Towards Better Internet Citizenship: Reducing the Footprint of Internet-wide Scans by Topology Aware Prefix Selection

Johannes Klick
Freie Universität Berlin
johannes.klick@fu-berlin.de

Stephan Lau
Freie Universität Berlin
stephan.lau@fu-berlin.de

Matthias Wählisch
Freie Universität Berlin
m.waehlisch@fu-berlin.de

Volker Roth
Freie Universität Berlin
volker.roth@fu-berlin.de

ABSTRACT

Internet service discovery is an emerging topic to study the deployment of protocols. Towards this end, our community periodically scans the entire advertised IPv4 address space. In this paper, we question this principle. Being good Internet citizens means that we should limit scan traffic to what is necessary. We conducted a study of scan data, which shows that several prefixes do not accommodate any host of interest and the network topology is fairly stable. We argue that this allows us to collect representative data by scanning less. In our paper, we explore the idea to scan all prefixes once and then identify prefixes of interest for future scanning.

Based on our analysis of the *censys.io* data set (4.1 TB data encompassing 28 full IPv4 scans within 6 months) we found that we can reduce scan traffic between 25-90% and miss only 1-10% of the hosts, depending on desired trade-offs and protocols.

1. INTRODUCTION

Fast Internet-wide scanning is growing in popularity among researchers. At the time of writing, researchers regularly scan the Internet for vulnerable SSL certificates [6, 12], SSH public keys [10], and for the banners of plain text protocols such as SMTP, HTTP, FTP, and Telnet [5]. The majority of researchers scan at least 2.8 billion addresses advertised in the IPv4 address space [5–8, 10–12, 15, 16, 19]. Hitrates, the fraction of probed addresses from which a response is received, are very often under two percent [7]. This means that most scan traffic is overhead. Most of these scans are done periodically for trend analyses, which exacerbates the amount of unnecessary scan traffic. For example, the ongoing Internet-wide research project *censys.io* [5,7] probes the IANA allocated address space for 19 protocols on a continuous basis. This results in 72.2 billion generated IP-packets per week. which causes several hostile responses ranging from threatening legal actions to conducted denial-of-service attacks [7]. Whereas scanning the IPv4 address space is feasible this is not any more the case for IPv6. When IPv6 becomes more popular, we need scanning strategies that limit scans to parts of the address space that are in use.

Many measurement scenarios require only partial scans instead of exploring the full IP address space. However, we currently lack a systematic understanding of the deployment of Internet services with respect to IP address ranges.

In this paper, we want to start the discussion how we can reduce scan traffic systematically. We present the *Topology Aware Scanning Strategy (TASS)*, a new IP prefix based and topology aware scanning strategy for periodic scanning. *TASS* enables researchers to collect responses from 90-99% of the available hosts for six months by scanning only 10-75% of the announced IPv4 address space in each scan cycle (protocol dependent). *TASS* is seeded with the results of a full advertised IPv4 address scan for a given protocol and time period. The prefixes for all responses will be selected for periodic scans of the given protocol.

Periodic scanning of only selected prefixes reduces scan traffic significantly while hitting most of the hosts of interest. For instance, our analysis reveals that responsive prefixes obtained from a full FTP scan cover

IMC 2016, November 14 - 16, 2016, Santa Monica, CA, USA

 ISBN 978-1-4503-4526-2/16/11…$15.00

DOI: http://dx.doi.org/10.1145/2987443.2987457

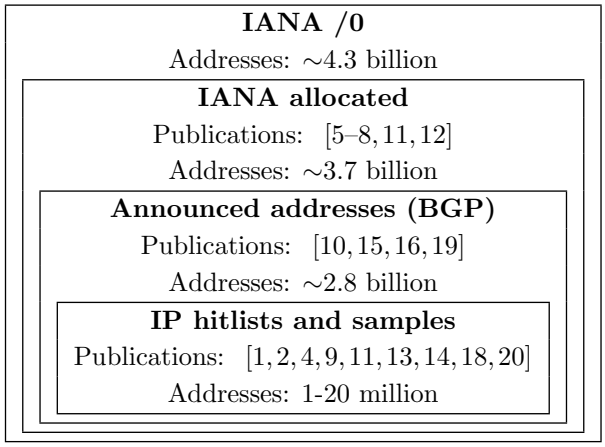


Figure 1: Current scanning strategies and their scoping of the IPv4 address space.

98% of all FTP hosts 6 months later, at the cost of scanning only 57.4% of the advertised addresses. The scanning overhead can be optimized further by omitting prefixes with a low *density*. Here, *density* denotes the fraction of hosts per address space size. For example, if we limit prefix selection to a 95% coverage of the responsive addresses then we can still find 92.3% of the FTP hosts after six months while scanning only 20.6% of the announced address space. Moving forward we plan to investigate whether the distributions of found and missed hosts are the same.

For our evaluation of *TASS* we use 4.1 TB of data derived from 28 full IPv4 scans obtained from *censys.io* [5]. For common protocols we show that, following an initial scan of the full IPv4 address space, the hitrate for responsive prefixes decreases by about 0.3 percent per month compared to what a full scan would find.

Consequently, periodical *TASS* scans are 1.25 to 10 times more efficient for a period of at least 6 months if researchers accept a single-digit percentage reduction in host coverage.

2. STATE OF THE ART

TASS represents a trade-off between scanning overhead and results accuracy. In what follows, we review the kinds of trade-offs other researchers have made previously. We identified three kinds of approaches in the literature: (*i*) full scans of IANA allocated addresses, (*ii*) scans of routable addresses and (*iii*) scans of address space samples.

IANA allocated address space. The most basic approach is to scan all IP addresses covered by the /0 prefix. Scans of this type seek to explore the reachability of all potential hosts. However, some (unicast) addresses do not offer public services per definition, for example, private networks addresses and the loopback addresses. Excluding these unallocated or reserved addresses is the first obvious step towards a reduction of scanning noise. This has been a common practice from the beginning [6, 8, 11, 12] and is still being practiced [5, 7].

Announced IP addresses (BGP). The second type of trade-off involves only addresses that are covered in global BGP tables [10, 15, 16, 19].

IP hitlists and samples. Several researchers sampled parts of the IPv4 address space in order to extrapolate from their data. For example, Alt *et al.* [1] scanned for honeypots by probing at least one host in all /24 blocks of the Internet. Rossow [18] used a random sample of 1 million IP addresses in his research on traffic amplification threats. Heidemann *et al.* [11] probed 1% of the address space repeatedly, which consisted of 24,000 /24 blocks. These blocks were compiled based on three different selection strategies: (*i*) 50% were selected randomly, (*ii*) 25% were selected if a host in this block was responsive before, and (*iii*) 25% were selected by other policies. This approach does not discriminate between prefixes of different sizes and therefore it does not utilize potentially important topology information.

Sampling leads to a reduction of scan traffic, but is less suited for research that requires precise statistics. Whereas samples tend to be probabilistic, hitlists are compiled based on predetermined characteristics. Fan and Heidemann [9] generated IP address hitlists by scanning the IPv4 address space repeatedly and by filtering out addresses that were consistently responsive. Their approach was applicable to only a third of the Internet, though, and exhibited 40-50% fluctuation after three months, probably caused by dynamic IP addresses. By comparison, *TASS* compiles prefix hitlists and exhibits only 1-10% fluctuation after six months. Dynamic IP addresses fluctuate within a particular prefix, which may explain why *TASS* is significantly more stable.

Cai and Heidemann [2] investigated the responsiveness of /24 blocks (note the difference from /24 network *prefixes*). They probed 1% of the Internet address space by selecting /24 blocks that were responsive to ICMP probes, as shown by a prior census of all allocated addresses. They clustered blocks with adjacent addresses and similar network behavior, and found that a fifth of the /24 blocks had a utilization less than 10%.

Plonka *et al.* [17] used passive IPv6 measurements of WWW clients for identification of stable and dense IPv6 prefixes. We use active scan data and focus on hosts, but we are aware that a combination of both approaches might lead to a more comprehensive solution.

Summary. The objectives of most of the measurement studies do not require a priori scanning of unreachable address space. The state of the art in Internet scanning appears to base trade-offs primarily on IP blocks and individual IP addresses. We are not aware of attempts other than ours to leverage network prefix responsiveness for scan traffic reduction.

3. TOPOLOGY-AWARE SCANNING

In this section, we give a high-level overview over TASS, followed by an empirical motivation why TASS is a promising trade-off between scanning overhead and accuracy. An evaluation of TASS performance over time is given in Section 4. We used the FTP, HTTP, HTTPS, and CPE WAN Management Protocol (CWMP). For brevity, we provide graphs primarily for FTP and HTTPS.

3.1 TASS in a Nutshell

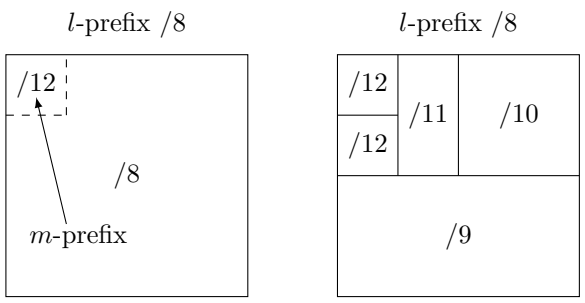
TASS amortizes the overhead of an initial scan of the full routable address space over repeated scans that cover only a subset of all prefixes. The core idea of *TASS* is to identify prefixes which are of primary interest when scanning the Internet repeatedly. The goal of *TASS* is to be *efficient*. Efficiency is measured as the number of successful protocol handshakes per number of connection attempts. *TASS* is parameterized by an adjustable target ratio ϕ that specifies the proportion of hosts that *TASS* shall cover in repeated scans. For this reason, we refer to ϕ as the host *coverage*. *TASS* works as follows:

1. At time t_0, perform a full scan and output all responsive addresses. Let N be their number. Count the number of responsive addresses c_i in each responsive prefix i. The sum of all c_i is N.
2. Calculate the density $\rho_i = c_i/2^{32-\text{prefix length}}$ of all responsive prefixes and their relative host coverage $\phi_i = c_i/N$ of responsive addresses.
3. Sort the prefixes in the descending order of density. Relabel prefixes so that $i < j \Leftrightarrow \rho_i > \rho_j$.
4. Find the smallest k so that $\sum_{i=1}^{k} \phi_i > \phi$.
5. Scan prefixes $1, \ldots, k$ repeatedly until time $t_0+\Delta_t$, then start over at step 1.

Within each time interval $[t_0, t_0 + \Delta_t]$ there will be a gradual loss of results accuracy as hosts leave or enter prefixes other than prefixes $1, \ldots, k$. On each full scan, full accuracy is recovered. We motivate in the remainder of this section why we expect this strategy to yield high accuracy with significantly reduced scan overhead. We evaluate the strategy in Section 4 and quantify the expected loss of accuracy over time, which yields an adjustable time period Δ_t.

3.2 Prefix derivation

TASS requires that addresses are mapped to prefixes. The *censys.io* dataset already contains prefix information that Durumeric *et al.* [5] apparently obtained from their outgoing AS. However, closer inspection reveals that the included information is often coarse-grained or even missing. For this reason, we chose to use the Routeviews Prefix-to-AS mappings (pfx2as) provided by CAIDA [3] instead. Said mappings reflect a topological view of the Internet, are fine-grained, and are used routinely for research.



(a) Announced prefixes. **(b) Resulting m-prefixes.**

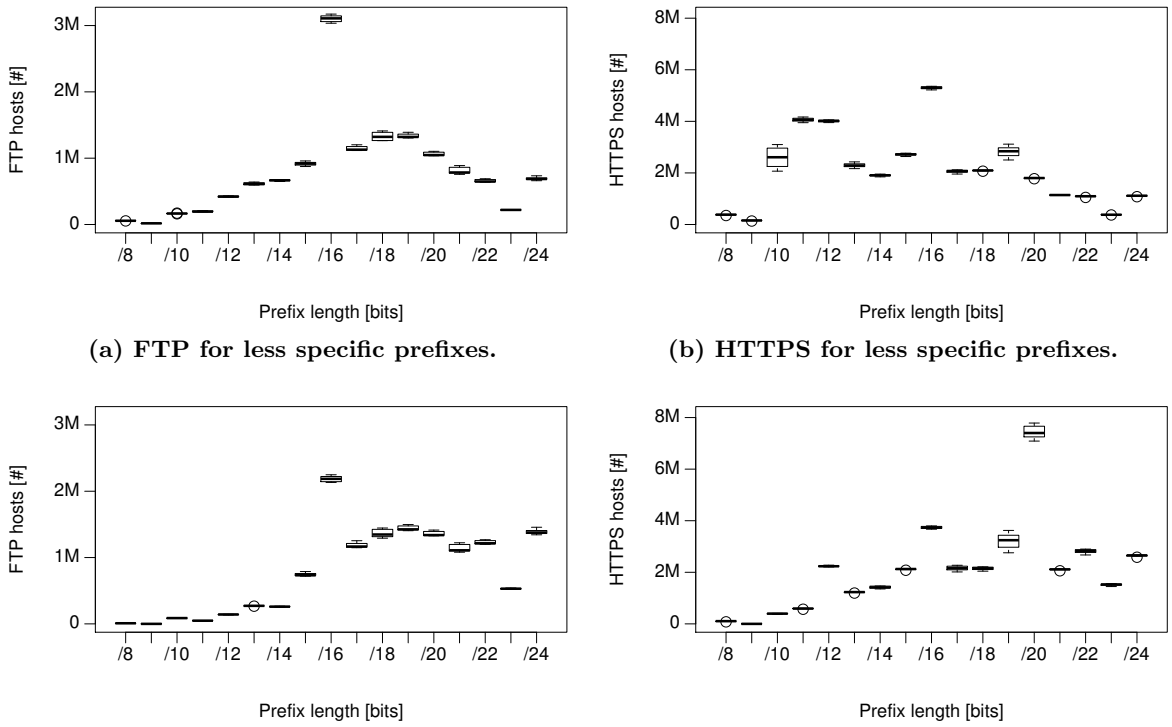
Figure 2: The less specific l-prefix /8 contains the more specific m-prefix /12. The l-prefix is then decomposed into the more specific one and the remaining smallest prefixes.

It is worth noting that prefixes in BGP may be loosely aggregated. In particular, more specific prefixes (m-prefixes, e.g. 100.0.0.0/12) may be announced in parallel to less specific prefixes (l-prefixes, e.g. 100.0.0.0/8). The CAIDA data includes a large fraction of more specific prefixes in addition to less specific prefixes. For example, the dataset of 2015/09/07 contains 595,644 prefixes of which 54% are m-prefixes. The m-prefixes account for 34.4% of the advertised IP space.

To reflect potential network characteristics, we deaggregate the l-prefix of each m-prefix into the minimal set of prefixes that contains the m-prefix. This approach allows us to take all routing information into account while maintaining a proper partition of the address space for scanning purposes. See Figure 2 for an illustration of this process. In the following two sections we show that this approach potentially reduces the number of scanned addresses.

3.3 Host stability versus prefix length

We expect that TASS performs well if hosts do not fluctuate significantly in between prefixes. In a first step, we analyzed the distribution of host numbers across prefixes of different lengths over a period of six months with 7 measurements. If the distribution variance was high then this would already indicate that TASS may miss hosts. Figures 3(a) and 3(b) show the results for the case of l-prefixes, and Figures 3(c) and 3(d) show the results for the case of m-prefixes. The host numbers appear to be stable and therefore the results do not contradict our expectation. Of course, this result is necessary but not sufficient by itself. We still need to investigate the fluctuation in between prefixes of the same length. This is future work that we intend to do with a larger dataset and for a full paper. The graphs also indicate a right-shift towards longer prefixes without a pronounced loss of stability. This lends support to our hypothesis



(a) FTP for less specific prefixes.

(b) HTTPS for less specific prefixes.

(c) FTP for more specific prefixes.

(d) HTTPS for more specific prefixes.

Figure 3: Shows the host distribution over prefix lengths based on seven different measurements from 09/2015 to 03/2016. Prefixes longer than /24 are negligible and have been omitted.

that m-prefixes are a better choice than l-prefixes because their density is potentially higher. For example, if all hosts in an l-prefix cluster in an m-prefix then the l-prefix minus the m-prefix need not be scanned.

3.4 Prefix density

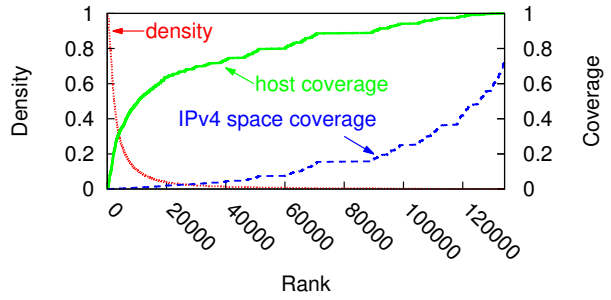
TASS yields a favorable trade-off if small reductions in coverage lead to large reductions in scan overhead. Towards an evaluation of the potential trade-off, we analyzed the density of prefixes in relation to the advertised address space. Recall that the density of a prefix is the number of responsive hosts in the prefix divided by the number of addresses in the prefix. Figures 4(a) and 4(b) show the results for the case of l-prefixes, and Figures 4(c) and 4(d) show the results for the case of m-prefixes. The graphs are sorted in the order of decreasing prefix density ρ, the red curve. Prefixes with zero density are not included. The green curve is the *cumulative relative host coverage* ϕ. The blue curve is the *cumulative relative address space coverage.* The graphs show a sharp decrease of prefix density combined with a sharp increase in host coverage and a modest increase of address space coverage over the range of prefixes. This clearly indicates that prefix selection based on prefix density is well suited to maximize the efficiency of scans. Based on the data we analyzed we can report the following statistics for the case of l-prefixes:

- 100% ($\phi = 1$) of all FTP hosts are found in ∼134 K prefixes representing 76.2% of the routed address space.
- 95% ($\phi = 0.95$) of all FTP hosts are found in ∼105 K prefixes representing 27.3% of the routed address space.
- 23.8% of the addresses were unresponsive.
- The first 20 K prefixes with a density of $\rho > 0.04$ contain 64% ($\phi = 0.64$) of all FTP servers but represent only 2% of the advertised address space.

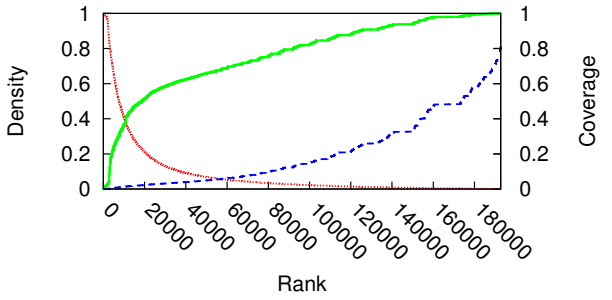
For m-prefixes, an address space coverage of 57.4% suffices to achieve full host coverage ($\phi = 1$), which is a reduction of 18.8 percentage points compared to l-prefixes. At the same time, prefix selection based on density is roughly twice as efficient as a full scan, for the FTP protocol. If one tolerates a 5% loss of FTP hosts then scanning 20.6% of the address space suffices to find 95% of the FTP hosts that a full scan would find. For detailed information, see Table 1.

4. ACCURACY OVER TIME

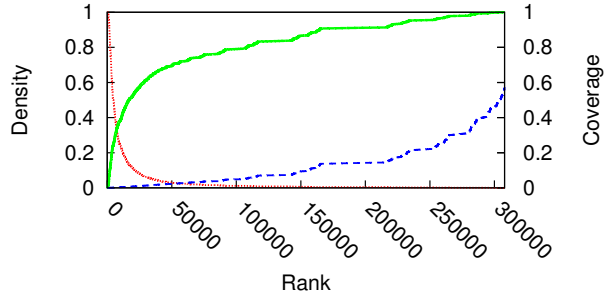
The findings we summarized in previous sections suggest that TASS can be an efficient scanning strategy. However, the benefits manifest only if the distribution of hosts across prefixes remains reasonably stable over



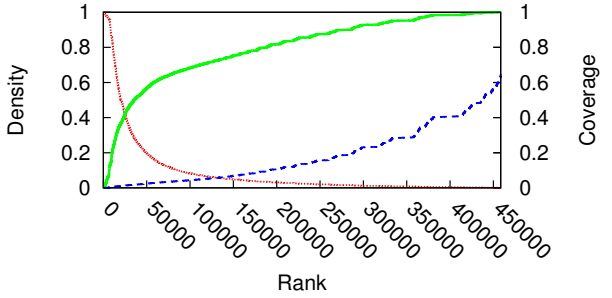
(a) FTP: Less specific prefixes.



(b) HTTP: Less specific prefixes.



(c) FTP: More specific prefixes.



(d) HTTP: More specific prefixes.

Figure 4: Shows responsive prefixes ranked by their density (dotted), the cumulative relative host coverage (solid), and the cumulative relative address space coverage (dashed) with density $\rho > 0$.

time. As a first step to quantify the accuracy of TASS over time we simulated TASS and an address-based hitlist approach using monthly snapshots of full IPv4 scans from *censys.io* [5] for the time period from 09/2015 to 03/2016 (7 snapshots). Then we determined the fraction of hosts that TASS and the hitlist approach would have uncovered in each scan cycle compared to a periodic full scan. We used the aforementioned datasets as our ground truth, again. We focused our analysis on four protocols, which were FTP, HTTP, HTTPS and TR-069 also known as the CPE WAN Management Protocol (CWMP), a 4.1 TB dataset in total. CWMP is used for remote management of residential gateways. We chose CWMP for contrast because its purpose differs markedly from the other two protocols.

Address Space Coverage	ϕ	FTP	HTTP	HTTPS	CWMP
less	1	0.762	0.828	0.832	0.477
	0.99	0.470	0.548	0.542	0.142
	0.95	0.273	0.362	0.343	0.099
	0.7	0.031	0.064	0.065	0.043
	0.5	0.008	0.021	0.024	0.024
more	1	0.574	0.648	0.645	0.332
	0.99	0.371	0.440	0.427	0.113
	0.95	0.206	0.279	0.262	0.085
	0.7	0.023	0.048	0.052	0.037
	0.5	0.006	0.017	0.020	0.021

Table 1: IPv4 address space coverage of the protocols using less and more specific prefixes.

4.1 Hitlist accuracy over time

The hitlist approach we simulated takes all addresses that are responsive in an initial full scan and subsequently scans only those addresses. This strategy exhibits maximal efficiency and accuracy for stable (unchanging) host distributions. Figure 5 show the results of our simulation. They indicate that the accuracy of the hitlist approach quickly drops to 80% within one month and continues to decrease over time for FTP, HTTP and HTTPS. The drop is much more pronounced for the CWMP protocol. A likely explanation is that residential gateways are connected to the Internet via dynamic IP addresses more often. Over the course of

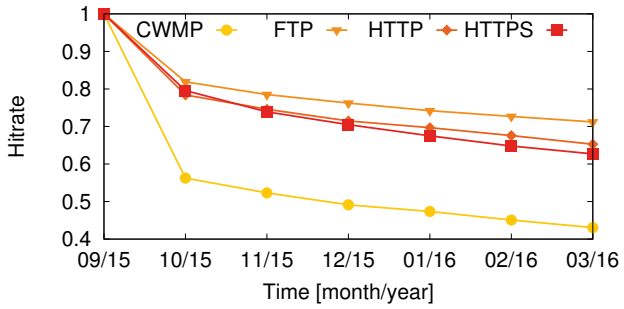
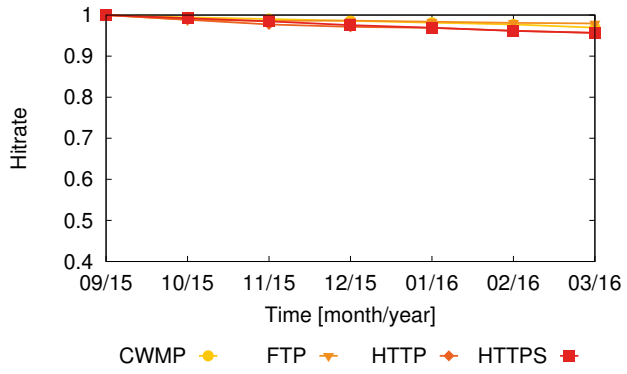
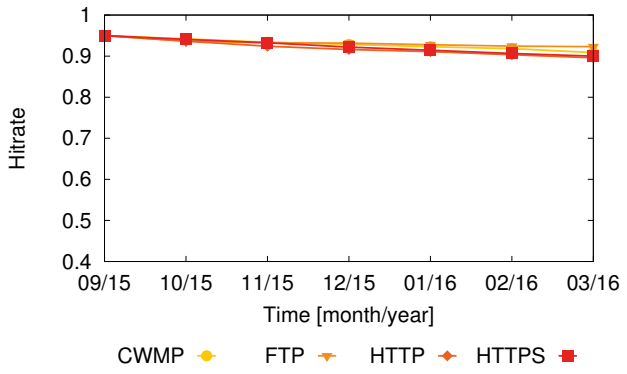


Figure 5: Hitrate using IP hitlists.



(a) Hitrate with $\phi = 1$.



(b) Hitrate with $\phi = 0.95$.

Figure 6: Hitrate of TASS compared to a full scan.

six months, the accuracy drops to 71% for HTTP and to 43% for CWMP. From these results we conclude that the hitlist approach is not recommendable for periodic scanning over time periods of several months.

4.2 TASS accuracy over time

We simulated TASS with l-prefixes and m-prefixes as described prior over the same six months time period. Figure 6(a) shows the results for a coverage setting of $\phi = 1$, that is, full host coverage. Recall that this selects all prefixes with a non-zero density, that is, $\rho > 0$. We found that accuracy decreases at a rate of 0.3% per month for l-prefixes. For m-prefixes, accuracy decreases at a rate of up to 0.7% per month or about 4.2% over the course of six months. The greater efficiency of m-prefixes is thus paid for by an accuracy reduction twice as much as for l-prefixes. We repeated our analysis for a host coverage setting of 95%, that is, $\phi = 0.95$. This reduced the accuracy further to 90-94%, depending on the protocol. Figure 6(b) summarizes the outcomes. We started a similar investigation of SSH and selected SCADA protocols but to our surprise we found that accuracy and densities increased over time. Further scrutiny of the ground truth datasets revealed that the snapshots for these protocols likely included data from prior scans. We have notified the main contributor of *censys.io* [5] who acknowledged the problem.

5. DISCUSSION AND FUTURE WORK

Our results indicate that *less* specific prefixes yield greater scanning accuracy over time than *more* specific prefixes. A likely cause is that l-prefixes reduce the overall number of prefixes, which renders it less likely that a host fluctuates in between prefixes. On the other hand, l-prefixes have a higher scanning overhead compared to m-prefixes. For a full host coverage setting ($\phi = 1$), the overhead differed by about 15-20 percentage points according to our analysis in Section 3.4. Consequently, we must consider this trade-off when deciding between l-prefixes and m-prefixes.

Likewise, the host coverage setting ϕ has a significant influence on the scanning overhead. Even a small reduction of host coverage, say from $\phi = 1$ to $\phi = 0.99$, results in a reduction of scan overhead by 20-30%. As part of our future work we intend to investigate more closely how the 1% of missed hosts are distributed in comparison to the other hosts.

Furthermore, in the context of the analysis of security incidents (e.g., *Heartbleed*) it is important to analyse whether vulnerable servers are distributed equally across both selected prefixes and omitted prefixes, for $\phi < 1$. If the distribution was fairly equal then regular estimates of vulnerable populations could be obtained with good efficiency and accuracy, for example, with $\phi = 0.5$ and a small address space coverage of 0.6-0.8% per scan cycle.

Finally, we suspect that more fine-grained prefixes may help to reduce the scanning overhead even further. Towards this end, it may be worthwhile to apply the clustering approach of Cai and Heidemann [2] to network prefixes. At any rate we are eager to investigate other data sets, additional protocols and distribution patterns for longer periods of time.

6. CONCLUSION

Fast Internet-wide scanning is an emerging topic for investigators who wish to conduct network research based on up-to-date real world data. This will likely lead to a proliferation of scanning activities. Projects such as *censys.io* already help to curtail the resulting scan traffic by making current datasets available to the Internet community for research purposes. However, there will always be objectives that call for individual data collection. The activities of corporations and individuals must be factored in as well because tools for fast Internet scanning are widely available. It is desirable to research and develop tools that tax the address space and the patience of scan targets more sparingly than brute force. With TASS, we hope to make progress towards the right direction: a scanning strategy that is

more efficient, without loosing significant accuracy of the results.

Our initial investigations are promising. By selecting prefixes for periodic scanning according to density and by adjusting host coverage, it is feasible to address a wide range of trade-offs. Particularly, small compromises with regard to host coverage can reduce scan overhead substantially, for four protocols that we investigated thus far (FTP, HTTP, HTTPS, and CWMP). TASS opens up a variety of options for further research. When IPv6 becomes popular, brute forcing the address space becomes infeasible. By then we ought to have better approaches for network scanning. Perhaps TASS can offer a blueprint for tackling that challenge as well.

Acknowledgements

We thank the anonymous reviewers, our shepherd, Roya Ensafi, and Dave Plonka for their helpful comments. We would also like to thank the *censys.io* team for having provided the measurement infrastructure and for their timely replies to our inquiries. Furthermore, we thank Carsten Schäuble and Christian Salzmann from the IT service of the CS department of our university for the compute power they provided for our analysis effort. This work was partly funded by grants from the German Federal Ministry of Education and Research (BMBF Grants No. 16KIS0254, 16KIS0528K).

7. REFERENCES

[1] L. Alt, R. Beverly, and A. Dainotti. Uncovering Network Tarpits with Degreaser. In *Proc. of ACM ACSAC*, 2014.

[2] X. Cai and J. Heidemann. Understanding Block-level Address Usage in the Visible Internet. *Proc. of ACM SIGCOMM*, 2011.

[3] Center for Applied Internet Data Analysis. Routeviews Prefix to AS mappings Dataset for IPv4 and IPv6. https://www.caida.org/data/routing/routeviews-prefix2as.xml.

[4] D. Cicalese, J. Augé, D. Joumblatt, T. Friedman, and D. Rossi. Characterizing IPv4 Anycast Adoption and Deployment. In *Proc. of ACM CoNEXT*, 2015.

[5] Z. Durumeric, D. Adrian, A. Mirian, M. Bailey, and J. A. Halderman. A Search Engine Backed by Internet-Wide Scanning. In *Proc. of ACM CCS*, 2015.

[6] Z. Durumeric, J. Kasten, M. Bailey, and J. A. Halderman. Analysis of the HTTPS Certificate Ecosystem. In *Proc. of ACM IMC*, 2013.

[7] Z. Durumeric, E. Wustrow, and J. A. Halderman. ZMap: Fast Internet-wide Scanning and Its Security Applications. In *Proc. of USENIX Security*, 2013.

[8] P. Eckersley and J. Burns. An Observatory for the SSLiverse. https://www.eff.org/files/defconssliverse.pdf, 2010.

[9] X. Fan and J. Heidemann. Selecting Representative IP Addresses for Internet Topology Studies. In *Proc. of ACM IMC*, 2010.

[10] O. Gasser, R. Holz, and G. Carle. A deeper understanding of SSH: results from Internet-wide scans. In *Proc. of IEEE NOMS*, 2014.

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

[12] N. Heninger, Z. Durumeric, E. Wustrow, and J. A. Halderman. Mining Your Ps and Qs: Detection of Widespread Weak Keys in Network Devices. In *Proc. of USENIX Security*, 2012.

[13] R. Holz, L. Braun, N. Kammenhuber, and G. Carle. The SSL Landscape – A Thorough Analysis of the X.509 PKI Using Active and Passive Measurements. In *Proc. of ACM IMC*, 2011.

[14] H. K. Lee, T. Malkin, and E. Nahum. Cryptographic Strength of SSL/TLS Servers: Current and Recent Practices. In *Proc. of ACM IMC*, 2007.

[15] D. Leonard and D. Loguinov. Demystifying Service Discovery: Implementing an Internet-Wide Scanner. In *Proc. of ACM IMC*, 2010.

[16] A. Nappa, Z. Xu, M. Z. Rafique, J. Caballero, and G. Gu. CyberProbe: Towards Internet-Scale Active Detection of Malicious Servers. In *Proc. of ISOC NDSS*, 2014.

[17] D. Plonka and A. Berger. Temporal and Spatial Classification of Active IPv6 Addresses. In *Proc. of ACM IMC*, 2015.

[18] C. Rossow. Amplification Hell: Revisiting Network Protocols for DDoS Abuse. In *Proc. of ISOC NDSS*, 2014.

[19] The Shadowserver Foundation. Open Resolver Scanning Project. https://dnsscan.shadowserver.org/.

[20] S. Yilek, E. Rescorla, H. Shacham, B. Enright, and S. Savage. When Private Keys are Public: Results from the 2008 Debian OpenSSL Vulnerability. In *Proc. of ACM IMC*, 2009.