieve Bartlett[1,2], Joseph Bannister[4]
[1] USC/Information Sciences Institute [2] USC/Computer Science Dept.
[3] Colorado State University [4] The Aerospace Corporation

Abstract

Prior measurement studies of the Internet have explored traffic and topology, but have largely ignored edge hosts. While the number of Internet hosts is very large, and many are hidden behind firewalls or in private address space, there is much to be learned from examining the population of *visible* hosts, those with public unicast addresses that respond to messages. In this paper we introduce two new approaches to explore the visible Internet. Applying statistical population sampling, we use *censuses* to walk the entire Internet address space, and *surveys* to probe frequently a fraction of that space. We then use these tools to evaluate address usage, where we find that only 3.6% of allocated addresses are actually occupied by visible hosts, and that occupancy is unevenly distributed, with a quarter of responsive /24 address blocks (subnets) less than 5% full, and only 9% of blocks more than half full. We also show that only about 34 million (about 16% of responsive addresses) are very stable, while the rest are used intermittently, with a median occupancy of 81 minutes. Finally, we show that many firewalls are visible, measuring significant diversity in the distribution of firewalled block size. To our knowledge, we are the first to take a census of edge hosts in the visible Internet since 1982, to evaluate the accuracy of active probing for address census and survey, and to quantify these aspects of the Internet.

1 Introduction

Measurement studies of the Internet have focused primarily on network traffic and the network topology. Many surveys have characterized network traffic in general and in specific cases [27, 34, 9, 42, 15]. More recently, researchers have investigated network topology, considering how networks and ISPs connect, both at the AS [11, 45, 13, 30, 8] and router levels [46, 28]. These studies have yielded insight into network traffic, business relationships, routing opportunities and risks, and network topology.

For the most part these studies have ignored the population of hosts at the *edge* of the network. Yet there is much to be learned from understanding end-host characteristics. Today, many simple questions about hosts are unanswered: How big is the Internet, in numbers of hosts? How densely do hosts populate the IPv4 address space? How many hosts are, or could be, clients or servers? How many hosts are firewalled or behind address translators? What trends guide address utilization?

While simple to pose, these questions have profound implications for network and protocol design. ICANN is approaching full allocation of the IPv4 address space in the next few years [21]. How completely is the currently allocated space used? Dynamically assigned addresses are in wide use today [49], with implications for spam, churn in peer-to-peer systems, and reputation systems. How long is a dynamic address used by one host? Beyond addresses, can surveys accurately evaluate applications in the Internet [17]?

We begin to answer these questions in this paper. Our first contribution is to establish two new methodologies to study the Internet address space. To our knowledge, we are the first to take a complete *Internet census* of edge of the network since 1982 [40]. We also evaluate the effectiveness of *surveys* that frequently probe a small fraction of the edge of the network. We are not the first to actively probe the Internet. Viruses engage in massively parallel probing, several groups have examined Internet topology [14, 44, 19, 39], and a few groups have surveyed random hosts [17, 48]. We are the first to use controlled probing of *all* visible addresses to understand edge host behavior, and the first to compare census and survey methods. We describe our methodology in Section 2, and in Section 4 explore the trade-offs between these approaches.

Ultimately our goal is to understand all the computers on the Internet. We cannot complete this goal in this paper because active probing has inherent limitations: many hosts today are unreachable, hidden behind network-address translators and firewalls. In fact, some Internet users take public address space but use it only internally, without even making it globally routable. Figure 1 captures this complexity, highlighting in the cross-hatched area the *visible* Internet, hosts with public unicast addresses that will respond to contact. While we cannot reach this goal, our methodologies take a significant step towards it in Section 3 by measuring the vis-

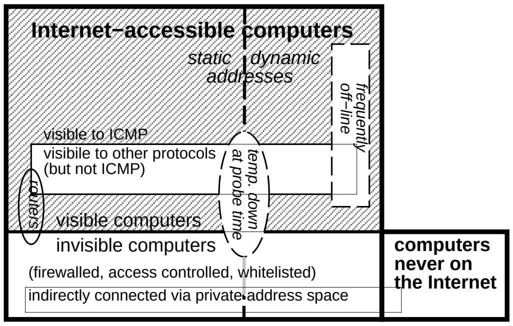

Internet−accessible computers
static
dynamic
addresses
frequently off-line
visible to ICMP
visibile to other protocols (but not ICMP)
routers
temp. down at probe time
visible computers
invisible computers
(firewalled, access controlled, whitelisted)
indirectly connected via private address space
computers never on the Internet

Figure 1: Classifying Internet addressable computers.

ible Internet and estimating specific sources of measurement error shown in this figure.

Our second contribution is to use census and survey to estimate characteristics of the Internet that have until now only been commented on anecdotally. In Section 5 we evaluate typical address occupancy, shedding light on dynamic address usage, showing that the median active address is continuously occupied for 81 minutes or less. We estimate the size of the stable Internet (addresses that respond more than 95% of the time), and show how this provides a loose upper bound on the number of servers on the Internet, overcounting servers by about a factor of two. Finally, with our three years of censuses, we show trends in address allocation and utilization and estimate current utilization. We find that only 3.6% of allocated addresses are actually occupied by visible hosts, and that occupancy is unevenly distributed, with a quarter of responsive /24 address blocks[1] less than 5% full, and only 9% of blocks more than half full.

Our final contribution is to study *trends in the deployment of firewalls* on the public Internet (Section 6). Firewalls respond to probes in several different ways, perhaps responding negatively, or not responding at all, or in some cases varying their response over time [41, 4]. Estimating the exact number of firewalls is therefore quite difficult. However, we present trends in firewalls that respond negatively over seven censuses spread over 15 months. Many such firewalls are visible and we observe significant diversity in the distribution of firewalled block size. While the absolute number of firewalled blocks appears stable, the ratio of coverage of visible firewalls to the number of visible addresses is declining, perhaps suggesting increasing use of invisible firewalls.

2 Census and Survey Methodology

Statistical population sampling has developed two tools to study human or artificial populations: *censuses*, that enumerate all members of a population; and *surveys* that consider only a sample. Our goal is to adapt these approaches to study the Internet address space. These tools complement each other, since a census can capture unexpected variation or rare characteristics of a population, while surveys are much less expensive and so can answer more focused questions and be taken more frequently. We expect censuses to capture the diversity of the Internet [35] as shown in our firewall estimates (Section 6), while surveys allow us to evaluate dynamic address usage (Section 5.1).

An Internet census poses several challenges. At first glance, the large number of addresses seems daunting, but there are only 2^{32}, and only about half of these are allocated, public, unicast addresses, so a relatively modest probe rate of 1000 probes/s (about 256kb/s) can enumerate the entire space in 49 days. Also challenging is how to interpret the results; we use censuses to study trends (Section 5.4) and firewalls (Section 6). We also must probe in a manner that is unlikely to be confused with malicious scans, and to understand the effects of lost probes on the results.

Complementing censuses, surveys avoid the problem of population size by probing a subset of addresses. Instead it poses the question of who is sampled and how often. Their primary challenge is to ensure that the sample is large enough to provide confidence in its representation of Internet, that it is unbiased, and to understand what measurement uncertainty sampling introduces.

We review these approaches next, then explore their limits and results in subsequent sections.

2.1 Probing Design

Like tools such as Nmap [36], our approaches to census and survey are forms of *active probing*. Our approaches share common choices in how probes are made and interpreted.

Requests: For each address, we send a single probe message and then record the time until a reply is received as well as any (positive or negative) reply code. We record lack of a reply after a liberal timeout (currently 5s) as a non-reply.

Several protocols could be used for probing, including TCP, UDP, and ICMP. Two requirements influence our choice. The first is *response ubiquity*—ideally all hosts will understand our probes and react predictably. Second, we desire probes that are innocuous and not easily confused with malicious scans or denial-of-service attacks.

We probe with ICMP echo-request messages because many hosts respond to pings and it is generally considered benign. We considered TCP because of the perception that it is less frequently firewalled and therefore more accurate than ICMP, but discarded it after one early census (*TCP$_1$*, Table 1) because that survey elicited thirty times more abuse complaints than ICMP surveys. We study this trade-off in Section 3.2, showing that while there is significant filtering, ICMP is a more accurate form of active than TCP.

Replies: Each ICMP echo request can result in several potential replies [23], which we interpret as following:

Positive acknowledgment: We receive an *echo reply* (type 0), indicating the presence of a host at that address.

Negative acknowledgment: We receive a *destination unreachable* (type 3), indicating that host is either down or the address is unused. In Section 6 we subdivide negative replies based on response code, interpreting codes for *network*, *host*, and *communication administratively prohibited* (codes 9, 10, and 13) as positive indication of a firewall.

[1] We use the term address *block* in preference to subnetwork because a subnet is the unit of router configuration, and we cannot know how the actual edge routers are configured.

No reply: Lack of response can have several possible causes. First, either our probe or its response could have accidentally failed to reach the destination due to congestion or network partition. Second, it may have failed to reach the destination due to intentionally discard by a firewall. Third, the address may not be occupied (or the host temporarily down) and its last-hop router may decline to generate an ICMP reply.

Only reply types 0 and 3 are usually solicited by an echo request. In addition, we receive a moderate number (30% of responses, 3% of probes) of time-exceeded replies (type 11), and a few (about 2%) other responses; we do not consider non-type 0 or 3 responses here.

Request frequency: Each run of a census or survey covers a set of addresses. Censuses have one pass over the entire Internet, while surveys make a multiple passes over a smaller sample (described below). Each pass probes each address once in a pseudo-random order.

We probe in a pseudo-random sequence so that the probes to any portion of the address space are dispersed in time. This approach also reduces the correlation of network outages to portions of the address space, so that the effects of any outage near the prober are distributed uniformly across the address space. Dispersing probes also reduces the likelihood that probing is considered malicious.

One design issue we may reconsider is retransmission of probes for addresses that fail to respond. A second probe would reduce the effects of probe loss, but it increases the cost of the census. Instead, we opted for more frequent censuses rather than a more reliable single census. We consider the effects of loss in Section 3.5.

Implementation requirements: Necessary characteristics of our implementation are that it enumerate the Internet address space completely, dispersing probes to any block across time, in a random order, and that it support selecting or blocking subsets of the space. Desirable characteristics are that the implementation be parallelizable and permit easy checkpoint and restart. Our implementation has these characteristics; we describe it in detail in Appendix A.

2.2 Census Design and Implementation

Our census is an enumeration of the allocated Internet address space at the time the census is conducted. We do not probe private address space [37], nor multicast addresses. We also do not probe addresses with last octet 0 or 255, since those are often unused or allocated for local broadcast in /24 networks. We determine the currently allocated address space from IANA [22]. IANA's list is actually a superset of the routable addresses, since addresses may be assigned to registrars but not yet injected into global routing tables [29]. We probe all allocated addresses, not just those currently routed, because routing may change over census duration as they come on-line or due to transient outages.

An ideal census captures an exact snapshot of the Internet at given moment in time, but a practical census takes some time to carry out, and the Internet changes over this time. Probing may also be affected by local routing limitations, but we show that differences in concurrent censuses are relatively small and not biased due to location in Section 3.3.

We have run censuses from two locations, one in the Western US and the other in the eastern US. Census probes run as fast as possible, limited by a fixed number of outstanding probes, generating about 166kb/s of traffic. Our western site is well provisioned, but we consume about 30% of our Internet connection's capacity at our eastern site. We have taken censuses since June 2003 and surveys since March 2006 (Table 1). (The NACK rates in two censuses, IT_{11w} and IT_{12w}, were corrected to remove around 700M NACKs generated from a single, oddly configured router.)

2.3 Survey Design and Implementation

Survey design issues include selecting probe frequency of each address and selecting the sample of addresses to survey.

How many: Our choice of how many addresses to survey is governed by several factors: we need a sample large enough to be reasonably representative of the Internet population, yet small enough that we can probe each address frequently enough to capture individual host arrival and departure with reasonable precision. We studied probing intervals as small as 5 minutes (details omitted due to space); based on those results we select an interval of 11 minutes as providing reasonable precision, and being relatively prime to common human activities that happen on multiples of 10, 30, and 60 minutes. We select a survey size of about 1% of the allocated address space, or 24,000 /24 blocks to provide good coverage of all kinds of blocks and reasonable measurement error; we justify this fraction in Section 4.2. A survey employs a single machine to probe this number of addresses. To pace replies, we only issue probes at a rate that matches the timeout rate, resulting in about 9,200 probes/second. At this rate, each /24 block receives a probe once every 2–3 seconds.

Which addresses: Given our target sample size, the next question is which addresses are probed. To allow analysis at both the address- and block-granularity we chose a clustered sample design [18] where we fully enumerate each address in 24,000 selected /24 blocks.

An important sampling design choice is the granularity of the sample. We probe /24 blocks rather than individual addresses because we believe blocks are interesting to study as groups. (Unlike population surveys, where clustered sampling is often used to reduce collection costs.) Since CIDR [12] and BGP routing exploit common prefixes to reduce routing table sizes, numerically adjacent addresses are often assigned to the same administrative entity. For the same reason, they also often share similar patterns of packet loss. To the extent that blocks are managed similarly, probing an entire block makes it likely that we probe both network infrastructure such as routers or firewalls, and edge computers. We survey blocks of 256 addresses (/24 prefixes) since that corresponds to the minimal size network that is allowed in global routing tables and is a common unit of address delegation.

We had several conflicting goals in determining which blocks to survey. An unbiased sample is easiest to analyze, but blocks that have some hosts present are more interesting, and we want to ensure we sampled unusual parts of the Internet address space. We also want some blocks to remain stable from survey to survey so we can observe their evolution over time, yet it is likely that some blocks will cease to respond, either becoming firewalled, removed, or simply unused due to renumbering.

Our sampling methodology attempts to balance these goals. We use three different policies to select which address blocks will be probed: stable/random, stable/spaced, and novel/random. Half of the blocks are selected with a stable policy, which means that we selected them when we began surveys in September 2006 and retain them in future surveys. We selected the stable set of blocks based on IT_{13w}. Of the stable set of blocks, half of those (one quarter of blocks in the entire survey) were selected randomly from all blocks that had any positive responses. This set is relatively unbiased (affected only by our requirement that the block show some positive response). The other half of stable blocks were selected to uniformly cover a range of availabilities and volitilities (approximating the A,U-values defined in Section 2.4). This half is therefore not randomly selected, but instead ensures that unusual blocks are represented in survey data, from fully-populated, always up server farms to frequently changing, dynamically-addressed areas.

The other half of blocks are selected randomly, for each survey, from the set of /24 blocks that responded in the last census. We chose this selection method to provide unbiased coverage of all the address space while making it likely that we will get responsive blocks. This preference for responsive blocks does however bias our selection to favor the actively used part of the address space, and it ensures that we will not see any "births" of newly used blocks in our survey data.

In spite of these techniques, we actually see a moderately large number (27%) of unresponsive blocks in our surveys, suggesting address usage is constantly evolving.

How long: We collect surveys for periods of about one week. This duration is long enough to capture daily cycles, yet not burden the target address blocks. We plan to expand collection to 14 days to capture two weekend cycles.

Datasets: Table 2 lists the surveys we have conducted to date, including general surveys and $\textit{ICMP-nmap}^{\text{survey}}_{\textit{USC}}$ used for validation in Section 3.2. We began taking surveys well after our initial censuses. These datasets are available from the authors on request.

2.4 Metrics

To begin to characterize the visible Internet we define two metrics: *availability* (A) and *uptime* (U). We define address availability, $A(addr)$ as the fraction of time a host at an address responds positively. We define address uptime, $U(addr)$, as the mean duration for which the address has a continuous positive response, normalized by the duration of probing interval. This value approximates host uptime, although we cannot differentiate between an address occupied by a single host and one filled by a succession of different hosts each time we probe. Thus an address that responds for the first half of the measurement period and is down the second half will have $(A,U) = (0.5, 0.5)$, while one that is up every other day for ten days of measurements will have $(A,U) = (0.5, 0.1)$. This computation assumes that each probe is representative of the address's stability for that entire census duration.

		Dur.	**Alloc.**	**ACKs**	**NACKs**	**Prohib.**
Name	**Start Date**	(days)	($\times 10^9$)	($\times 10^6$)	($\times 10^6$)	($\times 10^6$)
$ICMP_1$	2003-06-01	117	2.52	51.08	n/a	n/a
$ICMP_2$	2003-10-08	191	2.52	51.52	n/a	n/a
TCP_1	2003-11-20	120	2.52	52.41	n/a	n/a
IT_1	2004-06-21	70	2.40	57.49	n/a	n/a
IT_2	2004-08-30	70	2.40	59.53	n/a	n/a
IT_4	2005-01-05	42	2.43	63.15	n/a	n/a
IT_5	2005-02-25	42	2.43	66.10	n/a	n/a
IT_6	2005-07-01	47	2.65	69.89	n/a	n/a
IT_7	2005-09-02	67	2.65	74.40	46.52	17.33
IT_9	2005-12-14	31	2.65	73.88	49.04	15.81
IT_{11w}	2006-03-07	24	2.70	95.76	53.4*	17.84
IT_{12w}	2006-04-13	24	2.70	96.80	52.2*	16.94
IT_{13w}	2006-06-16	32	2.70	101.54	77.11	17.86
IT_{14w}	2006-09-14	32	2.75	101.17	51.17	16.40
IT_{15w}	2006-11-08	62	2.82	102.96	84.44	14.73
IT_{16w}	2007-02-14	50	2.90	104.77	65.32	14.49
IT_{17w}	2007-05-29	52	2.89	112.25	66.05	16.04

Table 1: IPv4 address space allocation (alloc.) and responses over time (positive and negative acknowledgments, and NACKs that indicate administrative prohibited), Censuses before September 2005 did not record NACKs.

		Duration	**/24 Blocks**	
Name	**Start Date**	(days)	**probed**	**responding**
IT^{survey}_{14w}	2006-03-09	6	260	217
IT^{survey}_{15w}	2006-11-08	7	24,008	17,528
IT^{survey}_{16w}	2007-02-16	7	24,007	20,912
IT^{survey}_{17w}	2007-06-01	12	24,007	20,866
$\textit{ICMP-nmap}^{\text{survey}}_{\textit{USC}}$	2007-08-13	9	768	299

Table 2: Summary of surveys conducted

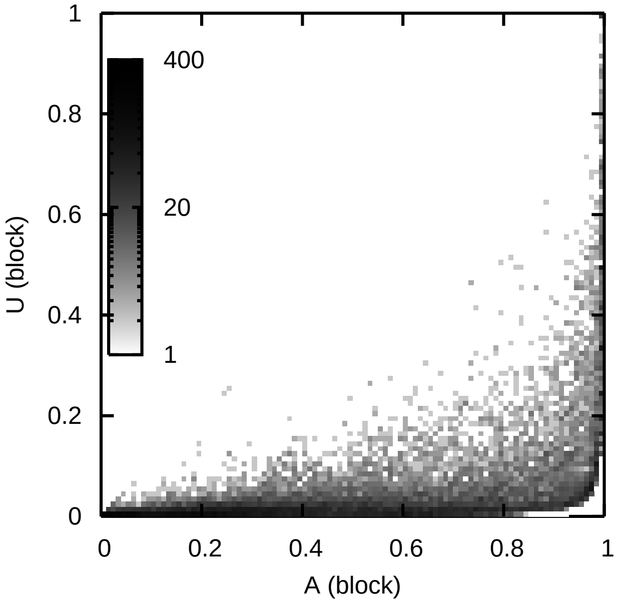


Figure 2: Density of /24 address blocks in survey IT^{survey}_{15w}, grouped by percentile-binned block availability and uptime.

We also define block availability and uptime, or $A(block)$ and $U(block)$, as the mean $A(addr)$ and $U(addr)$ for all addresses in the block that are ever responsive.

By definition, $A(block)$ is an estimate of the fraction of addresses that are up in that block. If addresses in a block follow a consistent allocation policy, it is also the probability that any responsive address is occupied.

Both A and U are defined for surveys and censuses. In censuses, the probe interval of months is sparse enough that should be considered a rough, probabilistic estimate rather than an accurate measurement. Infrequent samples are particularly problematic computing $U(addr)$ over censuses; we therefore focus on $U(addr)$ from surveys, where the sampling rate is a better match for actual host uptimes.

These measures are also not completely orthogonal, since large values of U can occur only for large values of A and small values of A correspond to small values of U. In fact, $U = A/N_U$ where N_U is the number of uptime periods. Finally, taking the mean of all addresses in a /24 block may aggregate nodes with different functions or under different administrative entities.

To illustrate these metrics and their relationship, Figure 2 shows a density plot of these values for responding blocks from IT^{survey}_{15w}. We show density by counting blocks in each cell of a 100×100 grid. Most of the probability mass is near $(A,U) = (0,0)$ and along the $U \simeq 0$ line, suggesting sparsely populated subnets where most addresses are unavailable. Additional evaluation and censuses and by address is available Appendix B.

3 Understanding the Methodology

Before evaluating the visible Internet, we first evaluate our methodology. Any form of active probing of a system as large and complex as the Internet *must* be imperfect, since the Internet will change before we can complete a snapshot. Our goal is therefore to understand and quantify sources of error, ideally minimizing them and ensuring that they are not biased. We therefore review inherent limitations of active probing, then consider and quantify four potential sources of inaccuracy: probe protocol, measurement location, multi-homed hosts, and packet loss.

Figure 1 relates what we can measure to classes of edge computers. Our methodology counts the large hatched area, and estimates most the white areas representing sources of error in our measurement. Since we have no way of observing computers that are never on-line, we focus on computers that are sometime on the Internet (the left box). This class is divided into three horizontal bands: visible computers (top cross-hatch), computers that are visible, but not to our probe protocol (middle white box, estimated in Section 3.2), and invisible computers (bottom white box; Section 3.2.1). In addition, we consider computers with static and dynamic addresses (left and right halves). Finally, subsets of these may be generally available, but down at probe time (central dashed oval; Section 3.5), frequently unavailable (right dashed box), or double counted ("router" oval; Section 3.4).

3.1 Active Probing and Invisible Hosts

The most significant limitation of our approach is that we can only see the *visible* Internet. Hosts that are hidden behind ICMP-dropping firewalls and in private address space (behind NATs) are completely missed; NAT boxes appear to be at most a single occupied address. While IETF requires that hosts respond to pings [5], many firewalls, including those in Windows XP SP1 and Vista, drop pings (although many home deployments are behind ping-responsive routers).

Characterizing the invisible Internet is a very difficult problem. To estimate measurement error, the next section characterizes what fraction of addresses are used but non-response for USC (or USC) and a random Internet sample. In Section 6 we look at visible firewall deployment. Studies of server logs, such as that of Xie *et al.* [49], may complement our approaches and can provide insight into NATed hosts since web logs of widely used services can see through NATs. Ultimately, a complete evaluation of the invisible Internet is an area of future work.

Network operators choose what to firewall and whether to block the protocols used in our probes. Blocking reduces our estimates, biasing them in favor of under-reporting usage. This bias is probably greater at sites that place greater emphasis on security. While we study the effects of firewalls and quantify that in the next section, our overall conclusions focus on the visible Internet.

3.2 Choice of Protocol for Active Probing

We have observed considerable skepticism that ICMP probing can measure active hosts, largely out of fears that it is widely filtered by firewalls. While *no* method of active probing will detect a host that refuses to answer any query, we next compare ICMP and TCP as alternative mechanisms. We validate ICMP probing by examining two populations. First, at USC we use both active probes and passive traffic observation to estimate active addresses. Because university policies may differ from the general Internet, we then compare ICMP and TCP-based probing for a random sample of addresses drawn from the entire Internet.

3.2.1 Evaluation at USC

We first compare ICMP and TCP based probing on a week-long survey $\textit{ICMP-nmap}^{\text{survey}}_{\textit{USC}}$ of the 81,664 addresses and about 50,000 students and staff at USC, comparing passive observation of all traffic with TCP and ICMP probing.

Our ICMP methodology is described in Section 2.2, with complete scans every 11 minutes. We compare this approach to TCP-based active probing and passive monitoring as described by Bartlett *et al.* [3]. TCP-based active probing uses Nmap applied to ports for HTTP, HTTPS, MySQL, FTP, and SSH, taken every 12 hours. For TCP probes, Nmap regards both SYN-ACK and RST responses as indication of host presence. Passive monitoring observes nearly all network traffic between our target network and its upstream, commercial peers. It declares an IP address active when it appears as the source address in any UDP packet or a non-SYN TCP packet. We checked for IP addresses that generate only TCP SYNs on the assumption that they are spoofed source addresses from SYN-flood attacks; we found none.

Table 3 presents this comparison. We show completeness of the three detection methods, normalized to detection by any method (middle column), and detection by any form of active probing (right column). Detection by any means

category:		any	active
addresses probed	81,664		
non-responding	54,078		
responding any	27,586	100%	
ICMP or TCP	19,866	72%	100%
ICMP	17,054	62%	86%
TCP	14,794	54%	74%
Passive	25,706	93%	
ICMP only	656		
TCP only	1,081		
Passive only	7,720		

Table 3: Comparison of ICMP, Nmap, and passive observation of address utilization at USC.

category:		active
addresses probed	1,000,000	
non-responding	945,703	
responding either	54,297	100%
ICMP	40,033	74%
TCP	34,182	62%
both ICMP and TCP	19,918	
ICMP only	20,115	
TCP only	14,264	

Table 4: ICMP-TCP comparison for random Internet addresses.

(the union of the three methods) represents the best available ground truth, but passive methods are not applicable to the general Internet, so the right column represents best-possible practical wide-area results as we use in the next section.

First, we consider the absolute accuracy of each approach. When we compare to ground truth as defined by all three methods, we see that active methods significantly undercount active IP addresses, with TCP missing 46% and ICMP missing 38%. While this result confirms that firewalls significantly reduce the effectiveness of active probing, it shows that active probing can find the majority of used addresses.

Second, we can compare the relative accuracy of ICMP and TCP as types of active probing. We see that ICMP is considerably *more* effective than TCP-based probing. While some administrators apparently regard ICMP as a security threat, others recognize its value as a debugging tool.

Our experiment used different probe frequencies for ICMP and TCP. This choice was forced because Nmap is much slower than our optimized ICMP prober. However, when we correct for this difference by selecting only ICMP surveys every 12 hours, ICMP coverage only falls slightly, to 59% of any responders, or 84% of active responders. We therefore conclude that coverage is dominated by the type of probing, not probe frequency.

3.2.2 Evaluation from a Random Internet Sample

While our dataset from USC allows us to define a strong ground truth, we cannot claim that it represents the Internet as a whole. To remove potential bias due to local or academic-specific policies, we next consider a survey of a random sample of one million allocated Internet addresses taken in October, 2007.

For each address, we send both an ICMP echo request and a TCP SYN to port 80, each within 200 microseconds. We observe the TCP reply, treating any TCP response (SYN-ACK or RST) from port 80 and a probed source address as positive response indicating the presence at that address. Some ICMP responses are also treated as positive indications, such as ICMP port 80 unreachable, ICMP protocol TCP unreachable, or ICMP host/net unreachable, provided their source is the probed address. Other ICMP replies, or replies sourced by an address other that probed are treated as a negative response confirming the absence of a host. Because we record all TCP and ICMP traffic, we do not have timeouts. To mimic our ICMP approach as closely as possible, SYN generation is done by a custom program and responses are observed through packet sniffing, so the prober's TCP stack is not involved and SYN retransmissions do not occur. Ordering of probe type was randomized in each pair, although we observe no difference in results when we examine subsets that fix ordering.

Table 4 shows the results of this experiment. If we define addresses that respond to either ICMP or TCP as ground truth of visible address usage, we can then evaluate accuracy of detection of active addresses relative to this ground truth. These results show that traffic filtering is more widespread in the Internet than at USC, since both ICMP and TCP response rates are lower (74% and 62% compared to 86% and 74% when we use the same baseline). This experiment confirms, however, that qualitatively, ICMP is more accurate than TCP-based probing, finding 74% of active addresses, 11% closer to our baseline. We conclude that *both* ICMP and TCP port 80 are filtered by firewalls, but ICMP is less likely to be filtered.

3.2.3 Implications on Estimates

We draw several conclusions from these validation experiments. First, they show that active probing considerably underestimates Internet utilization—single protocol active probing misses about one-third to one-half of all active addresses from our USC experiment. When we consider visible addresses (those that will respond to *some* type of active probe), single-protocol active probing underestimates by one-third to one-sixth of hosts from both experiments.

Our results suggest that, while hosts block one protocol or the other, multi-protocol probing can discover more active addresses than single protocol probing. The experiments also show that ICMP-only probing is consistently more accurate than TCP-only probing. Our operational experience is that TCP probing elicits 30 times more abuse complaints than ICMP. Since the resulting "please-do-not-probe" blacklists would skew results, we believe ICMP is justified as the best feasible instrument for wide-area active probing.

Returning to our Internet-wide census and survey, this evaluation suggests that our ICMP probing underestimates the number of active addresses by about 40% due to firewalls. Defining visible addresses as those that respond to any active probe, we show that our ICMP probing underestimates active visible addresses by about 25%. In either case, we conclude that our ICMP probes are *more accurate* than

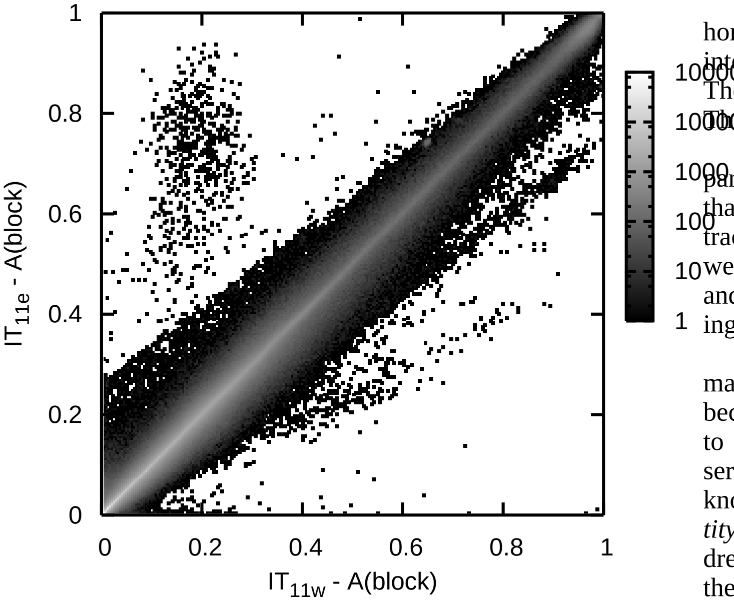


Figure 3: Subnets' A values from two censuses taken widely different network locations: IT_{11w} and IT_{11e}.

TCP would be.

3.3 Measurement Location

Measurement location is an additional possible source of bias. Our probers are all in the same place in the Internet; it may be that this location provides a poor view of parts of the Internet, perhaps due to consistently congested links or incomplete routing.

To rule out this source of potential bias, censuses since March 2006 have been done in pairs from two different locations. A "West" census IT_{11w} is taken from the ISI network in Marina del Rey, California, while an identical census IT_{11e}, is taken from the ISI's East-coast office in Arlington, Virgina. We use different seeds at each site so probe order varies, but the censuses are started concurrently. These sites have completely different network connectivity.

Figure 3 compares the $A(block)$ values measured from each vantage point in a density plot. As expected, the vast majority of blocks are near $x = y$, but for a few outliers. All metrics comparing $A(block)$ from east and west support strong stability of result independent of location: the PDF of this difference appears Gaussian, where 96% of values agree within ± 0.05, and correlation coefficient is 0.99999.

3.4 Multi-homed hosts and Routers

We generally assume that each host occupies only a single IP address, and so each responsive address implies a responsive host. This assumption is violated in two cases: some hosts and all routers have multiple public network interfaces, and some hosts use different addresses at different times.

Multiple public IP addresses for a single host are known as *aliases* in Internet mapping literature [14], and several techniques have been developed for *alias resolution* to determine when two IP addresses belong to the same host [14, 44].

One such technique is based on the fact that some multi-homed hosts or routers can receive a probe-packet on one interface and reply using a source address of the other [14]. The source address is either fixed or determined by routing. This behavior is known to be implementation-specific.

Because it can be applied retroactively, this technique is particularly suitable for large-scale Internet probing. Rather than sending additional probes, we re-examine our existing traces to find responses sent from addresses different than were probed. We carried out this analysis with census IT_{15w} and found that 6.7 million addresses responded, a surprisingly large 6.5% of the 103M total responses.

In addition to hosts with multiple concurrent IP addresses, many hosts have multiple sequential IP addresses, either because of associations with different DHCP servers due to mobility, or assignment of different addresses from one server. In general, we cannot track this since we only know *address* occupancy and not the occupying *host identity*. However, Section 5.1 suggests that occupancy of addresses is quite short. Further work is needed to understand the impact of hosts that take on multiple IP addresses over time, perhaps using log analysis from large services [49].

3.5 Probe Loss

An important limitation of our current methodology is our inability to distinguish between host unavailability and probe loss. Probes may be lost in several places: in the LAN or an early router near the probing machine, in the general Internet, or near the destination. In this section, we examine how lost probes affect observed availability and the distribution of $A(addr)$ and $A(block)$.

We minimize chances of probe loss near the probing machines in two different ways. First, we rate-limit outgoing probes to so that it is unlikely that we overrun nearby routers buffers. Second, our probers checkpoint their state periodically and so we are able to stop and resume probing for known local outages. In one occasion we detected a local outage after-the-fact, and we corrected for this by redoing the probe period corresponding to the outage.

We expect three kinds of potential loss in the network and at the far edge: occasional loss due to congestion, burst losses due to routing changes [26] or edge network outages, and burst losses due to ICMP rate-limiting at the destination's last-hop router. We depend on probing in pseudo-random order to mitigate the penalty of loss. With the highest probe rate to any /24 block of one probe every 2–3 seconds in a survey, or 9 hours for a census, rate limiting should not come into play. In addition, with a census, probes are spaced much further apart than any kind of short-term congestion or routing instability, so we rule out burst losses for censuses, leaving only random loss.

Random loss is of concern because the effect of loss is to *skew* the data towards a lower availability. This skew differs from surveys of humans where non-response is apparent, and where non-responses may be distributed equally in the positive and negative directions. Prior studies of TCP suggest we should expect random loss rates of a few percent (for example, 90% of connections have 5% loss or less [1]).

We account for loss differently in censuses and surveys. For censuses, data collection is so sparse that loss recovery is

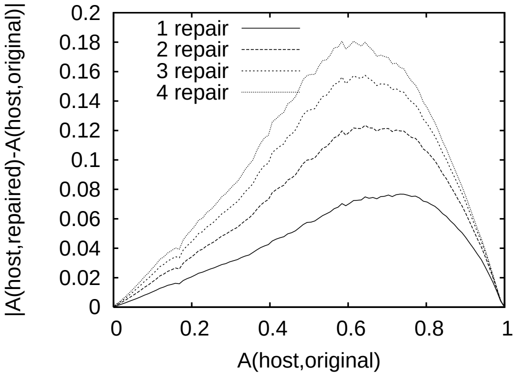


Figure 4: Distribution of differences between the k-repair estimate and non-repaired IT^{survey}_{15w}.

not possible. Instead, we reduce the effect of loss on analysis by focusing on $A(block)$ rather than $A(addr)$, since a few, random losses have less impact when averaged over an entire block. For surveys, we attempt to detect and repair random probe loss through a *k-repair process*. We assume that a random outage causes up to n consecutive probes to be lost. We repair losses of up to k-consecutive probes by searching for two positive responses separated by up to k non-responses, and replacing this gap with assumed positive responses. We can then compare $A(addr)$ values with and without k-repair; clearly $A(addr)$ with k-repair will be higher than without.

Figure 4 shows how much k-repair changes measured $A(addr)$ values for IT^{survey}_{15w}. Larger values of k result in greater changes to $A(addr)$; but the change is fairly small: it changes by at most 10% with 1-repair. We also observe that the change is largest for intermediate $A(addr)$ values (0.4 to 0.8). This skew is because in our definition of A, highly available addresses ($A(addr) > 0.8$) have very few outages to repair, while rarely available addresses ($A(addr) < 0.4$) have long-lasting outages that cannot be repaired.

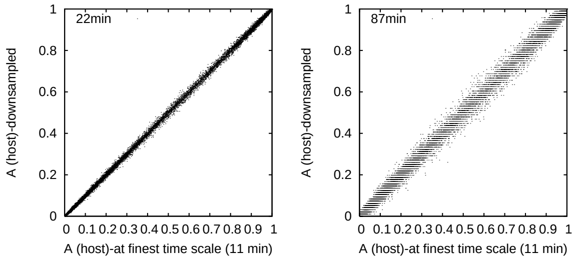
Finally, although we focused on how loss affects $A(addr)$ and $A(block)$, it actually has a stronger effect on $U(addr)$. Recall that U measures the continuous uptime of an address. A host up continuously d_0 days has a $U(addr) = 1$, but a brief outage anywhere after d_1 days of monitoring gives a mean uptime of $(d_1 + (d_0 - d_1))/2$ days and a normalized $U(addr) = 0.5$, and a second outage reduces $U(addr) = 0.33$. While k-repair reduces this effect, reductions in U caused by moderate outages are inherent in this metric.

4 Evaluating Methodology Parameters

We have described our approaches to taking a census and survey of Internet address usage. They trade off the complete spatial coverage provided by a census for covering a smaller area with finer temporal resolution with a survey. In this section we look at those tradeoffs and their basis in sampling theory, evaluating how varying temporal or spatial coverage affects our observations.

4.1 Sampling in Time

As Internet addresses can be probed at different rates, we would like to know how the probe rate affects the fidelity of our measurements. Increasing the sampling rate, while keeping the observation time constant, should give us more samples and hence a more detailed picture. However, probes that are much more frequent than changes to the underlying phenomena being measured cannot further improve accuracy (due to the Shannon sampling theorem [38]), and limited network bandwidth at the source and target argue for moderating the probe rate. Unfortunately, we do not necessarily know the timescale of Internet address usage. In this section we therefore evaluate the effect of changing the measurement timescale on our $A(addr)$ metric.



(a) 2x downsampling

(b) 8x downsampling

Figure 5: Effect of downsampling fine timescale $A(addr)$. Data from IT^{survey}_{15w}.

To examine what effect the sampling interval has on the fidelity of our metrics, we simulate different probe rates by decimating this fine timescale dataset. We treat the complete dataset as ground truth, then throw away every other sample to halve the effective sampling rate. Applying this process repeatedly gives exponentially coarser sampling intervals.

Figure 5 shows the results of two levels of downsampling for every address that responds in our fine timescale survey. In the figure, each address is shown as a dot with coordinates representing its accessibility at the finest time scale (x-axis) and also at a coarser timescale (the y-axis). If a coarser sample provided exactly the same information as finer samples we would see a straight line, while a larger spread indicates error caused by coarser sampling. We observe that this spread grows as sample interval grows. In addition, as sampling rates decrease, data collects into bands, because n probes can only distinguish A-values with precision $1/n$.

While these graphs provide evidence that sparser sampling increases the level of error, they do not directly quantify that relationship. To measure this value, we group addresses into bins based on their $A(addr)$ value at the finest timescale, then compute the standard deviation of $A(addr)$ values in each bin as we reduce the number of samples per address. This approach quantifies the divergence from our ground-truth finest timescale values as we sample at coarser resolutions. Figure 6 shows these standard deviations for a range of sample timescales, plotted by points. As expected, coarser sampling corresponds to wider variation in the measurement compared to the true value; this graph quantifies that relationship. We see that the standard deviation is the greatest for addresses with middle values of A (local maximum around $A = 0.6$) and significantly less at the extreme values of $A = 0$ and $A = 1$.

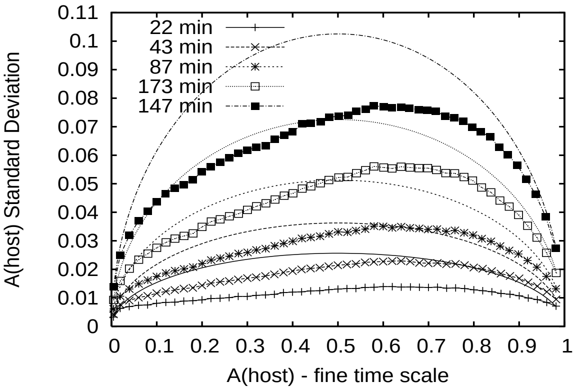


Figure 6: Standard deviation (from IT^{survey}_{15w}) as a function of ground truth $A(addr)$ metric (from IT^{survey}_{15w}) overlayed with theoretical curves $\sqrt{A(1-A)/n}$.

To place these values into context, assume for a moment that address occupancy is strictly probabilistic, and that an address is present with probability p. Thus $E(A(addr)) = p$, and each measurement can be considered a random variable X taking values one or zero when the host responds (with probability p) or is non-responsive (with probability $1-p$). With n samples, we expect np positive results, and $\hat{A}(addr)$ will follow a binomial distribution with standard deviation $\sqrt{np(1-p)}$. On these assumptions, we can place error bounds on the measurement: our estimates should be within $\hat{A}(addr) \pm 1.645\sqrt{\hat{p}(1-\hat{p})/n}$ for a 90% confidence interval; we show these estimates on Figure 6 as lines. We can see that the measured variance is nearly always below the theoretical prediction. This reduction is potentially caused by correlation in availability between hosts in same block. The prediction becomes more and more accurate as we increase the time scale and samples become more "random", approaching the binomial distribution.

These results assume our measurements are unbiased. This assumption is not strictly true, but we show in Section 3.5 that the bias is small.

4.2 Sampling in Space

We can survey an increasing number of addresses, but only at a diminishing rate. In the extreme case of our census, we probe every address only once every several months. Data so sparse makes interpretation of uptime highly suspect, because measurements are taken much less frequently than the known arrival and departure rates of hosts such as mobile computers. Much more frequent sampling is possible when a smaller fraction of the Internet is considered, however this step introduces sampling error. In this section we review the statistics of population surveys to understand how this affects our results. The formulae below are from Hedayat and Sinha [18]; we refer interested readers there.

In finding the proportion of a population that meets some criteria, such as the mean $A(addr)$ values for the Internet, we draw on two prior results of simple random sampling. First, a sample of size n approximates the true A with variance $V(\hat{A}) \simeq A(1-A)/n$ (provided the total population is large, as it is in the case of the IPv4 address space). Second, we can estimate the margin of error d with confidence $1-\alpha/2$ for a given measurement as:

$$d = z_{\alpha/2}\sqrt{A(1-A)/n} \qquad (1)$$

when the population is large, where $z_{\alpha/2}$ is a constant that selects confidence level (1.65 for 95% confidence).

Second, when estimating a non-binary parameter of the population, such as mean $A(block)$ value for the Internet with a sample of size n, the variance of the estimated mean is $V(\bar{A}(block)) = S^2_{\bar{A}(block)}/n$, where $S^2_{\bar{A}(block)}$ is the true population variance.

These results from population sampling inform our Internet measurements: by controlling the sample size we can control the variance and margin of error of our estimate. We use this theoretical result in Section 5.2 to bound sampling error at less than 0.4% for response estimates of our surveys.

5 Estimating the Size of the Internet

Having established our methodology, we now use it to estimate the size of the Internet. While this question seems simple to pose, it is more difficult to make precise. Our goal is to estimate the number of hosts that can access the Internet, yet doing so requires careful control of sources of error.

Figure 1 divides the Internet address space into several categories, and we have quantified the effects of protocol choice (Section 3.2) and invisible hosts (Section 3.2.1), our largest sources of undercounting. Section 3.4 also accounts for a overcounting due to routers.

Having quantified most sources of error, we can therefore estimate the size of the Internet through two sub-problems: estimating the number of hosts that use dynamic addresses and the number that use static addresses. We must understand dynamic address usage because dynamic addresses represent a potential source of both over- or under-counting. Dynamic addresses may be reused by multiple hosts over time, and they may go unused when an intermittently connected host, such as a laptop or dial-up computer, is offline.

Unfortunately, we cannot yet quantify hosts using dynamic addresses. The topic has only recently begun to be explored [49]; to this existing study we add an analysis of duration of address occupancy (Section 5.1). Here we focus on evaluating the size of the static, visible Internet (Section 5.2), and use this data to evaluate server usage (Section 5.3).

While we cannot quantify how many computers are *ever* on the Internet, we can define an Internet *address snapshot* as whatever computers are on-line at any instant. Our census captures this snapshot, modulo measurement error and non-instantaneous measurement time. We can then project trends in Internet address use by evaluating how snapshots change over time (Section 5.4), at least to the extent the snapshot population tracks the entire Internet host population.

5.1 Duration of Address Occupancy

We next use our address surveys to estimate how many Internet addresses are used dynamically. There are many reasons to expect that most hosts on the Internet are dynamically addressed, since many end-user computers use dynamic addresses, either because they are mobile and change addresses

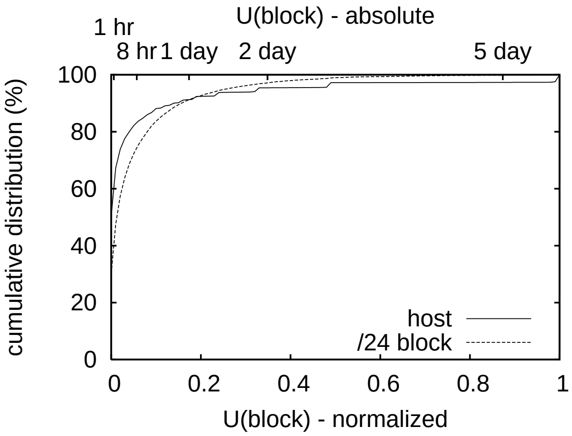


Figure 7: Duration of address occupancy: CDF of $U(addr)$ and $U(block)$ from 1-repaired Survey IT_{15w}^{survey}.

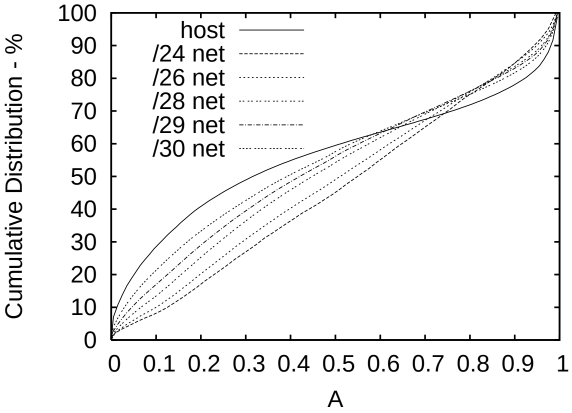


Figure 8: CDF of $A(addr)$ and $A(block)$ from from IT_{15w}^{survey}.

based on location, or because ISPs encourage dynamic addresses (often to discourage home servers, or provide static addressing as a value- and fee-added service). In addition, hosts that are regularly turned off show the same pattern of intermittent address occupation.

Figure 7 shows distribution of address and block uptimes (with 1-repair Section 3.5) from IT_{15w}^{survey}. This data shows that vast majority of addresses are not particularly stable, and are occupied only for a fraction of the observation time. We see that 50% of addresses are occupied for 81 minutes or less. A small fraction of addresses, however, are quite stable, with about 3% up almost all the time, and another 8% showing only a few (1 to 3) brief outages.

5.2 Estimating the Size of the Stable Internet

We next turn to estimating the size of the static Internet. Since we can only detect address usage or absence, we approximate the static Internet with the stable Internet. This approach underestimates the static Internet, since some hosts always use the same addresses, but do so intermittently.

We first must define stability. Figure 8 shows the cumulative density function of A values for addresses, and for different size blocks, computed over survey IT_{15w}^{survey} (others are similar). The addresses clustered at the right and left on the figure represent clear cut case stable and unstable hosts. We define addresses with 95% availability or better to be *very stable addresses*, concluding that this data suggests that 16.4% of responsive addresses in the survey are very stable.

We can next project this estimate to the whole Internet with two methods. First, we extrapolate from the survey to the census taken over the whole Internet. Our survey finds 1.75M responsive addresses in 17.5k responsive /24 blocks, suggesting a mean of 16.4 stable addresses per responsive block. The corresponding census finds 2.1M responsive blocks, suggesting an upper bound of 34.4M stable, occupied addresses in the entire Internet. This estimated upper bound depends on mapping between survey and census.

Second, we can project directly from our census. Given 103M responsive addresses in our census, we estimate that 16.4% of these, or 16.8M addresses, are potentially very stable. However, this estimate does not account for the fact that our survey was biased (by only choosing to survey previously responsive blocks, and blocks selected from a range of A, U values), and our survey is much more robust to packet loss, since each address is probed more than 916 times over a week-long survey rather than once in the three month census. We therefore consider our first estimate to be an upper bound on the size of the visible Internet.

Finally, we can list and quantify several potential sources of error in this estimate. Most importantly, we know that firewalls block access to about 38% of hosts (Section 3.2.1). Next, stable addresses could be multi-homed hosts or routers, overestimating the number of hosts (Section 3.4); we believe this represents an overcount of at most 6% of addresses. Third, some addresses were not stable because they were newly occupied mid-way through our census. We can detect births by stability after some point of time; we estimate they were less than 1% of addresses. Finally, we can place statistical bounds on our estimate due to sample size at about 0.4% (Equation 1). Taken together, these factors suggest an error-corrected estimated of 52M very stable addresses on the public Internet.

5.3 Bounding Servers in the Internet

One of our goals in evaluating stable addresses is to estimate an upper bound on the number of servers in the Internet. We hypothesized that most hosts offering services to other computers must be stable and accessible. By comparing observed address stability to observations at USC, we discovered that this assumption does not hold in today's network. While popular servers are stable, a surprising number computers offer services but are only intermittently available. We believe this result shows wide use of Internet-connected appliances and casually run servers; always-available data servers are only a majority of servers on the Internet.

To evaluate how stable addresses are used we revisit our week-long survey at USC, $\textit{ICMP-nmap}_{USC}^{\text{survey}}$. Table 5 compares the number of stable addresses to confirmed servers. We define stability as 95% responsive $A(addr)$, and compare this two alternatives: a TCP-determined server is an address that responds to active TCP probing on our five ports (HTTP, HTTPS, MySQL, FTP, and SSH) with a SYN-ACK;

category	count	vs. stable
addresses probed	81,664	
non-responding	54,078	
any	27,586	
TCP and passive	6,155	
stable ICMP	9,993	100%
stable servers	4,193	42%
stable non-servers	5,800	58%
non-stable servers	1,962	
non-stable non-servers	69,709	

Table 5: Comparison of ICMP, Nmap, and passive observation of server bounds at USC.

a passive-identified server's address appears in the source field of captured TCP SYN-ACK packets.

This evaluation shows that servers are often stable (by about a 2:1 ratio). There are also many addresses that are stable but not servers (58% of stable addresses). We assume these are client hosts are left on at all times, perhaps used in laboratories or public places. (These may represent an opportunity for energy conservation.)

We were surprised by the number of non-stable servers, though, and examined our data more closely since they represented important hosts (servers) missed in our survey. First, of the 1,962 non-stable servers, some servers may filter our ICMP probes. We classify these as 0-stability, unused addresses (with set notation, TCP-responsive \ ICMP-stable). There are 773 of these, 13% of ground truth. All of these servers will be visible to active probing with TCP, suggesting that multiple active techniques could more accurately bound on servers.

Second, a few servers do not reply to either ICMP or TCP probes yet appear in passive traces (passive \ (TCP-responsive ∪ ICMP-stable)). We expect that these servers respond only to a few whitelisted hosts. We observe 27 of these, 0.44% of ground truth. These servers will be invisible to any general active probing.

Finally, we concluded that a relatively large number of servers are simply not stable. We observe 1189 of these, 19% of ground truth. To understand this class of Internet hosts we extracted a random sample of 50 addresses and investigated them by looking at their hostnames, default web page contents, and operating system. We found that just over half of our sample were dynamically assigned addresses, about 18% were embedded devices (8 printers and 1 camera), two were default installation websites, while the rest did not respond or could not be identified. We conclude that most (78% of unstable servers, or 15% of ground truth) "servers" that are unstable are actually non-traditional servers—either dynamic hosts that happen to be running web servers, or embedded devices that are turned off at night.

From this analysis we can conclude that stable addresses can provide a very loose upper bound on the number of servers on the Internet, overestimating by a factor of about two. Assuming 42% of stable addresses are servers, and 34.4M stable addresses, or 52M stable addresses after correcting for firewalls, we suggest that there are are between 14M and 22M stable servers on the Internet.

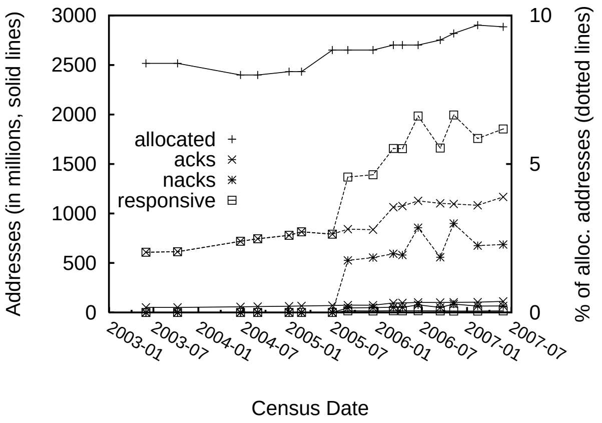


Figure 9: IPv4 address space allocation and utilization over time. Solid lines indicate absolute values, dashed are percentages of allocated addresses. (Data from all censuses.)

5.4 Trends in Internet Address Allocation

Since the IPv4 address space is finite and limited to 32 bits, the rate of address allocation is important. In fact, concerns about address space exhaustion [16] were the primary motivation for IPv6 [7] and CIDR [12] as an interim conservation strategy. They also motivated deployment of Network Address Translation (NAT) devices that allow many computers to share a single globally routable address [47]. We next consider how effective conservation of address space allocation has been 20 years after these initial studies.

Figure 9 and Table 1 show trends in address space allocation and utilization computed over each individual Internet address. To put these values into context, a total of 4.3B addresses are possible, after eliminating private and multicast address space, only 2.8 billion public unicast addresses are allocated. Finally, this evaluation represents the number of *addresses* and not actual host computers, since multiple computers may be hidden behind a single NAT box.

We see that allocation is growing at about 106M/year, (about 4%/year since 2004), and the number of visible hosts grows at 17.2M/year (28% of the 2004 baseline).

Care must be taking in interpreting these results, though, because address allocation is far from uniform. Many ISPs give out individual addresses to users, but these addresses are usually dynamic and change over time. Even users of "always-on" connections may shift addresses over time. Businesses and ISPs, on the other hand, are given addresses in power-of-two blocks, which are rarely filled.

6 Trends in Firewall Deployment

Large numbers of Internet hosts lie behind firewalls, which are configured to restrict, block or rate-limit traffic according to private local policies. Firewalls clearly affect the visibility of hosts to censuses. In this section we study trends in the deployment of visible firewalls over 15 months to begin to understand their effect on our observations.

Counting hosts behind firewalls is difficult since the goal of a firewall is often to shield hosts from external access. Measuring firewalls themselves is also difficult because many firewalls simply drop packets, making them invisible to our probing. Some firewalls, however, respond to ICMP echo requests with negative acknowledgments, indicating that communication is "administratively prohibited". We use this information to estimate the number of firewalls and firewalled addresses.

We begin with some terminology and definitions. We define a firewall as a software or hardware device that intentionally hides from our probes an active network interface that is otherwise connected to the public Internet and assigned a public IP address. (Since our focus is the public Internet, we do not attempt to count hosts behind NATs with private IP addresses.) A firewall intercepts packets before they reach their destinations. Firewalls include access-controls in routers, dedicated boxes, and end-host software. With regard to our probes, *silent firewalls* discard the probe without reply, while *visible firewalls* generate a reply that indicates communication is administratively prohibited. Access-control lists in routers are one implementation of visible firewalls. Many host operating systems include a software firewall that protects a single machine. We call these *personal firewalls*, in contrast to *block firewalls* which are typically implemented by routers, PCs or dedicated appliances and cover a block of addresses. When appropriate, we use the term firewall for all these different devices and software.

In this section, we use censuses to count the visible firewalls in the Internet, both personal and block firewalls, and estimate the address space they cover. Because we miss silent firewalls, these measurements provide only lower bounds of all firewalls. Finally, we analyze trends in firewall deployment over a 15-month period covered by censuses IT_7 through IT_{15w} (all censuses that recorded NACKs).

6.1 Methodology

To count firewalls we subdivide the negative replies to a census. We consider responses of type 3, *destination unreachable*, with code 9, 10, and 13, indicating *network*, *host* or *communication administratively prohibited*. to indicate the presence of a visible firewall. We then compare the probed address P to the source address of the reply message R. When $P = R$, the host itself replied, and so we classify P as a personal firewall. When $P \neq R$, we conclude that a block firewall with address R replied on P's behalf. We also consider a positive response (echo reply) or a negative response that is not administrative prohibited, to be a non-ICMP-firewalled address. In other cases, we cannot draw a conclusion about the presence of a firewall, since the address may be invisibly firewalled, the address may be empty, or the probe may have been lost.

To measure coverage, we examine all probed addresses P_i with the same reply address R to determine the *firewalled block* covered by firewall R. A block firewalled by R is the largest $[l,h]$ address range such that l and h elicit an administratively prohibited reply, and $\forall\ p \in [l,h]$, replies to probes to address p are either administratively prohibited from R, or a positive reply (echo reply, type 0) from p, or there is no response from p. We also require $h - l < 2^{16}$, and confirmed that this step avoids degenerate cases. This definition of firewalled blocks tolerates lost probes (by ignoring non-responses) and considers the common practice of allowing a few publicly-visible hosts (often web servers) in the middle of an otherwise firewalled range of addresses.

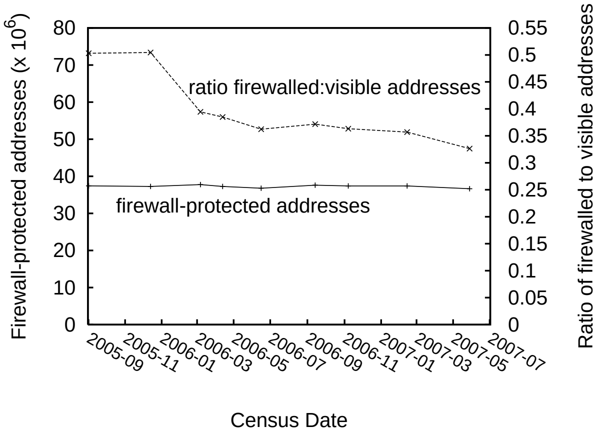


Figure 10: Number of addresses protected by visible firewalls (including personal firewalls), in absolute terms (left scale) and in ratio to visible, non-firewalled addresses. (Data from IT_7 through IT_{17w}.)

We analyze our censuses to estimate the number of firewalled addresses, the number of firewalled blocks, their distribution by size and their evolution over time.

6.2 Evaluation

We begin by considering the size of the firewalled address space. Figure 10 shows the absolute number of addresses protected by visible firewalls (left axis and bottom line), and the ratio of that count to the number of responsive addresses (right axis and top line). The number of firewalled addresses is then the sum of the size of all firewalled blocks.

We see nearly 40M addresses protected by visible firewalls. The visibly firewalled space is a very small fraction of the allocated address space (about 1.5% of 2.6B–2.8B addresses). The firewalled address space is, surprisingly, relatively stable over three years of observation. However, when we compare the ratio of addresses protected by visible firewalls to the number of responsive, non-firewalled addresses, we see a downward trend. In mid-2005, there was 1 visibly firewalled address for every 2 responsive addresses; by the end of 2006 this ratio had declined to nearly 1:3. We suspect that this trend is due to an increase in the number of invisible firewalls, but this hypothesis requires further investigation.

Turning to firewall block size, the address space covered by each firewall, we observe between 190k and 224k personal firewalls across our surveys (not shown in our figures), with no consistent trend over time. Personal firewalls greatly outnumber block firewalls, 4:1. However, the block firewalls cover more than 99% of firewalled address space.

Figure 11 shows the cumulative distribution of sizes of firewall blocks, omitting personal firewalls. We assume that the number of blocks corresponds to the number of block firewalls, although it is conceivable that a single firewall may handle multiple blocks. We see bumps at block sizes that are powers of two, with a pronounced bump at /24, but inter-

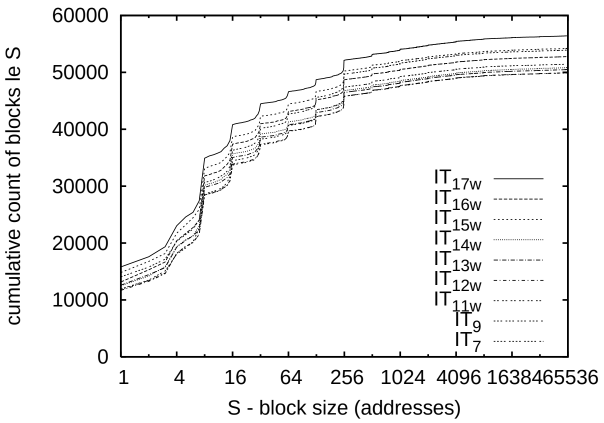


Figure 11: Cumulative distribution of firewalled blocksize. This graph omits 190–225k personal firewalls. (Data from IT_7 through IT_{17w}.)

estingly, also at /29 and /30 blocks. We also notice a slight increase in the number of blocks over the course of our study, mostly due to additional firewalls covering single addresses.

From these observations we make several conjectures about trends in firewall use. Since we see little increase in the number of firewalled hosts across our censuses, we conjecture that most newly deployed hosts are either visible, or go behind silent firewalls that our methodology is unable to account for. Given the relative stability in the number of visible firewalls, we conjecture that existing firewalls maintain visibility and most new firewalls are configured to be invisible. The latter may reflect the heightened sense of security in new deployments, while the former the inertia in changing existing configurations. Finally, this analysis confirms administrative structure in the Internet at sub-/24 block sizes, a structure hidden from previous BGP-based analysis [29].

6.3 Validation

To evaluate these observations we review data from our institution, $\textit{ICMP-nmap}_{USC}^{\text{survey}}$. First, we extracted the 35 firewalled blocks we detected in our university's network. We then confirmed these detections with our network administrators, using their knowledge of the network as ground truth. They validated that each range we detected corresponded to a router-level access control list, and therefore represents a true positive. They did not find any non-firewalls in our list, verifying that we have no false positives. In addition, they informally examined the block sizes that we discovered, suggesting that we accurately estimated the size of 24, and were off by a few addresses of six small blocks (sizes 4 to 27). For overall coverage, of the 2,674 firewalled addresses we discovered, operations confirmed 2,639 addresses as firewalled, and we were incorrect for 35 addresses. These small differences may be due to configuration changes between observation and examination.

There are two possible classes of false negatives with our methodology. First, invisible firewalls: we expect that the 7,720 passive-only addresses in Table 3 represent invisible firewalled space. Our algorithm correctly classifies these as indeterminate, but this represents a limitation of our definition. Second, there may be visible firewalls that we fail to detect. Because there is no central list of firewalls at our institution we cannot confirm that we observed all visible firewalls, frequent probing makes omission due to loss unlikely.

While this validation is based on a single enterprise, these results seem quite promising.

7 Related work

To our knowledge there has been no attempt to take a full Internet census since 1982 [40]. Smallberg's census in 1982 was aided by a independent, central enumeration of all hosts; our approach instead enumerates all possible IP addresses.

We are aware of only one other active survey of addresses. Robin Whittle surveyed the Internet address space, randomly pinging about 0.1% the routed space over 24 hours in March 2007 [48]. Projecting from the positive replies, he estimated about 107M responsive addresses, within a few percent of our census report of 103M in IT_{15w} four months earlier. His results corroborate ours with a methodology like our surveys.

He *et al.* use random sampling of addresses to study web content [17]. They study the open web while we study address usage in visible Internet. Our study of methodology may aid understanding of the accuracy of this type of survey.

An alternative way to enumerate the Internet is to traverse the domain name system [24]. While an important complement to our work, many hosts on the Internet lack names, and some named hosts are not on the Internet. Earlier surveys in this series estimated host counts by pinging randomly chosen IP addresses. However, the Internet has grown significantly since these surveys and it is unlikely that their results hold today. Our coverage is also more complete, allowing, for example, the study of firewall usage.

Closest to our methodology of active probing are several projects that measure Internet connectivity, including Rocketfuel [44], Mercator [14], Skitter [19], and Dimes [39]. The primary goal of these projects is to estimate the macroscopic, router-level connectivity of the Internet, a valuable but complementary goal to ours. These project therefore do not exhaustively probe edge-hosts in IP address space, but instead use tools such as traceroute to edge addresses to collect data about routers that make up the middle of the Internet.

Several other efforts use different approaches to also study properties of the IP address space. First, Meng *et al.* use BGP routing tables to study IPv4 address allocation at the block level [29]. Like ours, this work is a longitudinal study of address space allocation, they consider seven years of data. However, their approach considers only block-level information gathered from IANA and injected into the global routing tables, not a host-level study, and they consider only new blocks, not the entire IPv4 address space. Our edge study also reveals sub-/24 structure invisible to BGP.

As another example, Kohler *et al.* [25] studied the structure of IP destination addresses seen through passive observations on Internet links. Their measurements were conducted at a few locations that included access links to universities, ISP routers with local peerings, and a major ISP's backbone routers. Their data collection considered several links, each measured for several hours, observing between

70,000 and 160,000 addresses. They discover multifractal properties of the address structure and propose a model that captured many properties in the observed traffic. By contrast, our census unearthed upwards of 50 million distinct IP addresses through active probing of addresses and so focuses more on the static properties of address usage rather than their dynamic, traffic dependent properties.

Finally, Narayan *et al.* propose a model of IP routing tables based on allocation and routing practices [31] , and Huston [20] and Gao *et al.* [6] (among others) have measured the time evolution of BGP tables and address space. This work focuses on BGP and routing, not the the temporal aspects of address space usage that we consider.

Because compromised home private machines are the source of a significant amount of unsolicited e-mail, a number of anti-spam organizations maintain lists of dynamically assigned addresses (examples include [43, 32]). This work complements our study of the behavior of dynamic addresses, but uses primarily static or manually entered data, or semi-automated probing in response to spam.

Recent research has explored how to detect dynamic address space usage by examining login rates to a major on-line e-mail hosting service [49]. As with our work they characterize IP address usage, however their methodology is based on passive monitoring of a large web service. Their work complements ours in that they can reach strong conclusions about the addresses that contact their service, and they can peer behind NATs in ways we cannot, but they cannot evaluate addresses that do not contact their service, limiting their coverage to some subset of client computers in the Internet.

Much of the previous work on firewall detection has focused on discovering stealth firewalls. Early work was published on the Phrack website [10], detecting firewalls that did not verify checksums. Tools such as p0f [33] and nmap [36] have options to detect a firewall either by passively monitoring flows or actively sending specially crafted packets and analyzing responses. These tools are more accurate than our approach, but much more invasive, thus we can safely study the whole Internet.

8 Future Work and Conclusions

There are several directions for future work, including refining the methodology, (particularly considering loss), exploring time/space trade-offs, and improving our understanding of the visible Internet and characterization hosts and addresses hidden to active probing.

This paper is the first to show that censuses can walk the entire IPv4 address space, and surveys of about 1% of that space, and to begin to quantify sources of measurement error in these complementary approaches. Our preliminary application of this methodology shows trends and estimates of address space utilization and deployment of visible firewalls. However, we expect our methodology and datasets to broaden the field of Internet measurements from routers and traffic to the network edge.

Acknowledgments

This material is based on work partially supported by the United States Department of Homeland Security contract number NBCHC040137 (”LANDER”). John Heidemann is partially supported by the National Science Foundation (NSF) under grant number CNS-0626696, “NeTS-NBD: Maltraffic Analysis and Detection in Challenging and Aggregate Traffic (MADCAT)”. All conclusions of this work are those of the authors and do not necessarily reflect the views of DHS or NSF.

The authors would like to thank Tom Lehman and Forrest Houston at ISI East for hosting our probing machines there. We thank Los Nettos staff for their support of this work; help from Jim Pepin (now at Clemson), Maureen Dougherty, Walt Prue, and Sandy George, has been essential. Mark Baklarz and Luke Sheppard assisted in providing ground truth for our firewall validation. We also thank ISI’s Information Processing Center, particularly Christine Smiley and Drew Jongbloed, for handling first-round complaints about why we are pinging people. We thank Jelena Mirkovic and Robin Whittle for multiple careful readings of drafts.

9 References

[1] M. Allman, W. M. Eddy, and S. Ostermann. Estimating loss rates with TCP. *ACM Performance Evaluation Review*, 31(3):12–24, Dec. 2003.

[2] Anonymous. Tech. report omitted due to anonymization., 2007.

[3] G. Bartlett, J. Heidemann, and C. Papadopoulos. Understanding passive and active service discovery. In *Proc. of the ACM Internet Measurement Conference*. ACM, Oct. 2007.

[4] R. Beck. Passive-aggressive resistance: OS fingerprint evasion. *The Linux Journal*, Sept. 2001.

[5] R. Braden. Requirements for Internet hosts—communication layers. RFC 1122, Internet Request For Comments, Oct. 1989.

[6] T. Bu, L. Gao, and D. Towsley. On characterizing BGP routing table growth. Proc. of the IEEE Global Internet, November 2002.

[7] S. Deering and R. Hinden. Internet protocol, IP version 6 specification. RFC 2460, Internet Request For Comments, Dec. 1998.

[8] X. Dimitropoulos, D. Krioukov, M. Fomenkov, B. Huffaker, Y. Hyun, kc claffy, and G. Riley. AS relationships: Inference and validation. *ACM Computer Communication Review*, 37(1):29–40, Jan. 2007.

[9] N. Duffield and M. Grossglauser. Trajectory sampling for direct traffic observation. In *Proc. of the ACM SIGCOMM Conference*, pages 179–191, Stockholm, Sweeden, Aug. 2000. ACM.

[10] Ed3f. Firewall spotting and networks analysis with a broken CRC. `http://www.phrack.org/archives/60/p60-0x0c.txt`, Dec. 2002.

[11] M. Faloutsos, P. Faloutsos, and C. Faloutsos. On power-law relationships of the internet topology. In *Proc. of the ACM SIGCOMM Conference*, pages 251–262, Cambridge, MA, USA, Sept. 1999. ACM.

[12] V. Fuller, T. Li, J. Yu, and K. Varadhan. Classless inter-domain routing (CIDR): an address assignment and aggregation strategy. RFC 1519, Internet Request For Comments, Sept. 1993.

[13] L. Gao. On inferring automonous system relationships in the internet. *ACM/IEEE Transactions on Networking*, 9(6):733–745, Dec. 2001.

[14] R. Govindan and H. Tangmunarunkit. Heuristics for Internet Map Discovery. In *Proc. of the IEEE Infocom*, March 2000.

[15] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy, and J. Zahorjan. Measurement, modelling, and analysis of a peer-to-peer file-sharing workload. In *Proc. of the 19th Symposium on Operating Systems Principles*, pages 314–329. ACM, Oct. 2003.

[16] T. Hain. A pragmatic report on IPv4 address space consumption. *The Internet Protocol Journal*, 8(3), 2004.

[17] B. He, M. Patel, Z. Zhang, and K. C.-C. Chang. Accessing the deep web. *Communications of the ACM*, 50(5):94–101, May 2007.

[18] A. S. Hedayat and B. K. Sinha. *Design and Inference in Finite Population Sampling*. John Wiley & Sons, Inc., 1991.

[19] B. Huffaker, D. Plummer, D. Moore, and K. C. Claffy. Topology Discovery by Active Probing. In *Proc. of the Symposium on Applications and the Internet*, January 2002.

[20] G. Huston. Analyzing the Internet's BGP routing table. *Internet Protocol Journal*, 4(1), March 2001.

[21] G. Huston. IPv4 address report. `http://bgp.potaroo.net/ipv4/`, June 2006.

[22] Internet Assigned Numbers Authority. Internet protocol v4 address space. web page `http://www.iana.org/assignments/ipv4-address-space`, Sept. 2002.

[23] Internet Assigned Numbers Authority. ICMP type numbers. web page `http://www.iana.org/assignments/icmp-parameters`, Mar. 2007.

[24] Internet Software Consortium. Internet domain survey. web page `http://www.isc.org/ds`.

[25] E. Kohler, J. Li, V. Paxson, and S. Shenker. Observed structure of addresses in IP traffic. In *Proc. of the 2nd ACM Internet Measurement Workshop*, pages 253–266, Nov. 2002.

[26] C. Labovitz, A. Ahuja, A. Abose, and F. Jahanian. Delayed Internet routing convergence. In *Proc. of the ACM SIGCOMM Conference*, pages 175–187, Stockholm, Sweeden, Aug. 2000. ACM.

[27] W. Leland, M. Taqqu, W. Willinger, and D. Wilson. On the self-similar nature of Ethernet traffic (extended version). *ACM/IEEE Transactions on Networking*, 2(1):1–15, Feb. 1994.

[28] L. Li, D. Alderson, W. Willinger, and J. Doyle. A first-principles approach to understanding the Internet's router-level topology. In *Proc. of the ACM SIGCOMM Conference*, pages 3–14, Aug. 2004.

[29] X. Meng, Z. Xu, B. Zhang, G. Huston, S. Lu, and L. Zhang. IPv4 address allocation and the BGP routing table evolution. *ACM Computer Communication Review*, 35(1):71–80, Jan. 2005.

[30] W. Mühlbauer, O. Maennel, S. Uhlig, A. Feldmann, and M. Roughan. Building an AS-topology model that captures route diversity. In *Proc. of the ACM SIGCOMM Conference*, pages 195–204, Sept. 2006.

[31] H. Narayan, R. Govindan, and G. Varghese. On the impact of routing and address allocation on the structure and implementation of routing tables. In *Proc. of the ACM SIGCOMM Conference*, Aug. 2003.

[32] NJABL. Not just another bogus list. `http://www.njabl.org/`, 2007.

[33] p0f Project. p0f passive OS fingerprinting. `http://lcamtuf.coredump.cx/p0f.shtml`, Sept. 2006.

[34] V. Paxson. End-to-end Internet packet dynamics. *ACM/IEEE Transactions on Networking*, 7(3):277–292, June 1999.

[35] V. Paxson and S. Floyd. Why we don't know how to simulate the Internet. In *Proc. of the 29th SCS Winter Simulation Conference*, pages 1037–1044, Atlanta, Georga, USA, Dec. 1997.

[36] N. Project. Nmap network security scanner. `http://www.insecure.org/nmap/`, 1997.

[37] Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear. Address allocation for private internets. RFC 1918, IETF, Feb. 1996.

[38] C. E. Shannon. Communication in the presense of noise. *Proceedings of the Institute of Radio Engineers*, 37(1):10–21, Jan. 1949.

[39] Y. Shavitt and E. Shir. Dimes: let the Internet measure itself. *SIGCOMM Comput. Commun. Rev.*, 35(5):71–74, 2005.

[40] D. Smallberg. Who talks TCP? RFC 832, Internet Request For Comments, Dec. 1982.

[41] M. Smart, G. R. Malan, and F. Jahanian. Defeating TCP/IP stack fingerprinting. In *Proc. of the USENIX Security Symposium*, pages 229–240, Denver, Colorado, USA, Aug. 2000. USENIX.

[42] F. D. Smith, F. Hernandez, K. Jeffay, and D. Ott. What TCP/IP protocol headers can tell us about the web. In *Proc. of the ACM SIGMETRICS*, pages 245–256, Cambridge, MA, USA, June 2001. ACM.

[43] SORBS. Sorbs dynamic user and host list. `http://www.au.sorbs.net/faq/dul.shtml`, 2004.

[44] N. Spring, R. Mahajan, D. Wetherall, and T. Anderson. Measuring ISP topologies with Rocketfuel. *ACM/IEEE*

Transactions on Networking, 12(1):2–16, 2004.

[45] L. Subramanian, S. Agarwal, J. Rexford, and R. H. Katz. Characterizing the Internet hierarchy from multiple vantage points. In *Proc. of the IEEE Infocom*, pages 618–627, June 2002.

[46] H. Tangmunarunkit, R. Govindan, S. Jamin, and S. S. W. Willinger. Network Topology Generators: Degree-Based vs. Structural. In *Proceedings of ACM SIGCOMM*, pages 188–195, Pittsburgh, PA, 2002.

[47] P. F. Tsuchiya and T. Eng. Extending the IP Internet through address reuse. *ACM Computer Communication Review*, 23(1):16–33, Jan. 1993.

[48] R. Whittle. Probing the density of ping-responsive-hosts in each /8 IPv4 prefix and in different sizes of BGP advertised prefix. Web page `http://www.firstpr.com.au/ip/host-density-per-prefix/`, Nov. 2007.

[49] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldszmidt, and T. Wobber. How dynamic are IP addresses? In *Proc. of the ACM SIGCOMM Conference*, Kyoto, Japan, Aug. 2007. ACM.

A Implementation Details

Addition to our description of methodology (Section 2), we next describe several important aspects of the implementation.

We have implemented our census taker as a simple C++ program. It implements rate limiting by maintaining a fixed array of currently outstanding probe control blocks (PCBs). PCBs time out after a fixed controlled interval (5 seconds) and are replaced by newly constructed PCBs with different destination addresses. Thus, the effective probe rate is determined by the ratio of the PCB array size (currently 1200 entries) to the timeout. A scheduler paces probes within the time-out interval to avoid bursts.

We use the ICMP identifier field to match ICMP replies with PCBs. ICMP sequence numbers are used as indices in the PCB array for fast PCB lookups. A vast majority of replies are matched by this mechanism, but in some cases (remote ICMP implementations that do not echo the source message and sequence number) we resort to searching by IP address. We have also experimented with sending a 32-bit random cookie in the body of ICMP message to identify the probe, but many ICMP implementations do not return this cookie in the ICMP response.

Our census taker must enumerate the entire address space in an order that touches adjacent addresses at very different times. Our current implementation (in use since IT_{11}) uses a three-step algorithm. First, it enumerates all 32-bit addresses in order, guaranteeing completeness. To disperse probes to any given subnet across time, we bit-reverse this address, so that any n-bit block is probed only once every 2^{32-n} probes. Each census also exclusive-ors the series with an arbitrary constant, ensuring that each follows a different absolute order. This algorithm can be checkpointed with only 64 bits of state, and it parallelizes easily (currently over four machines). Finally, we exclude addresses that are unallocated or that opt out of our study by filtering addresses against a blacklist stored as a balance balanced binary tree.

We repeat censuses every three months or so. Since March 2006, each each census has been conducted concurrently from two sites, one on the east and west coasts of the United States. Each site uses four probing machines, all connected to a single Ethernet segment. The aggregate bandwidth required for our probes and responses is approximately 166kb/s. The Internet connection at the western site is well overprovisioned, but we consume about 30% of the Internet connection capacity at the east coast site.

Our basic survey software implementation is almost identical to that used for conducting a census, but nearly all of the address space is filtered. As an optimization, rather than do this filtering each pass, we compute it once and record the randomized probe order. Thus the prober can simply replay the probes as fast as feasible, limited by a fixed number of outstanding probes (to cap internal state) and a selected maximum probe rate (to cap bandwidth consumption).

A survey probes address approximately every 11 minutes, so a given block could see bursts of up to 254 probes. To reduce this effect, we pace probes across the 11 minute window, so any particular /24 block will typically see a probe once every 2–3 seconds.

B Comparing Coarse and Fine Timescale Measurements

Extending our evaluation comparisons of time- and space-tradeoffs in probing (Section 4), we next compare coarse and fine-timescale measurements.

A significant difference between our census and surveys is the timescale of measurement: a census probes a given address every 3 months, while a survey every 11 minutes. Thus while it makes sense to treat a survey's consecutive probes of the same address as a timeseries, it is more difficult to evaluate evolution across censuses because long-term host changes (renumbering and host birth and death) are significant. In addition, loss repair is not generally possible

However, we can compare a concurrent census and survey to gain some validation of their accuracy. Because $A(addr)$ is poorly defined for a single census, we compare $A(block)$ for /24 blocks in IT_{11w} and IT_{11w}^{survey}.

To compare census and survey, we arrange all blocks by increasing $A(block)^{survey}$ computed from 1-repaired survey data. Since this survey represents 916 probes of each address spread over one week, we consider this as ground truth. We then group subnets that have similar $A(block)^{survey}$ values, gathering 254 integral "bins" with about that number of responsive hosts in the block. Finally we calculate the corresponding $A(block)^{census}$ from census data for the same subnet. In each $A(block)^{survey}$ bin we therefore get some number of similar $A(block)^{census}$. From these values we plot the mean and 90% confidence intervals of $A(block)^{census}$.

This comparison is shown for IT_{15w} and IT_{15w}^{survey} in Figure 12. Ideally the means should match the diagonal and confidence intervals should be zero. We see a reasonable match (the correlation coefficient is 0.74). The values are close for blocks with lower availability ($A < 0.5$), but we see that the census under-estimates $A(block)$ value for higher availability

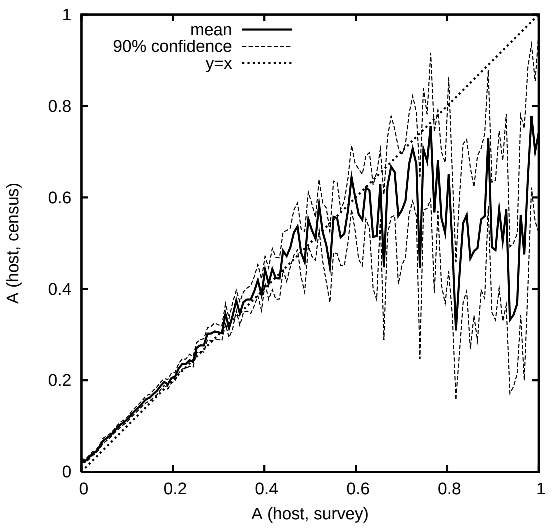


Figure 12: Comparison of $A(block)$ for coarse and fine time scale data for IT_{15w} and 1-repaired IT_{15w}^{survey}.

blocks.

We believe the match is poorer for large A values because there are many stable blocks with only one or two stable hosts. If a census misses one of these hosts due to probe loss, that block will show very high error. Two other potential causes are that survey IT_{15w}^{survey} lasted only 6 days, from Wednesday through Tuesday. It may be that more hosts are more frequently unavailable on weekends. A final possibility is that our survey's probe rate of 1 probe every 2–3 seconds is too high and is triggering block-level ICMP rate-limiting.

Because census estimates of $A(block)$ are relatively sparse, we had some concern that they might be overly altered by loss. From this comparison we conclude that block-level estimates are quite similar from both a census and a survey, providing confidence in the accuracy of $A(block)^{census}$

C Alternative (A,U) Evaluations

Section 2.4 introduced the metrics of *availability* (A) and *uptime* (U) to characterize addresses and blocks, and Figure 2 showed the distribution of survey address blocks across the (A,U)-plane. Here we show two alternative evaluations: census address blocks, and survey individual addresses. These alternatives help evaluate the sensitivity of these metrics to probe interval and degree of averaging.

C.1 Census Blocks on the (A,U)-Plane

We first consider valuations of $(A(block),U(block))$ over census data. As described in Section 2.4, computation of $U(block)$ over census data can be problematic because the sampling timescale is much coarser than the duty cycle of dynamic addresses, so we wish to see if this metric is at all similar to more accurate computations over survey data.

Figure 13 shows the (A,U) graph for the first 15 censuses. This figure considers only blocks that have some address that respond positively at some point. In fact, the majority of blocks are non-responsive, and so 8,256,560 blocks should appear at $(A=0,U=0)$.

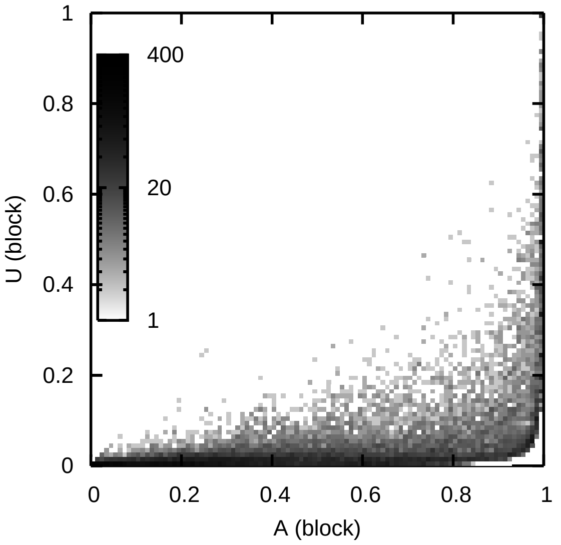


Figure 13: Density of all affirmative /24 Internet address blocks from IT_1 through IT_{15w}, grouped by percentile-binned block availability and uptime.

Despite this, this figure suggests some aspects of the Internet address space utilization. First, the vast majority of blocks are lightly utilized with low uptime, near $(A,U) = (0,0)$. However, a few blocks are heavily utilized and always up (near $(1,1)$) Manual examination suggests that these blocks near $(1,1)$ represent server farms, typically hosting many different web and mail sites. Second, blocks with a medium value of A get pulled apart, where larger U values suggest blocks with servers that turned on mid-way through our census, while smaller U values suggest blocks blocks where hosts come and go frequently.

Many hosts follow the $A=U$ diagonal. These hosts correspond to a single uptime occurrence, whether it's a server that is always up, or a non-responsive host that replied only once in all 14 censuses.

Finally, we can compare this figure to block-averaged results over survey data in Figure 2. In spite of the large difference in probing interval, these show several similarities. Both show most of the probability mass near $(0,0)$, with a few hosts tending towards $(1,1)$, and a void around $(A,U) = (1,0)$. The largest difference is the strong trend with census data towards the $A=U$ diagonal, while survey data shows that lower utilizations actually have much shorter uptimes (away from the diagonal). This difference reflects the statistical nature of very coarse probe intervals.

C.2 Survey Addresses on the (A,U)-Plane

Averaging over blocks may obscure the behavior of individual hosts. We therefore next use survey data to evaluate (A,U) for addresses instead of blocks,

Figure 14 shows a density plot for $(A(addr),U(addr))$ computed over one survey (IT_{15w}^{survey} and counted across a 100×100 grid. It is useful to compare it to a block-level survey plot (Figure 2) and our block-level census plot (Figure 13).

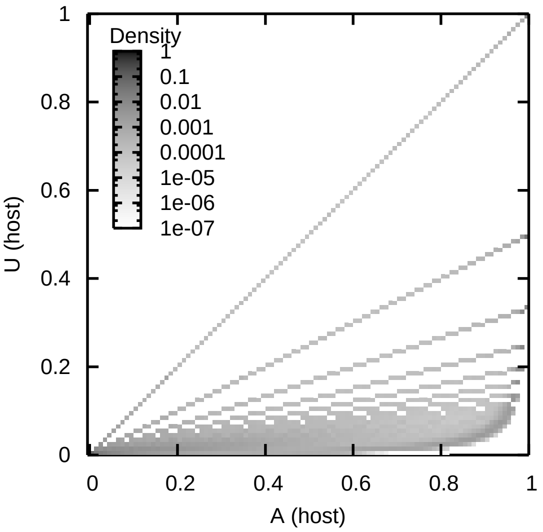


Figure 14: Density of all responding Internet addresses from IT^{survey}_{15w}, grouped by percentile-binned address availability and uptime.

The general probability mass in host-level analysis of survey data closely matches our block-level survey analysis (Figure 2), with most of the probability mass in $(0,0)$, but also large numbers of hosts along the $U = 0$ axis and some near $A = 1$.

The main new feature not seen in block-averaged survey data is the presence of a significant number of hosts along the $A = U$ diagonal, and, in general, on the $U = A/(n+1)$ diagonals for n a non-negative integer. These diagonals highlight the relationship between A and U, where the gap between $(A,U) = (1,1)$ and $(1,0.5)$, and, in general, between (d,d) and $(d, \frac{d}{(n+1)})$ follows because a single outage halves the U value of an otherwise continuously up host, and n outages track the $\frac{d}{(n+1)}$ diagonal. We partially correct for outages due to packet loss through loss repair (Section 3.5), but outages longer than our repair duration (presumably periods of true downtime) still cause this relationship. This feature was also present in census block data (Figure 13), but is obscured when block-level averaging is applied to survey data.