On Network-level Clusters for Spam Detection

Zhiyun Qian[1], Z. Morley Mao[1], Yinglian Xie[2], Fang Yu[2]
[1]University of Michigan and [2]Microsoft Research Silicon Valley

Abstract

IP-based blacklist is an effective way to filter spam emails. However, building and maintaining individual IP addresses in the blacklist is difficult, as new malicious hosts continuously appear and their IP addresses may also change over time. To mitigate this problem, researchers have proposed to replace individual IP addresses in the blacklist with IP clusters, e.g., *BGP clusters. In this paper, we closely examine the accuracy of IP-cluster-based approaches to understand their effectiveness and fundamental limitations. Based on such understanding, we propose and implement a new clustering approach that considers both network origin and DNS information, and incorporate it with SpamAssassin, a popular spam filtering system widely used today. Applying our approach to a 7-month email trace collected at a large university department, we can reduce the false negative rate by 50% compared with directly applying various public IP-based blacklists without increasing the false positive rate. Furthermore, using honeypot email accounts and real user accounts, we show that our approach can capture 30% - 50% of the spam emails that slip through SpamAssassin today.*

1 Introduction

With over 90% to 97% of all emails being spam [4], spam filtering remains critical to today's email systems. There are two main categories of spam filtering techniques: content-based and blacklist-based. While content-based filtering is the canonical way, the blacklist-based approach is receiving much attention recently because it does not always rely on email content and can be more efficient and less susceptible to evasion. All widely-used blacklists (*e.g.,* Spamhaus, Spamcop [9, 8]) today rely on IP addresses to block email traffic originated from hosts with consistent spamming behavior.

While IP-based blacklist is simple and lightweight, compiling and maintaining such lists is challenging—hosts may change IP addresses over time; more hosts may be compromised and existing compromised hosts may be patched. Therefore most IP blacklists today provide a very limited coverage of all malicious IPs [25].

Rather than constructing blacklists based on individual IP addresses, previous work has studied building blacklists based on IP clusters, *e.g.,* clustering using BGP prefixes [28]. By identifying a range or a cluster of IP addresses within the same administrative boundary, we are able to construct the reputation for the entire cluster instead of individual IP addresses. The cluster-based reputation allows one to infer the reputation of IP addresses never previously observed.

In this paper, we thoroughly analyze the effectiveness of IP-cluster-based blacklists for spam detection. In particular, we explore the following questions:

- Under what scenarios do IP-clusters work? And how much coverage improvement can we obtain from IP clusters compared with an individual-IP based scheme?
- What is the right granularity for IP-clusters and how to obtain such clusters with accurate cluster boundaries?

To answer the above questions, we thoroughly studied three different clustering approaches: BGP-based, DNS-based, and combined clusters. We select these clusters because they all reveal the administrative boundaries of IP addresses. BGP clusters are constructed from the routing perspective, while DNS clusters are from the Web/email relay perspective. In particular, we propose to examine the *reverse authoritative name server* (rANS) and reverse DNS (rDNS) names for DNS clusters, because they are configured by the IP address owners and cannot be easily modified by spammers. As BGP and DNS clusters can form complex bipartite graphs, the combined clusters capture their intersections and thus are more fine-grained.

Based on these observations, we further propose and implement a combined cluster-based approach that can be easily incorporated into SpamAssassin, a popular spam filtering system that uses a combination of content-

and blacklist-based spam filtering techniques. We apply our approach to both honeypot email accounts and a 7-month email log that contains more than seven million emails. Our key findings include:

- As expected, most large BGP prefixes (*e.g.,* /8,/9) are too coarse-grained for building cluster-based blacklists. However, we observe 17.7% of mid-size BGP prefixes (*e.g.,* /15 - /20) are also too coarse-grained for spam filtering.
- DNS information can augment BGP prefixes. It can help break 26.3% of the BGP prefixes into smaller clusters, thereby reducing the false negative rate by 5-10%.
- We have built a system that combines BGP and DNS information to produce a cluster-based blacklist with a significant advantage over the existing IP-based blacklists (DNSBL) [9, 8, 7]. It can detect more than 50% of the spam not captured by the existing IP-based blacklists while maintaining comparable false positive rates.
- The combined cluster-based blacklist can be easily integrated into spam filtering systems such as SpamAssassin. When applied to honeypot email accounts, the integrated system can capture 30% - 50% spam emails missed by SpamAssassin.

Our work is the first to systematically examine the accuracy and potential of various IP-cluster based approaches for spam detection. Our results show that it is critical to obtain the correct boundaries for IP-clusters. In practice, it is desirable to combine different sources of information, *e.g.,* BGP, DNS to obtain fine-grained clusters with good coverage on new IP addresses. Performance-wise, our system currently uses only tens of millisecond for a single IP lookup without any optimization and 2.2GB database storage space to store the information for 2.7 million IP addresses.

The remainder of the paper is structured as follows. We first review related work in §2. We use two examples to show the complex relationship between BGP cluster and rANS clusters, followed by the implication on blacklisting in §3. We then present our data collection and experimental setup in §4. §5 and §6 elaborate on our detailed analysis on different clusters and how we combine them. In §7, we show how cluster-based reputation can be effectively applied in spam mitigation. Finally, §8 concludes the paper.

2 Related Work

Spam detection has been the subject of active research for years. Numerous techniques have been proposed. Some are content-based (*e.g.,* [19, 27, 20]), and some newly proposed ones are behavior-based (*e.g.,* [12, 21, 22, 14]). Many focus on detecting individual spam emails, as opposed to identifying spam-campaigns as a group (*e.g.,* [30, 31, 11]). Although spam-campaign detection can be highly effective for organizations with access to a large amount of spam emails, it is usually challenging for small organizations with a limited view. As a result, they usually resort to third-party provided blacklists such as Spamhaus [9], SpamCop [8], SORBS [7], and NJABL [13].

In this paper, we focus on improving blacklist-based spam filtering given its popularity and importance. Most blacklists today are based on individual IP addresses. In practice, many IP addresses are bi-modal in their spamming behavior [28]: they have either consistently high or low spam ratios over time. Thus, various blacklists [9, 8, 7] are created to block persistent spamming IP addresses. However, since a majority of spamming IP addresses appear only once and we continuously observe previously unseen IP addresses send only a few emails (either spam or legitimate emails), it is difficult to predict whether a new IP is good or bad. Consequently, IP-based blacklists are largely incomplete in terms of their IP address coverage [25].

To improve the coverage of spamming IPs, previous studies have proposed to replace individual IP addresses with clusters [28]. Clusters can capture the administrative/configuration boundaries of IP addresses, so that IPs within the same cluster are likely subject to similar security or network policies [18]. Typically IP clusters can be constructed using information from BGP [28], AS number [21], and dynamic IP ranges [29]. In fact, a recent study [14] claimed that AS number is the most important feature in their spam detector. This is followed by a more detailed study [26] on how to determine whether a BGP prefix is bad. By considering the reputation for a cluster of IP addresses instead of those of individual ones, we can significantly increase the spam-filtering coverage of unseen IP addresses. Although this sounds appealing, it relies on the assumption that IP clusters capture the correct boundaries between good and bad IP addresses. Given that the granularities of different IP clusters differ, these existing cluster-based spam filtering approaches often introduce a high false positive rate that prevents them from being adopted in practice. To reduce the false positive rate, our clustering techniques refine the AS number and BGP prefix based clusters into much more fine-grained ones that more accurately capture the administrative boundaries, hence making IP-cluster-based blacklists more practical.

More specifically, in our study, in addition to previously used BGP information, we also examine reverse DNS records as a way to construct IP clusters. This and other DNS information previously have not been fully explored for clustering IP addresses. The closest work uses rDNS information to identify dynamic or dial-up IP rDNS names (*e.g.,* regular expression) [3, 2] and blocks the IP addresses with such rDNS naming convention. They differ from our proposal in that they are using a set of manually crafted heuristics or rules to identify certain types of networks(*e.g.,* dial-up user networks) while our cluster using DNS information is much more general and can be fully automated.

Note that we do not use forward DNS mappings because it can be easily modified by spammers, *e.g.,* fast-flux networks typically employed by scam sites [17, 15], to evade detection. While the forward mapping between DNS names and IPs can change very frequently, the reverse mapping, which is set up by IP address owners, usually changes less frequently. It is difficult to create large-scale DNS fast-flux techniques on reverse DNS mapping.

3 Motivating Examples

Previous studies have investigated using BGP prefix as the network-aware cluster to group the spamming behavior [28]. However, the accuracy of the administrative boundary it captures depends on the granularity of the BGP prefix information. For instance, a large prefix can be further assigned into smaller prefixes that may not be externally observable in public routing data. As a result, what a prefix captures is often a coarse-grained administrative domain, and thus may not be detailed enough to block spams. DNS information, such as rANS and rDNS names, also reveals the administrative boundary of IP addresses. It can be more fine-grained than BGP information for some IPs but more coarse-grained for other IPs. Next, we show two motivating examples from real data to illustrate the complex relationship between these two clustering approaches.

First we study the example in Figure 1, with one prefix and four rANS names. They form a bipartite graph, where the upper-level nodes represent BGP prefix clusters and the lower-level ones denote rANS clusters. A line is drawn between a prefix and a rANS whenever there is an IP (1) belonging to the prefix cluster and (2) the rANS is responsible for resolving this IP address.

Figure 1 shows that prefix 69.61.0.0/17 has two sets of rANS names: `ns1-2.gunsprohibited.com` and `ns1-2.webserverdns.com`. Figure 2 illustrates the detailed IP range inside the prefix. The IP ranges with different set of rANS actually have distinct spamming behavior. The IP addresses under the rANS `ns1-2.gunsprohibited.com` send purely spam, while the IP addresses under the rANS `ns1-2.webserverdns.com` send only legitimate emails. The disjoint behavior of these two sets of addresses is likely due to different organizations these two groups of IPs belong to, as manifested by the rANS names. By assigning IP addresses into two corresponding rANS-based clusters, we are able to separate the good IP addresses from bad ones in terms of spamming behavior.

In the previous example, we show that rANS can help find smaller and more accurate clusters within a large prefix cluster. Now we show a contrasting example where prefixes are more fine-grained compared with rANS clusters. Consider a large ISP - comcast.net (Figure 3) with hundreds of BGP prefixes. All of the prefixes share the same set of rANS, namely `dns101-103.comcast.net`. Obviously the granularity of rANS clusters is too coarse given many IP prefixes within the same rANS. But the question is whether it is indeed necessary to decompose `comcast.net` into several hundreds of smaller BGP prefix clusters? Are they better in terms of finding the boundary between spamming and non-spamming behavior? In this case, the answer is yes. From our data, we found the spam ratio for rANS cluster is 0.76 for all three rANS clusters (since they always appear at the same time), which means that there are both legitimate emails and spam originating from the IPs under each rANS cluster. But if we study the BGP prefix clusters, their spam ratios are either close to 1.0 or well below 0.5. In fact, we found that the legitimate incoming/outgoing mail servers of `comcast.net` fall into two distinct BGP prefix clusters, and other prefixes are mostly dynamic IP ranges for DSL users. This information can be obtained by examining the Sender Policy Framework (SPF) [5] of `comcast.net`, which is encoded as a TXT record (a type of DNS record) and can be queried via normal DNS lookups. The response of `comcast.net` looks like the following:

```
  comcast.net.  300 IN TXT "v=spf1
ip4:76.96.28.0/23 ip4:76.96.27.0/24
ip4:76.96.30.0/24 ip4:76.96.59.0/24
ip4:76.96.60.0/23 ip4:76.96.62.0/24
ip4:76.96.68.100 ip4:76.96.68.101
ip4:76.96.68.102 ip4:76.96.68.103
?all"
```

In the SPF response, “ip4” indicates that the address range is IPv4. They correspond to the expected IP ranges for outgoing mail servers for a domain. “?all” indicates for all other IP addresses, their behavior of sending emails is unspecified. Correlating with the BGP prefix clusters, all the SPF IP ranges belong to BGP prefixes 76.96.24.0/21 and 76.96.48.0/20 as shown in Figure 3. These IP ranges differ from those DSL ones in that they are more likely to send legitimate emails given the SPF information. Further investigation shows that the aggregated spam ratios of these two prefix clusters are indeed

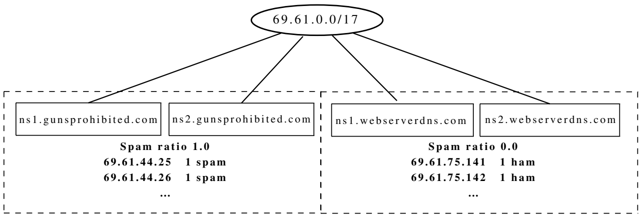

Figure 1. More detailed spamming behavior from DNS data.

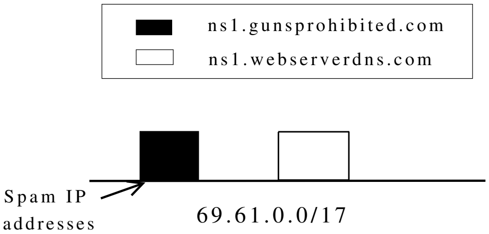

Figure 2. Smaller administrative boundaries in 69.61.0.0/17

very low while most other prefixes send purely spam.

Note we cannot simply whitelist IP addresses in the SPF records while blacklisting all other ones, because many ISPs or domains may either not provide SPF data or the SPF provided is too relaxed (*e.g.,* all the IP addresses are listed). Also spammers can spoof SPF records as well, *i.e.,* spammers who own the domains can set the SPF records such that all their IP addresses are listed.

In summary, we have illustrated examples where BGP prefix is either more accurate or less accurate compared to the DNS information for classifying spamming vs. non-spamming mail server IPs, motivating our work of combining these two sources of information.

4 Methodology

In this section, we discuss our data-collection methodology to investigate the properties of various network clusters based on BGP prefix and DNS information.

4.1 Data and experimental setup

Data source. The data is collected from the mail servers of University of Michigan EECS department, over the time period of 2008.12.7 - 2009.7.9 ranging over 155 days. It consists of about seven million emails, of which more than 5.5 million emails are spam emails (according to SpamAssassin) from 2,737,006 distinct IP addresses, 52,498 distinct BGP prefixes. Each log entry has four pieces of information: timestamp, sender IP, spam tag, and spam score output by SpamAssassin.

Spam filter - SpamAssassin. Our mail server runs SpamAssassin [1] as the spam filtering system. It employs several detectors which include Spamhaus [9] (IP-based blacklist) and a locally maintained IP-based blacklist. Every email is labeled as either spam or non-spam based on its score computed by SpamAssassin. The score is combined from the result of all detectors. If the score exceeds a fixed threshold (5.0 in our case), the corresponding email will be labeled as spam.

Although our mail server is a single vantage point, it does receive spam from a variety of IP address ranges. Figure 4 shows the CDF of IPs observed by the mail server. It roughly conforms to the range in previous studies, *e.g.,* Spamscatter [11] and the work by Ramachandran *et al.* [21].

Other data. To study the characteristics of clusters, we also leverage the dynamic-IP ranges produced by UDMap [29] to correlate with the clusters produced from the university data. This information of dynamic-IP ranges is of interest because dynamic IPs are more likely to send spam emails [29]. Further, we use the Hotmail history correlated with the IP addresses in the university data set from about the same time period to enhance the visibility of our dataset.

Experiment setup. At the end of each day, we extract the mail server log that contains the connecting IP and SMTP session for each email, and perform the following three DNS queries on the IP addresses we see for that day:

1) rDNS query on the IP to obtain its rDNS name (or hostname) and its reverse authoritative name servers (rANS).

2) Query on reverse domain name's MX record as well as the MX record of rANS domain.

3) Queries on three popular IP-based blacklists: Spamhaus, Spamcop, and Sorbs. The results are used for comparison with the cluster-based reputation.

We use an example to illustrate this process. Given an IP address, 141.211.22.134, we first perform the reverse iterative DNS query to get the rDNS `mx1.umich.edu` and the rANS `dns.itd.umich.edu` and `dns2.itd.umich.edu`. We subsequently extract the reverse domain name. The rDNS name `mx1.umich.edu` has the domain name `umich.edu`. The rANS `dns.itd.umich.edu` and `dns2.itd.umich.edu` has the domain name `itd.umich.edu`. If not already cached, we then perform MX record query on both domain names `umich.edu` and `itd.umich.edu` to get the MX records: `mx1.umich.`

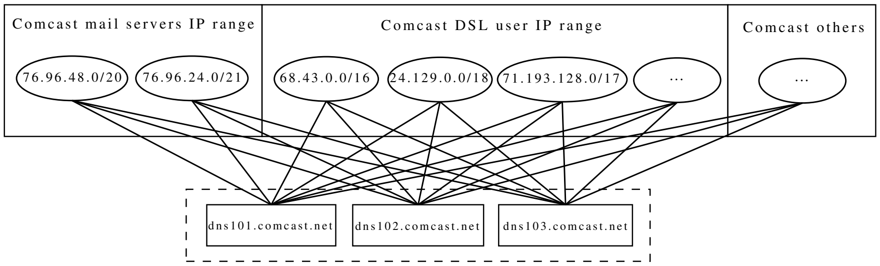

Figure 3. Prefix to DNS bipartite graph for `comcast.net` to illustrate DNS and BGP prefix relationships.

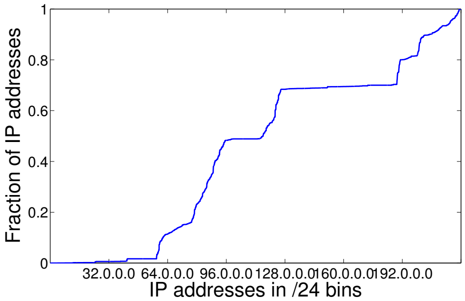

Figure 4. CDF of spamming IP range observed

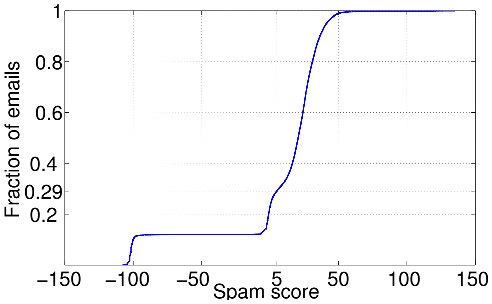

Figure 5. CDF of spam scores from SpamAssassin

edu, `mx2.umich.edu` and `mx3.umich.edu`. We finally record the results of querying the IP address 141.211.22.134 for the three blacklists.

4.2 Appear-once IP addresses and new IP trend

In our data set, we observe that more than 80% of IPs were active only for one day. Most of these IPs sent only one to five spam emails, however, collectively they contributed to about 40% of all the spam emails and 2.4% of all the ham (non-spam) emails received. Further analysis shows that about 34.6% of these appear-once IPs are dynamic IPs. In contrast, only 15.3% of the remaining IPs are dynamic, suggesting that appear-once IP addresses are more likely to be dynamic IPs, thus are less likely legitimate mail servers. Nevertheless, we cannot simply block them as they might send valid emails as well.

These appear-once IPs span across the entire training period. To quantify their impact, we compute the number of new IP addresses observed on a monthly basis. The result indicates this number keeps increasing each month starting at 189,994, reaching 603,284 in the last month. We expect it to continue growing given our small vantage point until we observe most IPs. Thus, maintaining a reputation history for individual IP addresses will not work well, as there will always be new instances of unseen IPs appearing every day. This result confirms previous results that about 85% of IP addresses send less than 10 emails [21].

4.3 Data cleaning

Since we use the spam classification produced by an imperfect detector – SpamAssassin, we may not have the ground truth. Figure 5 shows a mostly bimodal spam score distribution assigned by SpamAssassin where most of the emails either have a high score (much higher than 5.0) or a low score (much less than 0). However, we do see a fraction of emails (less than 8%) that fall into the score range of $5(\pm 3)$, which may contribute to false positives and false negatives using the default threshold of 5.

To investigate the false negative behavior, we set up a honeypot account within our department (advertised since 2007 on personal Web pages but not used), through which we found that SpamAssassin has a non-negligible false negative rate—155 (16%) missed spams out of 965 recent spam emails. The false positive rate of SpamAssassin, however, is very low as reported by previous work [25].

It is important to reduce the false negatives of SpamAssassin, as those IP addresses sending spam may appear legitimate and thus cause inaccurate evaluation of clustering. One source of information we use is IP-based DNSBLs. Previous studies [16, 23, 21] show that significantly more spam are detected by combining multiple blacklists. Further, these studies show that a large fraction of spammer IP addresses will eventually be listed after some time (*e.g.,* 2 months or so) in blacklists if not at the time spams from such addresses are received. As a result, we decided to conduct the data cleaning process by querying 5 popular blacklists (Spamhaus, Spamcop, SORBS, NJABL, CBL) approximately 2 months after we received the emails. Any IP address listed in at least two of the blacklists will be considered bad and we treat all the emails they sent as spam.

Another source of information to facilitate our analysis is obtained from Hotmail servers. IP addresses that send purely spam to Hotmail server (reported by the Hotmail spam filter) will be considered as spamming addresses as well. For IPs that send any legitimate email to Hotmail server, we conservatively do not consider them as bad IPs to prevent false positives.

To summarize, we consider a given IP address as bad or spamming IP if it satisfies any of the following three conditions.

1. It has a high spam ratio $> 90\%$ (reported by SpamAssassin)
2. It is captured by at least two blacklists that are queried after two months.
3. It never sends legitimate email to Hotmail and all the emails sent by the IP are classified as spam.

We are able to reduce the false negative rate down to 4.5% in the honeypot account after applying these rules while introducing 0 false positives in 3 of our personal accounts. After the data cleaning process, we can characterize the spamming behavior more accurately.

Note that our approach is conservative to obtain fairly accurate false negative evaluation by ensuring that bad IPs identified for evaluation are not misclassified. Nevertheless, it is not possible to completely ensure that all bad IPs are caught in this manner, and those IPs may appear in the evaluation result as good IPs while classified as bad IPs due to the its cluster's reputation, thus still causing slightly inflated false positives in our evaluation.

4.4 Different clusters analyzed

In this study, we measure the characteristics of each cluster type based on BGP prefix and DNS information and then propose ways to integrate them to obtain increased benefit while minimizing the shortcomings of individual approaches. Specifically, the clusters we build include BGP prefix clusters, DNS clusters (rANS information combined with rDNS name information). Since BGP prefix is a well-known technique, we use it as the baseline for comparison. In the next section, we explore the relationship between the BGP prefix and DNS clusters to investigate how to utilize the DNS information in combination with the BGP prefix information to cluster IP addresses for accurate spam detection.

5 Cluster-based Spam Filtering

In this section, we study the effectiveness of cluster-based blacklists. We first consider BGP prefix clusters since they are one of the most common ways of clustering IP addresses. However, as stated before, the accuracy of BGP prefix cluster depends on the visibility of the prefix structure in public routing data. We also study DNS-based clusters as DNS information may be used to track the corresponding host administrative boundaries. We found DNS-based clusters sometimes yield better prediction accuracy than BGP prefixes, especially in the cases where they provide more fine-grained boundaries. Given that these two types of information can complement each other, we consider combining them to derive "combined clusters," which are found to improve spam detection by more accurately predicting host spamming behavior.

5.1 BGP prefix clusters

From Route Views [24], a public BGP data source, we sampled routing table snapshots of 7 distinct days from March 7th to March 13th 2009. We use the longest matching prefix of the format "1.2.3.4/16" to represent IP's cluster. For those 5000 IP addresses not associated with any prefix, we resort to whois database [10] to find their associated prefixes.

5.2 DNS clusters

To utilize DNS information, we consider two ways for cluster construction.

rANS clusters: The *Reverse Authoritative Name Server Cluster* (rANS cluster) groups hosts by their authoritative name servers. For each incoming IP, we perform reverse DNS lookup iteratively to identify its authoritative name servers. Note that a recursive lookup is needed to collect several levels of name servers, each returned by a different level of authority such as `1.in-addr.arpa.`, `2.1.in-addr.arpa.` and so on. For instance, take Figure 3 as an example, `dns101-103.comcast.net` is the last level rANS for IPs in all BGP prefixes. Note that all the three rANS are associated with all other BGP prefixes, meaning that their granularity is the same. However, there can be some rANS that are associated with more BGP prefixes (*e.g.,* our university uses rANS from another university as backup). Even for IPs without a reverse hostname, *i.e.,* NXDOMAIN response, we can still obtain additional information of other servers in the DNS hierarchy. Note we do not use the host name reported in the HELO message of an SMTP connect because the hostname can be easily spoofed. An IP address may be resolved by multiple name servers (*e.g.,* for load balancing). We only pick the name servers in the lowest level which represent the most fine-grained administrative domain. If there are multiple ones at the lowest level (as is the case in Figure 3), we pick the one that is associated with the minimum number of BGP prefixes indicative of the most fine-grained administrative domain.

Naming cluster: Within an administrative boundary, hosts play different roles. Usually only a subset of IPs are used to set up mail servers. It is necessary to separate these mail-servers from the remaining ones. The reverse DNS-names of IP addresses provide hints on how to classify them. Within each administrative boundary identified by a BGP prefix and/or rANS, we identify four common naming patterns:

1. All rDNS names are in the same domain and share a similar naming pattern. Table 1 shows one such example, where all IPs are dynamic IPs with an aggregated spam ratio of 99.6%.
2. All rDNS names are within the same domain, but with non-uniform naming patterns. For instance, an enterprise can have several legitimate mail servers as well as other non-server hosts. To discover the former, we resort to other sources of information - MX records, SPF sender IP ranges and naming convention as specified in RFC [6] that recommends the DNS names of mail servers to begin with keyword 'smtp', 'mail', or 'mx'. The example in Table 2 includes a legitimate mail server listed in the first row, confirmed by MX records of the domain tvtel.pt. For clustering purposes, these servers are separated into their own naming clusters.
3. IPs without any rDNS names are mixed together with IPs with rDNS names. They are separated into different naming clusters as shown in Table 3.
4. Many domains exist, with each including only a few IPs. Table 4 shows one such example. Based on our observation, such cases are usually correlated with spamming behavior. With many domains registered within the same administrative domain, it is highly likely that they are owned by spammers who set up corresponding MX records or SPF to make them appear as legitimate mail servers.

Table 1. Cluster's naming pattern 1 - consistent naming.

IP address	rDNS name	Spam Count	Ham Count
190.82.167.51	190-82-167-51.adsl.tie.cl	1	0
190.82.165.55	190-82-165-55.adsl.tie.cl	1	0
190.82.164.20	190-82-164-20.adsl.tie.cl	1	0
190.82.151.205	190-82-151-205.adsl.tie.cl	1	0
190.82.151.169	190-82-151-169.adsl.tie.cl	1	0
190.82.151.158	190-82-151-158.adsl.tie.cl	1	0
...	...	...	...

Table 2. Cluster's naming pattern 2 - mixed w/ legitimate mail servers.

IP address	rDNS name	Spam Count	Ham Count
88.157.32.73	webmail.tvtel.pt	0	1
88.157.237.48	rev-88-157-237-48.tvtel.pt	1	0
88.157.113.191	rev-88-157-113-191.tvtel.pt	1	0
88.157.204.61	rev-88-157-204-61.tvtel.pt	1	0
88.157.218.127	rev-88-157-218-127.tvtel.pt	1	0
88.157.71.28	rev-88-157-71-28.tvtel.pt	2	0
88.157.85.30	rev-88-157-85-30.tvtel.pt	2	0
...	...	...	...

Here we normally use the last two tokens as domain name (*e.g.,* umich.edu from www.umich.edu). However, if the last code is country code, then we use the last three tokens as domain name (*e.g.,* www.sjtu.edu.cn from sjtu.edu.cn). Exception for this is that when there are only three or even two tokens altogether for a country-code-ending rDNS name, we will still use the last two tokens as domain name (*e.g.,* yahoo.cn from www.yahoo.cn). To fully utilize DNS information, we construct ***DNS clusters*** by converting naming-based clusters to rANS clusters. For each rANS cluster, we attempt to match any of the four common naming patterns to further split it into smaller clusters. This enables us to discern good IPs that may share the same administrative domain with spamming IPs.

5.3 Cluster granularity

Ideally, we want to identify clusters consisting of mostly good or bad IPs. Obviously the granularity of the clusters plays an important role here. The extreme case, where each cluster consists of a single IP, falls back to the per-IP based scheme and is no longer useful for predicting spamming behavior of unseen IPs.

First we cluster the 2,737,006 distinct IP addresses described previously into 92,449 BGP prefix clusters and 60,659 rANS clusters, respectively. Thus on average, BGP prefix clusters are more fine-grained than rANS clusters. However, it is the distribution instead of the average that matters. Figure 6 shows the distribution

Table 3. Cluster's naming pattern 3 - (Some) IPs without rDNS names.

IP address	rDNS name	Spam count	Ham count
208.53.152.220	mta220.pmxa-net.net	1	0
208.53.152.221	mta221.pmxa-net.net	0	2
208.53.152.219	mta219.pmxa-net.net	0	2
208.53.185.230	N/A	1	0
208.53.185.234	N/A	6	0
208.53.185.228	N/A	8	0
208.53.147.84	N/A	1	1
...	...	...	...

Table 4. Cluster's naming pattern 4 - many domains.

IP address	rDNS name	Spam count	Ham count
89.30.144.62	familyhunterburns.net	1	0
89.30.145.171	myfeedstore.net	1	0
89.30.145.170	myephoto.net	2	0
89.30.144.110	advantageatv.net	2	0
89.30.145.175	mywirelesscentral.net	2	0
89.30.145.173	mymusicchannel.net	1	0
89.30.145.178	newdatasystems.net	2	0
89.30.145.179	nirvanashopping.net	1	0

of the number of BGP prefixes an rANS cluster spans, and vice versa. About 10% of the BGP prefix clusters can be further divided into smaller clusters by considering rANS information. When applying naming clusters to rANS clusters, we obtain 106,356 DNS clusters which is slightly more than BGP prefix clusters.

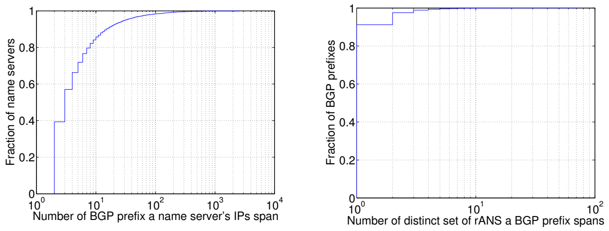

Figure 6. The granularity relationship between BGP prefix clusters and rANS clusters.

5.4 Spamming behavior

We examine the distribution of spam ratio for each cluster type over the 7-month duration as shown in Figure 7. The figure shows how diverse the spamming behavior is within the same cluster (How likely that some IPs send spam while others send legitimate emails within the same cluster) and how they differ between different types of clusters. We can see that in the figure, the spam ratio distribution of BGP prefix clusters is closer to that of IP than of DNS clusters. BGP prefix and IP-based clusters both have about 0.4% of cases with 0 spam ratio. The curve shapes for BGP prefix clusters and DNS clusters are similar, but their initial values differ by more than 9%. In particular, more than 17% DNS clusters have spam ratio of 0. This suggests that BGP prefix clusters can effectively identify clusters with spamming IP addresses while DNS is useful for uncovering legitimate servers. This is explained by the observation shown later in §5.6 that BGP prefix cluster more closely reflects dynamic IP ranges where a significant amount of spam comes from [29].

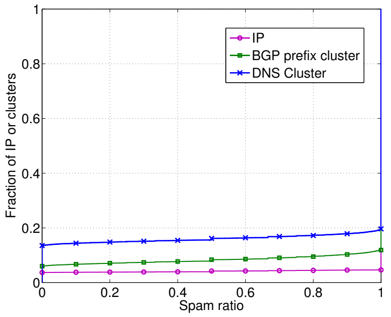

Figure 7. CDF of spam ratio for different clusters.

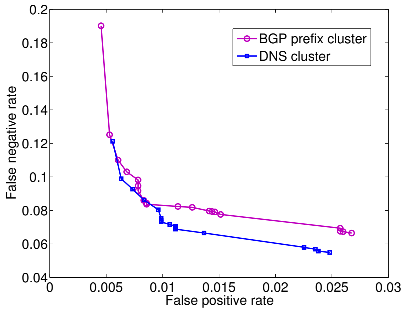

Figure 8. Clustering false positive and false negative rate.

Next, we study the effectiveness of using cluster's spamming behavior to classify good and bad IPs. The idea is that we define a threshold of spam ratio for deciding whether a particular cluster is good or bad. If the spam ratio exceeds the threshold, we consider any IP address within the cluster as bad IP. We pick one day as testing and the remaining earlier days of data as training data, changing the testing data over 30 days from Jun 10th to Jul 9th, 2009. We plot the graph by adjusting the threshold from 0.8 to 1.0 at the granularity of

0.01, averaging across different testing days in Figure 8. We can observe that DNS clusters have higher accuracies compared with BGP prefix clusters since they can further divide an administrative domain into potentially good IPs and bad ones based on the naming pattern.

5.5 Cluster persistence

Behavior of clusters clearly is more persistent than that of individual IPs due to longer history. Most IPs appear only for a day or two, but 80% of the clusters appear in at least 8 days out of the 7-month duration of our study. The cluster behavior also shows consistency as the average standard deviation of spam ratio across all clusters over the 7-month duration is as low as 0.09 (consistent with the result by Venkataraman *et al.* [28]). This suggests that whenever a cluster is identified as good or bad, it remains so for a relatively long time period. Cluster-based analysis is therefore more effective for spam filtering compared to purely IP-based approaches. In Figure 9, we show an example of how different IPs appear across time in one of the largest prefix, which belongs to a large ISP in India. Interestingly, in this case, all IPs are sending purely spam from this prefix, showing persistent spamming behavior within a cluster.

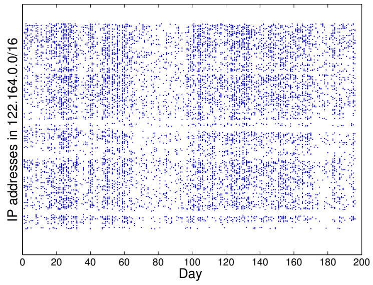

Figure 9. Different IP addresses appeared across different days for one of the largest prefix

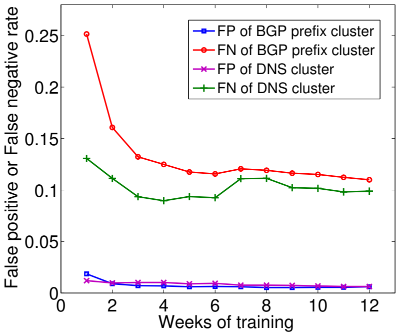

Figure 10. Training time vs. FP and FN

Table 5. Correlation of dynamic IPs with spam

	Dynamic IP	Static IP
No. of Spam	1,199,468 (22.1%)	4,235,613 (77.9%)
No. of legitimate email	12,258 (0.5%)	2,327,932 (99.5%)

In Figure 10, with the spam ratio threshold set to 0.98, we can see that the false positive and false negative rate of both BGP prefix clusters and DNS clusters decrease with increased training time. The result further suggests that with sufficient training time, we can classify new IPs based on their cluster history with high accuracy. Longer training time leads to more stable results which in turn supports persistent spamming behavior of clusters. Note that we do not observe clusters completely disappear during our study, *i.e.,* all IP addresses within a cluster stop sending emails at some point, despite a possible conjecture that spamming botnets may create dynamic time-based behavior of the clusters (*e.g.,* rDNS names changed and thus old clusters based on DNS information may become obsolete). We plan to investigate this further, as our clustering approach opens new potential opportunities for detecting botnets.

5.6 Correlation with dynamic IPs

Given that past studies have shown that spam often originates from dynamic IPs, we correlate the cluster data with dynamic IP ranges based on UDMap [29] which is the latest known, accurate source for such data. The dynamic IP ranges are gathered by analyzing the Hotmail user login traces [29]. We found that out of the 2,737,006 IP addresses, 786,460 are identified as dynamic with each IP assigned to a unique dynamic IP range ID. Surprisingly, we observe that dynamic IPs only contribute to 22.1% of the total spam as shown in Table 5. This deviates from the finding in UDMap [29] that shows dynamic IPs contribute to 42.2% of the total spam. Two possible reasons can explain this difference: (1) Our vantage point captures different spamming behavior from what was seen in Hotmail. (2) Spammers are shifting to using static IP addresses for sending spam to improve spam delivery, possibly because of the more prevalent blocking of direct connection to port 25 from dynamic IPs and the inclusion of dynamic IP ranges by many popular blacklists [9, 7].

We exclude prefixes containing at least one static IP address based on UDMap classification results, thus obtaining 4325 **BGP prefixes** that contain purely dynamic IPs. In contrast, there are only 254 **rANS clusters** containing purely dynamic IP addresses. Further, by correlating them with the dynamic IP range ID produced by UDMap (each dynamic IP range is assigned a unique ID), we found that these BGP prefixes covers 9785 dynamic IP range ID (7% of all the ranges) while rANS only covers 406 dynamic IP ranges (0.5%). Further analysis reveals that out of these 4317 BGP prefixes, 64% of the BGP prefixes match exactly with one dynamic IP range. 34% of them strictly contains more than one dynamic IP range. The remaining 2% BGP prefixes are either a strict subset of one dynamic IP range or overlap with more than one dynamic IP range (Those dynamic IP ranges are also covered by other BGP prefixes).

This indicates that BGP prefix clusters correlate better with dynamic IP addresses than rANS clusters, and it can be explained by the coarse-grained properties of most rANS clusters. For instance, in the previous example illustrated in Figure 3, some of the BGP prefixes belong to DSL IP ranges, while many others are Comcast legitimate mail servers. In this case, all Comcast IPs fall into the same set of rANS clusters but belong to distinct BGP prefix clusters.

5.7 Choice of cluster type

To study which cluster performs the best and when, we need to define the metric for good clusters. The natural metric would be whether IPs within a cluster share the same spamming behavior (sending mostly spam or mostly legitimate emails). The cluster granularity clearly plays an important role in determining its accuracy. We study each type of clusters separately.

The accuracy of **BGP prefix clusters**, as previously discussed, depends on the visibility of BGP routing table. Sometimes, a larger prefix may be reallocated into smaller ranges not externally observable. As a result, we conjecture that larger BGP prefixes may not accurately discern spamming behavior.

From Figure 8, by setting the threshold of spam ratio to 0.95, we are able to obtain a relatively low false positive rate. We then study the false negative distribution by varying BGP prefix sizes. Figure 13 illustrates the average false negative count per cluster for each BGP prefix size, indicating that /8 and /9 BGP prefix have the worst false negative performance, and smaller prefixes such as /20 - /24 have quite low false negative count per cluster. This result validates our conjecture. We can conclude that fine-grained BGP clusters are needed to accurately capture spamming behavior.

rANS clusters, as discussed in §5.3, are in general more coarse-grained than BGP prefixes. In fact, based on our reverse DNS query results, we found that the largest rANS clusters observed are those close to root name servers, *e.g.,* `tinnie.arin.net`, `ns.lacnic.net`, `ns-sec.ripe.net`, and `ns-pri.ripe.net`. Some countries such as Korea also contribute to this. We found that most of the IPs in Korea do not have a hostname and they all share the same set of reverse name servers such as *e.g.,* `a.dns.kr` and `b.dns.kr`. Although poorly maintained DNS information usually implies spamming behavior, *i.e.,* large rANS clusters all have spam ratios greater than 90%, they may still include legitimate mail servers. Also, poorly maintained DNS information occur in different networks. We found that IP addresses under these rANS do not cluster well because they may belong to different BGP prefixes, thus crossing different administrative boundaries. Especially for rANS clusters close to root name servers, sometimes they are associated with up to several thousand BGP prefixes.

The use of **Naming cluster** is only applicable for a known administrative boundary since the observation of naming patterns are drawn from within an administrative boundary. Although 1/3 of the IP addresses do not have rDNS name, they are mixed with IP addresses that have rDNS name and can be considered as as a type of pattern as mentioned in Table 3. As a result, we exploit naming pattern that can be a good indicator to split an existing cluster into finer-grained ones that may exhibit different spamming behavior.

In conclusion, neither BGP prefix cluster nor rANS cluster is perfect. However, they complement each other in terms of capturing the administrative boundaries. This leads to the idea of combining them along with naming clusters to more accurately separate good IP addresses from bad ones.

6 Combined clusters

In previous examples, we show that more fine-grained clustering can usually lead to more accurate identification of administrative boundaries and effectively separate good IP addresses from bad ones in terms of spamming behavior. In this section, we discuss in more detail how we can combine different types of cluster information. The idea is to first combine BGP prefix clusters with rANS clusters to identify more accurate administrative boundaries. We subsequently apply naming based clustering within each administrative boundary to perform further separation. We show that combined clusters are indeed qualitatively better than applying isolated clustering method individually.

Overall, the clustering process has two phases as shown in Figure 11. The first training phase generates a bipartite graph based on the BGP prefix and rANS information for each IP address (as described in §3). This is followed by the cluster assigning phase which takes the bipartite graph and assigns new IP addresses to intermediate clusters according to a clustering assignment algorithm. Intermediate clusters combine BGP prefix and rANS information. The naming clustering process

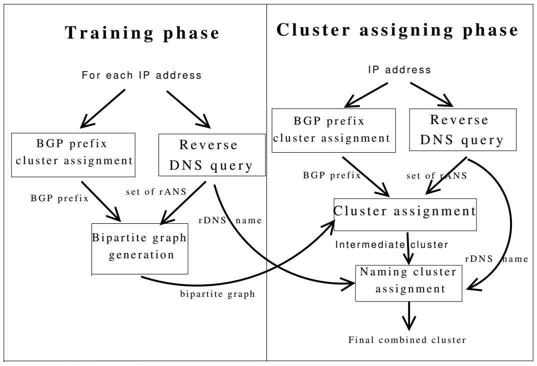

Figure 11. Combined cluster construction using prefix and DNS information.

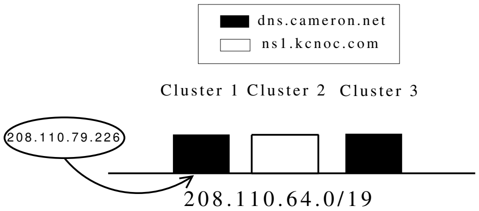

Figure 12. Assigning an IP address into its corresponding administrative boundary in prefix 208.110.64.0/19.

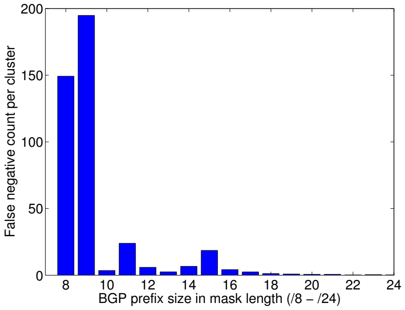

Figure 13. False negative count per cluster sorted by BGP size (in mask length)

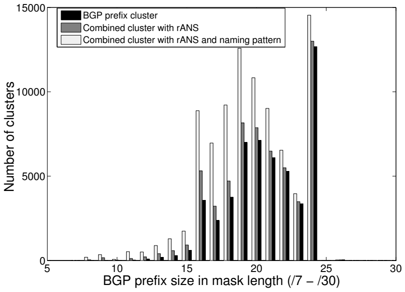

Figure 14. Number of clusters for different cluster types sorted by BGP prefix size (in mask length).

then performs naming pattern matching within each intermediate cluster to obtain the final combined cluster. The intuition here is that naming pattern works better when we already capture a good administrative boundary which can be obtained by combining BGP prefix and rANS information.

6.1 Combined cluster assignment

Since BGP prefix is generally finer-grained, we start with a BGP prefix cluster and refine it into smaller clusters by considering rANS information. The cluster assignment algorithm takes an IP address as an input, looks up the corresponding BGP-prefix cluster using the longest prefix matching and finds one or more corresponding rANS clusters.

Recall from Figures 3 and 1, we construct a bipartite graph based on BGP prefix clusters and rANS clusters by drawing an edge between a BGP prefix cluster and an rANS cluster whenever at least one IP address within the BGP prefix cluster also belongs to the rANS cluster. The degree of the rANS cluster in the bipartite graph represents its granularity: an rANS cluster with a smaller degree indicates a more fine-grained administrative boundary due to less sharing across prefixes. For instance, some rANS might be third party rANS shared by dramatically different administrative domains. Using such rANS will inaccurately treat distinct administrative domains as a single domain.

Since we know that coarse-grained administrative boundaries produced by rANS clusters may not work well for classifying spamming behavior, we find the *minimum-degree rANS cluster* for a given IP address to increase cluster granularity. After identifying such a minimum-degree rANS cluster, we form a fine-grained cluster by including other IPs sharing the same rANS cluster within the BGP prefix cluster. The resulting cluster can be illustrated in Figure 12. In this example, we first find the minimum-degree rANS for IP address 208.110.79.226 (`dns.cameron.net` in this case) and then assign it into cluster 1 where all IP addresses (including 208.110.79.226) within cluster 1 has the same minimum-degree rANS and they also belong to the same BGP prefix 208.110.64.0/19.

Note that from the above cluster assignment, BGP prefix clusters may be divided into smaller clusters, or they may be already sufficiently fine-grained without requiring further splitting. After we combine BGP prefixes with rANS clusters, we effectively find a more accurate administrative boundary in which the naming patterns can be applied to obtain the final combined cluster. Table 6 shows the result of assigning all 2,737,006 IP addresses into these clusters. These IP addresses fall into

Table 6. Distribution of 2,737,006 IP addresses on different types of cluster assignment.

Type of cluster assignment	Number of IP addresses	Number of clusters
BGP prefix cluster	1,160,491	68,161
Combined cluster split from BGP prefix cluster	1,576,515	101,050

68,161 BGP prefix clusters and 101,050 combined clusters. It shows that a significant portion of the total IP addresses (42.2%) falls into combined clusters split from BGP prefix cluster. Previously, §5.3 shows about 10% BGP prefix cluster can potentially be split into smaller clusters considering rANS information. But here we found that about 26.3% of the original BGP prefixes can be further split into smaller clusters by considering rANS and naming pattern. Further, we found that larger BGP prefixes such as /8 and /9 will almost always be split into smaller clusters. 19.6% of even smaller BGP prefixes such as /15 - /20 can also be further split.

6.2 Cluster granularity

In Figure 14 we plot the number of BGP prefix clusters sorted by size in mask length and compare with the result of further breaking them into more fine-grained clusters. We found that the number of the clusters that range from /24 to /16 increase significantly. However, for BGP prefixes with size of /8 - /15, although the absolute increase in the number of clusters is small, the ratio of increase is significantly larger. This confirms our previous observation that larger BGP prefixes are too coarse-grained and do not represent accurate administrative boundaries. Overall, most BGP prefix clusters, regardless of their sizes, can be split into smaller clusters. We discuss the implication on inferred spamming behavior next.

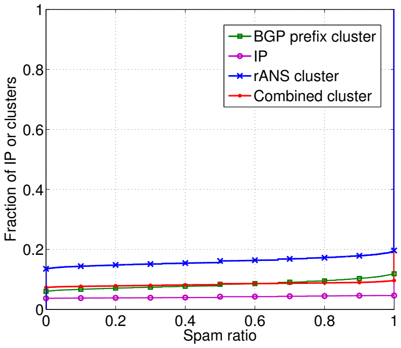

Figure 15. CDF of spam ratio for three different clusters.

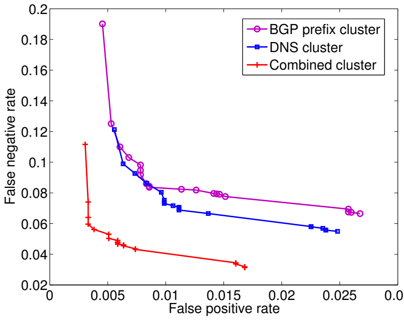

Figure 16. False positive and false negative rate for spam ratio threshold from 0.8 - 0.1 with granularity of 0.01

6.3 Spamming behavior

As shown in Figure 15, combined clusters have similar behavior to that of BGP prefix clusters in terms of identifying clusters containing either mostly good IPs or mostly bad IPs, suggesting that the accuracy property is preserved from BGP prefixes. Furthermore, in Figure 16, it clearly shows that combined clusters have the best false negative and false positive result at all spam ratio threshold from 0.8 to 1.0 with granularity of 0.01. Using combined clusters, we can reduce the false negative rate by about 6% - 10% compared to using BGP-prefix clusters, without increasing the false positive rate. The result is expected because finer-grained clusters better capture the boundaries between good and bad IPs as discussed before.

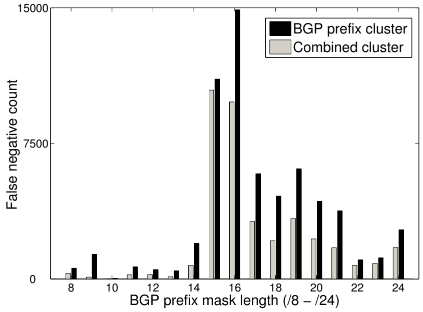

Figure 17. False negative count comparison for different clusters sorted by BGP prefix size.

We now analyze the false negative breakdown by

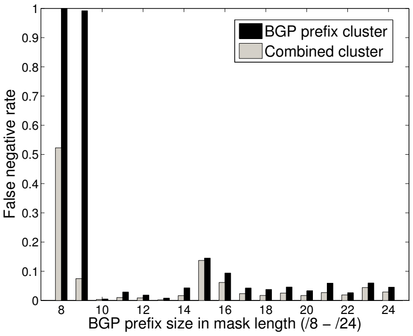

Figure 18. False negative rate comparison for different clusters sorted by BGP prefix size.

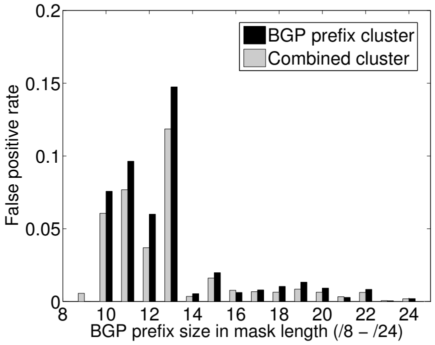

Figure 19. False positive rate comparison for different clusters sorted by BGP prefix size.

BGP prefix size. Figure 17 shows that most false negatives are distributed in /15 - /20 where there is the most increase in terms of the number of clusters. Combined clusters consistently have fewer false negatives compared to BGP prefix clusters for each prefix size. Examining false negative rate across prefixes of different sizes, we show in Figure 18 that larger BGP prefixes clearly have higher false negative rate. As discussed before, a large BGP prefix can be further divided into smaller ones for different organizations not externally visible. As a result, we show that by combining DNS information, we are able to significantly reduce the false negative rate for such large clusters.

However, as we can see, for /8 BGP prefixes, the false negative rate is still around 50% using combined clusters. A closer look reveals that it is caused by a large /8 BGP prefix belonging to MIT, which originated a non-negligible fraction of spam. These addresses also contribute to many legitimate emails. In this case, this cluster is considered to be a "good" cluster with all its spam treated as false negatives. In fact, most of the spam is contributed by a few IP addresses which appear to be legitimate mail servers. We suspect that these spam is due to mail forwarding, and plan to confirm it. Note that IP-based blacklists will not block these legitimate server IPs either.

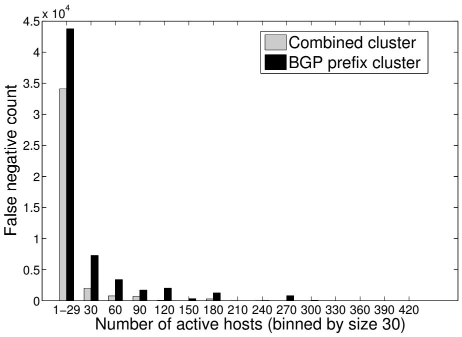

Figure 20. False negative count comparison for different clusters sorted by the number of active hosts within each cluster.

We find that combined clusters can also reduce false positive rates. As we can see in Figure 19, the false positive rate at each BGP prefix size is reduced except for /9 prefixes. The reason is that originally all /9 BGP prefixes are considered to be good clusters (close to 100% false negative rate). But in fact, by splitting /9 BGP prefixes into smaller combined clusters, we can separate good IP addresses from bad ones that are originally mixed together in /9 BGP prefix clusters. It can greatly reduce the false negative rate, but with a slight increase in false positive rate.

We further examine the false negative breakdown by the number of active hosts within each cluster in Figure 20. The X-axis shows the size of each cluster binned by 30. For example, the first bar shows the false negative count for clusters with host population ranging from 1 to 29. First of all, we observe that most of the false negatives are contributed by small clusters due to the lack of sufficient history (*i.e.,* sample size is too small). Even a very small number of misclassified spam emails by spam filters would significantly bias the spam ratio for the entire cluster. With more history, the spam ratio of the clusters becomes more stable, as evidenced by fewer false negatives for clusters of larger sizes in the same figure. Consistent with earlier observations, combined clusters can further reduce false negatives incurred by BGP prefix clusters. However, the reduction is limited for clusters of smaller host populations. This is also due to the lack of sufficient history for BGP-prefix clusters, making it more difficult to further split prefixes into smaller combined clusters.

6.4 Detection coverage

Previously, we have discussed using clusters to assign reputation for unseen IPs. Since new IP addresses

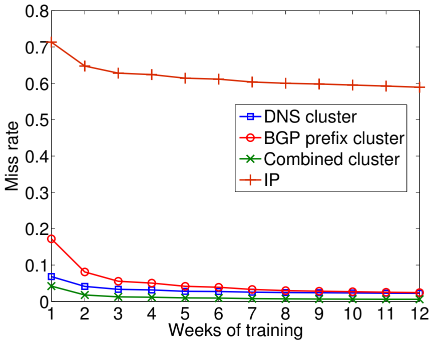

Figure 21. Training time vs. miss rate

appear daily, it is important to quantify the improved detection coverage of IPs by using clusters compared with IP-based approaches. Similar to Figure 10, we fixed one day for testing and varied the training length.

For Figure 21, we define the metric *miss rate* to be the number of emails whose sender IPs do not fall into any existing cluster divided by the total number of emails. For IP-based reputation, it is defined as the number of emails from unseen IPs divided by the total number of emails. We exclude IPs from our university network to avoid any bias caused by the data collection location.

Obviously, the miss rate decreases with more training data. In particular, the miss rate for individual IPs is as high as 60% even with 12 weeks of training. However, clusters help reduce the miss rate to well below 20%, especially for combined clusters with a miss rate of only about 0.6% with 12 weeks of training – two orders of magnitude difference compared with the miss rate of IP-based reputation. Combined clusters also have a smaller miss rate than both BGP prefix clusters and DNS clusters. This is explained by falling back to DNS and BGP prefix cluster to obtain history information whenever a new IP falls into a combined cluster with a lack of history. Further analysis reveals that combined clusters actually can help assign reputation for more than 93% of the unseen IP addresses.

7 Spam detection using cluster-based reputation

As discussed before, building IP-based blacklists can be very challenging. In previous section, we have shown how to build a fine-grained cluster that can outperform existing clusters. In this section, we evaluate the classification result of our clusters in terms of the false positive and false negative rate against existing popular IP-based blacklists. We also integrate the cluster reputation with SpamAssassin to examine the amount of additional spam detected and evaluate the number of new false positives introduced. Note that our approach can work well even with only our local vantage point, which makes it easier to deploy. The performance in terms of the lookup time and storage space of our system is quite low. The average lookup time per IP is about 60ms which is sufficiently low to be practical. The storage space required for storing information about the 2.7 million IP addresses (including the DNS information) is about 2.2GB on disk which can easily run on any modern commodity hardware.

7.1 Comparison between cluster-based and IP-based blacklists (DNSBL)

As previously described, we build the cluster-based blacklist purely based on the aggregated spam ratio of clusters. As demonstrated in Figure 8 with varying threshold values, there exist trade-offs between false positives and false negatives.

Note that we did not utilize any local IP-based reputation history to help further reduce false positives, thus it is likely that our detector will have a higher false positive rate. However, in order to compare against the IP-based blacklist, we do consider the local history information to reduce the false positive at the cost of some increase in false negative rate. If an IP address falls into a brand new cluster for which we have no history, we resort to content-based spam filters to decide whether it is a spam. Once we have seen enough history for the cluster, we can make use of cluster reputation for future spam detection.

We empirically set the spam ratio threshold to 0.97, 0.98 and 0.99 respectively to compare with each DNSBL averaged over 30 different days randomly selected from June to July 2009. The training data begins from the first day of our data collection to the day before the testing day. The DNSBL we choose are Spamhaus [9], Spamcop [8] and SORBS [7]. Table 8 illustrates the result of using DNSBL alone compared with using only our cluster, and using the combined approach. It shows that our standalone cluster-based detector already outperforms each individual DNSBL except SORBS which has a slightly better false positive rate but a much worse false negative rate.

With BGP cluster alone, to maintain the same level of false positive rate, the false negative rate will increase to more than 20% as shown in Figure 16. Further, by incorporating cluster-based detection, we can detect more than half of the spam missed by these blacklists while maintaining comparable false positive rate. In fact, after combining our cluster-based detector with Spamcop, it produces a detector that has significant improvement (both false positive and false negative) over Spamhaus alone. To understand the false negative improvement, we investigate scenarios where a blacklist misses bad IPs that were caught by our cluster-based detector. One rea-

Table 7. Results of integrating cluster-based reputation with SpamAssassin

Spam Ratio threshold	Score assigned	Honeypot account			Personal account 1			Personal account 2			Personal account 3		
		Spam	Ham	FN	Spam	Ham	FN	Spam	Ham	FN	Spam	Ham	FN
		1025	0	144	3750	14143	521	1340	12231	89	11	1550	1
		FNR	Matched	FPI	FNR	Matched	FPI	FNR	Matched	FPI	FNR	Matched	FPI
0.7	1	46	96	0	185	356	4	40	67	5	0	0	0
0.7	2	63	96	0	194	356	4	42	67	6	0	0	0
0.7	3	79	96	0	199	356	5	43	67	8	0	0	0
0.8	1	45	90	0	175	335	3	37	63	4	0	0	0
0.8	2	62	90	0	179	335	3	39	63	6	0	0	0
0.8	3	75	90	0	183	335	3	42	63	6	0	0	0
0.9	1	44	88	0	171	317	0	34	59	1	0	0	0
0.9	2	61	88	0	175	317	1	38	59	1	0	0	0
0.9	3	74	88	0	180	317	1	40	59	1	0	0	0

Table 8. Comparison with existing IP-based blacklists (DNSBL).

Blacklist name	FN	FP	Threshold	Our FN	Our FP	Comb FN	Comb FP
Spamhaus	11.54%	0.31%	0.97	8.6%	0.27%	5.32%	0.33%
Spamcop	22.32%	0.18%	0.98	11.5%	0.18%	6.56%	0.22%
SORBS	63.06%	0.10%	0.99	15.3%	0.17%	11.34%	0.20%

son is that these IPs are less frequently used by spammers, thus less likely observed by spam traps to be blacklisted. Indeed, we found that more than 75% of such IPs fall into smaller clusters (with active host size smaller than 15) which are potentially less likely abused by spammers given a limited number of likely compromised IPs. And yet since most IPs in the cluster send spam, the aggregated spam ratio of those clusters are high enough to identify newly appearing spammer IPs within the cluster.

7.2 Integration with SpamAssassin

The cluster reputation history collected can be used as a feature to predict future spam. We attempt to integrate it with SpamAssassin to quantify how many of its false negatives we can reduce by using cluster-based reputation history. As we have previously shown in Figure 5, about 4% of emails fall into the score range from two to five given the threshold of 5 which may contribute to false negatives. To obtain the ground truth of whether a particular email is spam, we would need to examine the email content. Due to privacy concerns, we can only examine several of our own personal accounts with permission. We also use a honeypot account with all its emails considered as spam along with the personal accounts to estimate the overall improvement from cluster-based reputation. SpamAssassin generates a false negative rate of 16% for the honeypot email account.

We study how much of SpamAssassin's false negatives can be reduced as well as how much false positives may be introduced by incorporating the cluster-based reputation scheme. We assign scores for IP addresses that fall within bad clusters with varying parameters and evaluate the accuracy as shown in Table 7. *FNR* stands for False Negative Reduced. *FPI* denotes False Positive Introduced. *Matched* indicates how many IP addresses fall into existing clusters built over 7-month of training data. The number of matched IPs serves as an upper bound for emails that can be classified as spam by the cluster-based scheme. *Spam ratio threshold* is the threshold for determining whether the cluster is considered bad, and additional score is added for an incoming email. *Score assigned* is the score to be added to the original score assigned by SpamAssassin.

Since we are not blocking emails directly based on cluster reputation, we relax the *spam ratio threshold* to be 0.7, 0.8 and 0.9 respectively with the *score assigned* to be 1, 2 and 3 respectively. We can see that for the honeypot account, we are able to detect about 50% of the missed spam by SpamAssassin when we set the threshold of spam ratio to 0.9 and the score assigned to 3. This is despite the fact that we only have the history for 60% of clusters that the spammer's IP addresses fall into. For other personal accounts, we observe similar false negative reduction with a fairly small amount of false positives introduced. In fact, if we use the spam ratio threshold of 0.9 and assigned score of 3, we only incur at most one false positive instance for all accounts which translates into only 0.0036% false positive rate. Upon inspection, the particular false positive email is a paper invitation sent from China (the conference was held in China) whose IP address falls into a cluster from which almost all of IP addresses sent purely spam to us. Interestingly, this IP has no reverse DNS name and is listed on SORBS blacklist which indicates that either the same machine is compromised at some point or the IP resides in a dynamic IP range (although we have checked that this IP is not identified as dynamic IP by UDMap).

On the other hand, with BGP cluster applied directly to the same account with spam ratio of 0.9 and assigned score of 3, although we can still reduce a similar number of false negatives, we observe 7 false positives introduced, clearly indicating the downside of its inaccurate administrative boundary. For personal account 3, we do not observe any false positives for any threshold experimented. However, we cannot reduce any false negatives either due to the fact there is only one false negative instance out of 11 spam emails by SpamAssassin and the IP address of this spam happens to fall within a cluster for which we do not have any history.

8 Concluding remarks

In conclusion, we have studied the characteristics of different types of network clusters and investigated how to combine them into a uniform one. We compare the performance of a combined clustering approach integrating both DNS and prefix information with previously proposed BGP prefix clusters and existing widely used IP-based blacklist (DNSBL) to demonstrate improved spam detection accuracy. We also integrate our proposed cluster-based reputation into SpamAssassin to catch 30-50% of the spam that are missed by SpamAssassin at the cost of very small false positive increase. Our technique is designed to be robust to potential evasion attempts due to the inherent stable properties of the network information used. Another advantage is that our system can work well in a single vantage point, thus can be easily deployed locally without requiring multiple vantage points (presumably much harder to obtain).

We argue that our cluster scheme is robust against various attacks. The most likely strategies of spammers would be to cause us to construct either too coarse-grained clusters where good and bad IPs are mixed or mislead us to construct too fine-grained clusters, which in the extreme become IP-based blacklists. We consider next how likely spammers can succeed in such endeavors.

BGP prefix information cannot be easily controlled by spammers unless they perform prefix hijacking attacks or own a fairly large prefix, both of which are unlikely due to high cost or overhead. DNS information is more amenable to modification, if spammers own an IP range and thus control its reverse DNS mapping. Spammers can construct rDNS names in a way that is most beneficial to them, *e.g.,* by setting rANS to be the same as that of their neighboring good IP ranges. To be truly effective, such neighboring IP ranges must belong to the same prefix as spammers' IP ranges. Furthermore, they need to make sure the rANS can resolve reverse DNS requests for them. They can also construct their rANS in a way that every single IP has a different rANS. This will cause our clustering algorithm to falsely cluster each IP into a separate cluster. However, this attack would again require spammers to own IP address ranges, and such rANS naming pattern itself would be an indication of malicious activities because constructing rANS in such a fashion is highly unusual.

IP-based blacklist captures the individual IP's history, which includes a sudden behavioral change of an IP address (*e.g.,* legitimate mail servers become compromised to send many spam). It is more difficult for the cluster-based approach to drastically modify a cluster's behavior as it must observe behavioral change for many IP addresses in the cluster. Note that by tracking history over a sufficiently long period of time, our approach can dynamically adapt to the behavioral changes in spamming. However, we expect the case where legitimate mail servers become compromised for spamming to be relatively rare (compared to DSL users get compromised and abused for spamming). In our data-set, as previously shown in §5.5, we did not observe much significant history changes for clusters. Another point to note is that our clustering approach attempts to capture regions of the Internet that "should" not have legitimate servers with high probability (*e.g.,* DSL clusters). In that sense, any sending host is potentially bad. The detailed analysis on the behavioral change of clusters is out of the scope of this paper and we plan to pursue as future work.

Acknowledgments

We thank our anonymous reviewers and Feng Qian for suggestions that improved the quality of this paper. This work is in part supported by NSF CNS-0643612, DARPA, and Department of Navy.

References

[1] The apache spamassassin project. `http://spamassassin.apache.org/`.

[2] Dialup rdns. `http://home.comcast.net/~mcwebber/blocking.txt`.

[3] Generic regular expressions for popular naming conventions. `http://www.ddf.net/spam/bad_relays.txt`.

[4] Microsoft: 3% of e-mail is stuff we want; the rest is spam. `http://arstechnica.com/security/news/2009/04/microsoft-97-percent-of-all-e-mail-is-spam.ars`.

[5] Rfc 4408, sender policy framework (spf) for authorizing use of domains in e-mail, version 1. http://tools.ietf.org/html/rfc4408.

[6] Rfc draft, suggested generic dns naming schemes for large networks and unassigned hosts. http://tools.ietf.org/id/draft-msullivan-dnsop-generic-naming-schemes-00.txt.

[7] SORBS. http://www.au.sorbs.net/.

[8] Spamcop. http://www.spamcop.net/.

[9] Spamhaus. http://www.spamhaus.org/.

[10] Whois ip address/domain name lookup. http://cqcounter.com/whois/.

[11] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker. Spamscatter: Characterizing Internet scam hosting infrastructure. In *14th conference on USENIX Security Symposium*, 2007.

[12] K. Chiang and L. Lloyd. A case study of the Rustock rootkit and spam bot. In *The First Workshop in Understanding Botnets*, 2007.

[13] Dynablock dynamic IP list. http://www.njabl.org/, recently aquired by spamhaus, http://www.spamhaus.org/pbl/index.lasso, 2007.

[14] S. Hao, N. A. Syed, N. Feamster, A. Gray, and S. Krasser. Detecting Spammers with SNARE: Spatio-temporal Network-level Automatic Reputation Engine. In *Proceedings of Usenix Security Symposium*, March 2009.

[15] T. Holz, C. Gorecki, K. Rieck, and F. Freiling. Measuring and detecting fast-flux service networks. In *Proceedings of the Network and Distributed System Security Symposium*, 2008.

[16] J. Jung and E. Sit. An empirical study of spam traffic and the use of dns black lists. In *IMC '04: Proceedings of the 4th ACM SIGCOMM conference on Internet measurement*, 2004.

[17] M. Konte, N. Feamster, and J. Jung. Dynamics of online scam hosting infrastructure. In *Proc. Passive and Actice Measurement Conference (PAM)*, 2009.

[18] B. Krishnamurthy and J. Wang. On network-aware clustering of web clients. In *In Proceedings of ACM SIGCOMM*, 2000.

[19] F. Li and M.-H. Hsieh. An empirical study of clustering behavior of spammers and group-based anti-spam strategies. In *CEAS 2006: Proceedings of the 3rd conference on email and anti-spam*, 2006.

[20] B. Medlock. An adaptive, semi-structured language model approach to spam filtering on a new corpus. In *CEAS 2006 - Third Conference on Email and Anti-Spam*, July 2006.

[21] A. Ramachandran and N. Feamster. Understanding the network-level behavior of spammers. In *Proceedings of Sigcomm*, 2006.

[22] A. Ramachandran, N. Feamster, and S. Vempala. Filtering spam with behavioral blacklisting. In *Proceedings of the 14th ACM conference on computer and communications security*, 2007.

[23] A. Ramachandran, N. Feamster, and S. Vempala. Filtering spam with behavioral blacklisting. In *CCS '07: Proceedings of the 14th ACM conference on Computer and communications security*, 2007.

[24] Route views project. http://www.routeviews.org.

[25] S. Sinha, M. Bailey, and F. Jahanian. Shades of Grey: On the Effectiveness of Reputation-based "Blacklists". In *Malware 2008*, 2008.

[26] S. Sinha, M. Bailey, and F. Jahanian. Improving spam blacklisting through dynamic thresholding and speculative aggregation. In *Proc. of the 17th Annual Network and Distributed System Security Symposium (NDSS)*, 2010.

[27] V. M. Telecommunications and V. Metsis. Spam filtering with naive bayes – which naive bayes? In *Third Conference on Email and Anti-Spam (CEAS)*, 2006.

[28] S. Venkataraman, S. Sen, O. Spatscheck, P. Haffner, and D. Song. Exploiting network structure for proactive spam mitigation. In *Proceedings of 16th USENIX Security Symposium on USENIX Security Symposium*, 2007.

[29] Y. Xie, F. Yu, K. Achan, E. Gillum, M. Goldszmidt, and T. Wobber. How dynamic are ip addresses? In *SIGCOMM*, 2007.

[30] Y. Xie, F. Yu, K. Achan, R. Panigrahy, G. Hulten, and I. Osipkov. Spamming botnets: Signatures and characteristics. In *SIGCOMM*, 2008.

[31] L. Zhuang, J. Dunagan, D. R. Simon, H. J. Wang, I. Osipkov, G. Hulten, and J. Tygar. Characterizing botnets from email spam records. In *LEET 08: First USENIX Workshop on Large-Scale Exploits and Emergent Threats*, 2008.