Measuring Email Sender Validation in the Wild

Casey Deccio
Brigham Young University
Provo, UT
casey@byu.edu

Tarun Yadav
Brigham Young University
Provo, UT
tarun141@byu.edu

Nathaniel Bennett
Brigham Young University
Provo, UT
bntnate5@byu.edu

Alden Hilton
Brigham Young University
Provo, UT
aldenhilton@byu.edu

Michael Howe
Brigham Young University
Provo, UT
mhowe20@byu.edu

Tanner Norton
Brigham Young University
Provo, UT
tannort5@byu.edu

Jacob Rohde
Brigham Young University
Provo, UT
jrohde@byu.edu

Eunice Tan
Brigham Young University
Provo, UT
eunice_tan@byu.edu

Bradley Taylor
Brigham Young University
Provo, UT
bradleytaylor@byu.edu

ABSTRACT

Email is a critical Internet application, and its security is important. The Sender Policy Framework (SPF), DomainKeys Identified Mail (DKIM), and Domain-based Message Authentication, Reporting, and Conformance (DMARC) were developed to enable mail servers to detect and reject email coming from fraudulent sources. In this paper we study the state of SPF, DKIM, and DMARC validation across a large number of mail servers, the first such study at scale that we know of. We consider two behaviors of sender-validating mail servers: behavior when an email with a valid sender is received and behavior when an email from a invalid sender is received. Our techniques allow us to elicit SPF, DKIM, and DMARC validation behavior of the servers without spam. We find that as many as 85% of mail servers are deploying SPF validation, and over half are deploying all three mechanisms: SPF, DKIM, and DMARC. We also observe there are some nuanced behaviors with regard to adherence to the SPF specification.

CCS CONCEPTS

• **Information systems** → **Email**; • **Networks** → *Network measurement*; *Security protocols*; *Naming and addressing*;

KEYWORDS

Email, SPF, DKIM, DMARC, Measurement, Security, DNS

ACM Reference format:
Casey Deccio, Tarun Yadav, Nathaniel Bennett, Alden Hilton, Michael Howe, Tanner Norton, Jacob Rohde, Eunice Tan, and Bradley Taylor. 2021. Measuring Email Sender Validation in the Wild. In *Proceedings of The 17th International Conference on emerging Networking EXperiments and Technologies, Virtual Event, Germany, December 7–10, 2021 (CoNEXT '21),* 13 pages. https://doi.org/10.1145/3485983.3494868

1 INTRODUCTION

Electronic mail (email) is the flagship application for Internet communication. While other messaging protocols have been widely adopted across the Internet since email protocols were initially developed, email continues to have a significant presence. Additionally, email is used for much more than general communication. It is among the principal methods for password reset, and therefore a gateway into many types of account. It is also a vehicle for fraud, fake news, and phishing. Securing email communications therefore contributes significantly to the general posture of security and privacy in Internet communications.

With email spoofing, an attacker sends an email to a victim, impersonating a party trusted by the victim. By making the email look as if it had come from a trusted party, the attacker can increase his/her effectiveness in convincing the recipient to trust the content of the email. Trusting such content exposes the recipient to attacks such as ransomware, adware, or other malware, leading to theft of confidential information. Detecting sender spoofing is thus a key first line of defense for protecting end users from email-borne exploits.

The Sender Policy Framework (SPF) is used to *detect* email messages with spoofed sender addresses. DomainKeys Identified Mail (DKIM) allows an organization to cryptographically *sign* outgoing messages, providing the receiving organization an additional mechanism to validate that signature and gain assurance that the sender is legitimate. SPF and DKIM are coupled with Domain-based Message Authentication, Reporting, and Conformance (DMARC) to *prevent* an email message with spoofed sender from even reaching a user's inbox, where the user might otherwise be compromised by trusting malicious content. However, detection and prevention

CoNEXT '21, December 7–10, 2021, Virtual Event, Germany
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-9098-9/21/12…$15.00
https://doi.org/10.1145/3485983.3494868

come only with proper deployment and configuration of *all* systems involved. In particular, the *domain owner* must publish and maintain policies for SPF and DMARC, the *sending mail server* must be configured to sign outgoing messages with DKIM and publish the public key, and the *recipient mail server* must perform SPF, DKIM, and DMARC validation. If publication of policies and signing of emails are not coupled with validation of policies, a spoofed email could make its way to the inbox of an unsuspecting end user.

In this paper we present an active measurement study that informs the state of SPF, DKIM and DMARC validation in the wild—the first *large-scale* study of its kind, to the best of our knowledge. We analyze the SPF validation behaviors of two data sets: domains associated with a mass email communication to world-wide operators in October 2020; and domains associated with DNS lookups for mail exchange (MX) records at our home institution, Brigham Young University (BYU), in February 2021. We seek to answer questions such as how many servers are validating SPF, DKIM, and DMARC, and what different validation behaviors are observed across the servers we analyze. We employ techniques that are designed to maximize the SPF validation activity from target servers, minimize what might be perceived as abuse, and execute the experiments in an efficient manner. We present the following as major contributions of this paper:

- a methodology for eliciting SPF validating behavior for analysis, without any illegitimate mail to user inboxes; and
- a large-scale analysis of SPF and DMARC validation behaviors employed by mail servers of popular domains.

As part of our study, we observed SPF validation activity from up to 85% of the domains that were part of the large-scale email communications. Of those domains, 53% utilize SPF, DKIM, and DMARC validation, and another 24% validate SPF and DKIM. We also observe nuanced behaviors from servers, such as DMARC validation with SPF or DKIM validation, or violation of DNS lookup limits imposed by SPF specification. For example, in the sample set of domains we analyzed, over 25% of SPF-validating mail servers violate the maximum number of DNS lookups set by specification. It is our hope that this study can be used to better understand and ultimately improve the quality of email sender validation across the Internet.

2 BACKGROUND

An SPF policy is used to designate the systems that are authorized to send email on behalf of a domain. A policy is a textual string consisting of various *mechanisms*, each of which is either: 1) a literal block of one or more IP addresses; 2) a hint that can be translated to one or more IP addresses using the Domain Name System (DNS); or 3) a recursive SPF policy to replace or extend the current policy [9]. The collection of IP addresses yielded by the policy constitute those explicitly authorized to send mail for a domain. To illustrate, we consider the following contrived policy for foo.com:

v=spf1 ip4:192.0.2.1 a:bar.foo.com include:foo.net -all

This policy indicates that 192.0.2.1 is a valid sender ("ip4"), as is the IPv4 or IPv6 address for bar.foo.com ("a"). The policy for foo.net is also included ("include"). Finally, no other senders are authorized ("-all").

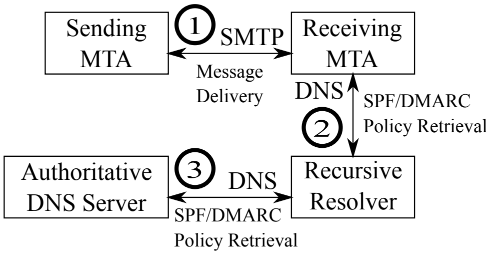

Figure 1: SPF policy retrieval using the DNS (2 and 3), in conjunction with SMTP message delivery (1).

SPF policies are published in and served from the DNS as TXT records [11, 12], as illustrated in Figure 1. When a Mail Transfer Agent (MTA) (i.e., *server*) sends the MAIL command with sender user@foo.com to another MTA over the Simple Mail Transfer Protocol (SMTP) [10] (1), the receiving MTA looks up the SPF policy for foo.com via a DNS query for foo.com type TXT. This query is sent by the MTA to its *recursive resolver* (2), which, in turn, queries the foo.com *authoritative servers* (3). The answer is returned to the recursive resolver, which returns it to the receiving MTA.

SPF specification establishes SPF syntax, allowed mechanisms, and guidelines for validation behavior, including handling of errors with retrieval of a policy (e.g., DNS lookup errors). However, what an MTA does with an email message that fails SPF validation is under-specified; indeed "there is no comprehensive normative requirement for message handling in response to any particular result" [9].

DomainKeys Identified Mail (DKIM) [3] is another mechanism for authentication of a sender, wherein the sending MTA inserts a cryptographic signature which can only be validated with the public key associated with the sender domain. The DKIM signature is included in a header in the email message. The public key to validate the signature is published in the DNS, as a record of type TXT. The DKIM header includes a reference to the domain name where that TXT record can be found—a subdomain of the sender domain. It is looked up in the same way an SPF policy is looked up, see Figure 1.

DMARC requires that either SPF or DKIM pass, and it gives a sender domain the ability to specify what an MTA should do with email that fails both. A DMARC policy also designates an email address to which violations should be sent. The DMARC policy is looked up in the same way an SPF policy is, as shown in Figure 1.

3 PREVIOUS WORK

Various studies have looked at the presence of SPF policies, DKIM signatures, and DMARC policies in use by mail *senders*. Mori, et al. [13] performed an early survey of SPF deployment, looking not only at the presence of SPF, but also of different types of syntactic errors found in SPF policies. Durumeric, et al. [5] measured SPF, as well as DKIM, DMARC, and STARTTLS by analyzing email messages arriving at Google's mail servers. They found that in 2015, Google was able to successfully validate 92% of email messages that it received using SPF; SPF validation of 0.42% failed due to failures retrieving policy; and all other messages come from domains with no policy. They uncovered a number of issues plaguing these deployments, including vulnerabilities that diminish their effectiveness. Adoption of various SMTP authentication mechanisms over time, including SPF and DMARC adoption, was studied by Gojmerac, et al [8]. They also looked at common errors in SPF configurations.

Two recent studies have investigated the prevalence of SPF and DMARC validation by *receiving* mail servers. Foster, et al. [7] studied the prevalence of SPF and DMARC validation by popular mail providers by looking for a DNS query corresponding to the SPF record they published, observing whether or not an invalid message was rejected during SMTP, and noting whether or not an invalid message that was delivered was flagged as spam. They observed DNS queries associated with SPF validation for 91% of the mail server providers they tested, DKIM-related DNS queries for 50%, and DMARC-related DNS queries for 41%. More recently, Scheffler, et al. [14], used SPF validation to investigate properties of DNS resolvers, indicating potential issues with the combination of a policy and a validation practice. They tested some of the limits of SPF validation, showing how some policies might cause degraded service in some configurations. However, their work was not so much a study of SPF deployment as it was a study of the side effects exhibited by servers that do deploy SPF. As such, they did not report on percentages relating to deployment.

The main differences between the previous validation studies and the current study are scale, depth, and perspective. The work of Foster, et al., quantified SPF, DKIM, and DMARC deployment using just a single test on the mail servers for just 22 popular mail providers. Scheffler, et al., performed three SPF validation tests on just over 8,000 MTAs discovered in a TCP SYN scan, but did not quantify SPF deployment. We perform multiple unique SPF validation tests on two sets of domains, one consisting of over 26,000 domains to which legitimate email messages are sent (i.e., for more than just measurement), and one consisting of over 22,000 domains, which are high-demand domains for a medium-sized institution of higher learning. This allows us to provide a broader and more detailed analysis of SPF validation behaviors.

Finally, in recent work, Shen, et al. [15], looked at ways in which the security provided by SPF, DKIM, and DMARC might be circumvented using clever techniques, such as header manipulation, subdomains of legitimate domains, and unauthorized forwarding.

4 EXPERIMENT DESIGN

In this section we describe the set of MTAs that we analyze and the methodology we employed for their analysis.

4.1 Data Sets

Our experiment design is driven by three main requirements:

- *Perspective.* We compare validation behavior of MTAs when confronted both with email messages that pass validation tests (i.e., SPF, DKIM, DMARC) and those that fail them.
- *Ethics.* Our measurement is minimally intrusive, neither causing degraded performance to any MTA nor illegitimate mail to any end user.
- *Representation.* The set of MTAs evaluated is relatively large and representative of email recipient domains.

NotifyEmail		**TwoWeekMX**	
TLD	**% Domains**	**TLD**	**% Domains**
com	26%	com	49%
net	13%	org	17%
ru	8.3%	edu	9.0%
pl	5.0%	net	6.3%
br	4.5%	us	3.6%
de	4.0%	gov	1.1%
ua	2.5%	uk	1.1%
it	1.9%	cam	1.0%
cz	1.6%	ca	0.76%
ro	1.6%	de	0.66%
Total TLDs: 259		**Total TLDs: 218**	

Table 1: Ten most prevalent TLDs corresponding to domains in the NotifyEmail and TwoWeekMX data sets, along with the percentage of domains with that TLD.

These three requirements, considered together, necessitate us evaluating at least *some* domains by sending a legitimate mass email communication. Sending messages that serve no purpose other than measurement would violate the ethics requirement. Sending *only* messages that purposely fail validation would violate the perspective requirement—and perhaps the ethics requirement, if those emails are accepted for delivery. Finally, sending legitimate emails to users representing only a small number of recipient domains would violate the representation requirement.

The mass email that we use for our experiment is in connection with an October 2020 communication to administrators of networks world-wide to disclose a vulnerability detected in their network. This vulnerability disclosure was in conjunction with a network measurement study identifying networks lacking destination-side source address validation (DSAV) [4]—a study otherwise unrelated to the current work. In total 26,695 domains (i.e., email suffixes) were extracted from the set of 42,924 email addresses whose email message was accepted for delivery—as indicated by a `250` (OK) response from the receiving MTA. The domains were distributed across 259 top-level domains (TLDs), the top 10 of which are shown in Table 1. The most represented TLDs in the data set are `com` (26% of domains), `net` (13%), and `ru` (8.4%). We refer to the resulting set of domains as the **NotifyEmail** set. While we use the **NotifyEmail** domains for testing MTA behavior when confronted with emails that we expect to properly validate, we also carry out an experiment for testing the behavior of those same MTAs when confronted with emails that are deliberately designed to *not* validate. While the domains are the same for both experiments, we refer to the data set as **NotifyMX** when used in conjunction with this latter experiment.

The **NotifyEmail** set is large relative to the set of domains previously studied. However, there is some bias in this data set, for which the data set does not fully meet the representation requirement. The domains are not randomly selected; rather, they are deliberately selected based on whether or not their network was vulnerable and whether or not their email address was legitimate. Relatedly, the domains are not necessarily high-demand in terms of email delivery.

To analyze a more representative set of domains, we integrate another data set. We collected the domain names for which MX queries were made by BYU's outgoing mail servers over a two-week period, February 1 through February 14, 2021. This data set, referred to hereafter as the **TwoWeekMX** set, was built from 23,735 domains for which queries of type MX were made by outgoing MTAs. These domains were further filtered to include only the 22,548 (95%) that actually yielded an answer when queried for an MX record type. We note that 27 of these have the suffix byu.edu, which is the domain name associated with BYU. Because these "local" domains represent only 0.12% of the **TwoWeekMX**, we are not concerned about this biasing the results.

Unlike **NotifyEmail**, we cannot send legitimate emails to these domains. Rather, these domains are used for testing MTA behavior in response to email messages that fail validation tests, in connection with the perspective requirement. To be true to the ethics requirement, we cannot depend on an MTA rejecting our email message. Thus, for this data set we always disconnect from an MTA before an email message is transmitted; there is no chance of delivery, independent of MTA's action or inaction. Our methodology is described in more detail in Section 4.6.

There are alternative approaches for selecting a representative set of domains for analysis. Several prominent lists of high-demand domains are available. However, none of these lists is primarily concerned with email recipient domains. The Alexa Top list and Majestic Million are based on Web site popularity. Cisco Umbrella is agnostic to application-layer protocol. While there is almost certainly some correlation between *some* domains that are popular due to their Web presence or non-protocol-specific DNS lookup frequency and their association with email delivery, there are several issues. One is that domains do not necessarily use the same domains for their Web infrastructure as they do for their corporate email. Relatedly, it is not safe to assume that all of these domains represent legitimate email suffixes. Rather than using popular domain lists, some researchers [14] have selected MTAs by performing a random scan for machines listening on port 25—the SMTP port—and then deriving possible target domains based on the IP addresses that responded. While this approach rules out domains that do not receive email, it does not distinguish high-demand domains from those that get little or no legitimate use.

In summary, we carry out three experiments to measure MTA behavior: one in which legitimate emails are sent and should properly validate (**NotifyEmail**); and two in which validation is designed to fail, but no emails are delivered (**NotifyMX** and **TwoWeekMX**). The methodologies used for these two different perspectives are discussed in subsequent sections. The **NotifyEmail** and **NotifyMX** data sets are largely the same, with differences discussed in Section 4.2. Table 2 contains a summary.

	Date	**Valid**		**MTAs**	
Data Set	**Run**	**Email**	**Domains**	**IPv4**	**IPv6**
NotifyEmail	Oct 2020	Y	26,695	17,252	1,599
NotifyMX	Jun 2021	N	26,390	26,196	2,700
TwoWeekMX	Apr 2021	N	22,548	10,666	471

Table 2: Data sets used for experimentation.

NotifyEmail		**TwoWeekMX**	
AS	**% Dom.**	**AS**	**% Dom.**
AS16509 (Amazon)	2.3%	AS15169 (Google)	32%
AS26211 (Proofpoint)	1.7%	AS8075 (Microsoft)	20%
AS22843 (Proofpoint)	1.6%	AS16509 (Amazon)	4.3%
AS46606 (Unified Layer)	1.3%	AS22843 (Proofpoint)	4.1%
AS16276 (OVH)	0.95%	AS26211 (Proofpoint)	3.2%
AS24940 (Hetzner)	0.92%	AS30031 (Mimecast)	2.3%
AS16417 (IronPort)	0.91%	AS14618 (Amazon)	1.7%
AS14618 (Amazon)	0.88%	AS26496 (GoDaddy)	1.6%
AS12824 (home.pl)	0.54%	AS46606 (Unified Layer)	1.3%
AS52129 (Proofpoint)	0.43%	AS16417 (IronPort)	1.2%
Total ASes: 10,937		**Total ASes: 1,795**	

Table 3: Ten most prevalent ASes corresponding to MTA IP addresses in the NotifyEmail and TwoWeekMX data sets. The percentage shown represents the fraction of the total domains in each data set. An AS is counted once for each domain having an MTA whose IP address is in an IP prefix announced by that AS.

4.2 Target MTAs

As noted previously, the **NotifyEmail** data sets consist of 26,695 domains. In October 2020, mail associated with **NotifyEmail** was delivered to MTAs at 18,851 different IP addresses, including 17,252 IPv4 and 1,599 IPv6. The IP addresses were distributed across 10,937 autonomous systems (ASes), looked up via the IP-to-AS mapping tool made available by the Center for Applied Internet Data Analysis (CAIDA) [6]. The top 10 ASes represented in the data set are listed in Table 3, ranked by the percent of domains that have MTA IP addresses announced by each AS. In the table, an AS is counted once for each domain having an MTA whose IP address is in an IP prefix announced by that AS.

To build the set of MTAs associated with **NotifyMX**, we performed DNS lookups of type A and AAAA for every name designated in MX records corresponding the **NotifyEmail** data set. The **NotifyMX** DNS lookups were executed in June 2021. Of the original 26,695 **NotifyMX** domains, DNS lookups failed to yield any IPv4 or IPv6 addresses for 305 (1%) of the total. The remaining 26,390 domains used for the **NotifyMX** constitute 99% of **NotifyEmail** domains. The DNS lookups yielded a total of 26,196 IPv4 and 2,700 IPv6 addresses.

What might seem like a discrepancy in number of IP addresses between **NotifyEmail** and **NotifyMX** is due to the fact that notification emails were delivered to only the first responsive MTA (see Section 4.6). However, even in that case, interactions with more than one IP address for a given domain name are still possible in several circumstances—for example, if a domain received more than one notification (e.g., because a single email address received more than one notification or because two addresses shared a common suffix), and a different MTA was selected each time, or if delivery to one IP address failed, and delivery at another IP address (i.e., corresponding to an MX record with equal or higher priority) succeeded.

For each of the 22,548 domain names returned in response to MX queries for the **TwoWeekMX** domains, we performed a lookup of type A and AAAA. We performed the lookups in April 2021. These address lookups resulted in 11,137 unique IP addresses—10,666 IPv4 addresses and 471 IPv6 addresses The top 10 ASes represented in the **TwoWeekMX** are listed alongside those for **NotifyEmail** in Table 3. Again, they are ranked by the percent of domains that have MTA IP addresses announced by each AS.

4.3 Experimental Configurations

We designed SPF, DMARC, and DKIM (**NotifyEmail** only) policies that allowed us to analyze MTA behavior with regard to the validation of these email security mechanisms.

The DMARC policy configuration is consistent across all three experiments. A strict reject policy was published for every domain from which experimental email was issued.

Because the purpose and methodology are so different for **NotifyEmail** than they are for **NotifyMX** and **TwoWeekMX**, the policies for SPF and DKIM are specific to our analysis of each. We next explain the experiment-specific SPF and DKIM policies.

4.3.1 ***NotifyEmail*** *Policies.* The SPF and DKIM policies for **NotifyEmail** are designed to convince the destination MTA, as much as possible, that our message was legitimate and should be delivered and trusted; the content was in fact an important notification. The SPF policy associated with domains from which **NotifyEmail** emails are sent identifies (only) the sending MTA's IPv4 and IPv6 addresses as legitimate senders. Every outgoing email is signed with DKIM, and the public key is made available in the DNS for validation.

The SPF policy for **NotifyEmail** is designed not only to properly authenticate the sending MTA, but also to elicit additional validation behavior for analysis. Specifically, it tests whether the remote MTA performs DNS lookups in serial or in parallel. It is described in more detail in Section 6.1 and Section 7.1.

4.3.2 ***NotifyMX/TwoWeekMX*** *Policies.* For the **NotifyMX** and **TwoWeekMX** experiments we designed 39 SPF test policies, each of which tested a specific behavior related to SPF validation. This set of test policies was collectively used to elicit and characterize the SPF validation behavior of a target MTAs. The fact that we created 39 SPF test policies is mentioned for completeness, though only a subset of the test policies are discussed in our results (Section 6 and Section 7). This is largely for two reasons. First, while the test policies were created based on research questions, not all the policies yielded useful results; we chose to include only the most interesting. Second, for some test policies, further analysis is needed to properly assess the results, which is beyond the scope of this paper.

In almost all cases the domain being evaluated is the suffix of the email address provided in the MAIL command. For one test policy the domain being evaluated is that provided in the EHLO/HELO command. These are described in more detail in Section 6.

The experimental SMTP activity associated with the **TwoWeekMX** domains does not include DKIM signatures. This is because the place for a DKIM signature is in the headers of an email message. However, as we explain in Section 4.6, our experimental activity with the **TwoWeekMX** MTAs stops short of including an email message, so there is no place for a DKIM signature.

4.4 Envelope To and From Addresses

We evaluated the email protection mechanisms employed by MTAs by interacting with each MTA over SMTP. For each email address (**NotifyEmail**) or MTA (**NotifyMX** and **TwoWeekMX**) we generate a uniquely identifiable **From** address to use in the SMTP communication with each (i.e., for MAIL command). **From** addresses associated with the **NotifyEmail** experiment use the template spf-test@**domainid**.dsav-mail.dns-lab.org, where **domainid** is a unique identifier corresponding to the domain tested. **From** addresses corresponding to the **NotifyMX** and **TwoWeekMX** experiments use the template spf-test@**testid**.**mtaid**.spf-test.dns-lab.org, where **mtaid** is a unique identifier corresponding to a specific MTA tested, and **testid** is an identifier corresponding to a specific test policy. For brevity, we will often use the generic, contrived suffix spf.com throughout the remainder of this document, in place of spf-test.dns-lab.org and dsav-mail.dns-lab.org.

For the **NotifyEmail** experiment the recipient email addresses (i.e., used with the RCPT SMTP command) are naturally those to which the notification email is being sent. For the **NotifyMX** and **TwoWeekMX** experiments, only a domain is available to us, so we have to contrive a username to create an email address. We apply each of the following usernames, in turn, to create the **To** address, until no SMTP error (i.e., invalid recipient) is encountered or we reach the last username: michael, john.smith, support, postmaster. The first names were simply common names that might actually correspond to a an actual account. postmaster is used as the fallback username; the presence of a postmaster account is generally expected to be a valid email recipient at a domain (e.g., to report misconfiguration or abuse).

4.5 SPF/DNS Instrumentation

Our control of the servers authoritative for dns-lab.org, the suffix common to all **From** domains, allows us to observe DNS queries related to our experimental activity. By extracting the query name from an incoming DNS query, we associate that query with the SPF/DKIM/DMARC validation activity of a domain or MTA for a specific test policy using the **domainid** or **mtaid** and **testid** fields (see Section 4.4). Because each **From** domain is unique, this association can be made even if multiple MTAs are validating simultaneously.

Many of the test SPF policies we created required DNS lookups beyond the initial TXT lookup with which the base policy was retrieved. For example, an "include", "a", or "mx" mechanism in a base policy results in a subsequent TXT, A/AAAA, or MX query, respectively. In order to uniquely associate any such follow-up query with MTA and test policy, query names for follow-up queries included the same identifying labels as the base query from which they were induced. For example, the policy for the **From** domain t01.foo.spf.com might include "include:l1.t01.foo.spf.com", while the policy for `t01.bar.spf.com` would instead include "include:l1.t01.bar.spf.com".

Given the necessity of unique domain names, both for **From** domains and follow-up DNS queries, our experiment required a large number of DNS records. For any one domain in the **NotifyEmail** experiment, 24 DNS records were required to handle the corresponding SPF/DKIM/DMARC policies. For any MTA in the **NotifyMX** or **TwoWeekMX** experiments a total of 704 DNS records were required to handle the set of 39 test policies. This translates to a total of $26695 \times 4 = 107K$ DNS records for **NotifyMX** and $39533 \times 704 = 27.8M$ DNS records for **NotifyMX** and **TwoWeekMX**. While authoritative DNS servers are typically capable of running easily on commodity hardware, generating and hosting nearly 28 million records is resource prohibitive, even for non-production experimentation.

To address the scalability issues with the required number of DNS records, we developed a custom authoritative DNS server that creates responses to SPF-related queries on-the-fly, synthesized from the query name. The server identifies the pattern of labels in the query name to produce the appropriate response, e.g., an SPF policy containing "include:l1.t01.foo.spf.com" in response to a `TXT` query for `t01.foo.spf.com`. This innovative solution not only allowed us to carry out the experiments for this study in a timely fashion but also to handle any future experimentation, using the same test policies, without any modifications.

4.6 Experimental SMTP Activity

We now describe the SMTP activity that was conducted by our dual-stack (IPv4/IPv6) SMTP client (acting as a sending MTA) to elicit SPF activity of candidate MTAs associated with the **NotifyEmail** and **TwoWeekMX** domains.

For the **NotifyEmail** set, we issued emails complying as closely as possible to specification, including mail server selection, IPv4 vs. IPv6 delivery, etc. To accomplish this, we issue the **NotifyEmail** emails using the Exim4 MTA, with mostly default settings plus DKIM signing. Once an email is delivered for a given domain, using a given MTA, designated by `MX` records, no further MTAs are probed.

For the **TwoWeekMX** domains, we interacted with *every* MTA (i.e., not just one per domain) using a **From** address for every one of the test policies in Section 4.3.2. This interaction was carried out using a custom mail client developed specifically for this research. The client establishes a TCP connection with a given MTA and issues the following commands:

- `EHLO` (or, if unsupported, `HELO`);
- `MAIL`, using the envelope **From** address corresponding to the MTA and test policy (see Section 4.4);
- `RCPT`, using the envelope **To** address corresponding to a domain for which the target MTA was designated for delivery (see Section 4.4); and
- `DATA`.

We introduce a 15-second "sleep" immediately before issuing each of the `MAIL`, `RCPT`, and `DATA` commands This allows us to more definitively time the SPF validation relative to the SMTP interactions (see Section 6.2), and it reduces our footprint in terms of resource consumption of the MTA, in compliance with the ethics requirement. After the server response from the `DATA` command, we *disconnect* our TCP connection without sending any message data. There is thus *no chance* of an email being accepted for delivery because there is no message. This complies with the requirement of minimally intrusive experimentation (see Section 4.1).

5 ETHICAL CONSIDERATIONS

Abiding by ethical principles associated with abuse and exploit is important when conducting measurement of Internet systems. We took the following measures to minimize perceived abuse as part of our experimentation.

5.1 No Illegitimate Emails Delivered

As mentioned in Section 4.1 and Section 4.6, the only email messages actually delivered were those that were associated with the notification emails for the **NotifyEmail** experiment. All other emails—which were mostly designed to fail validation anyway—stopped short of actually delivering a message.

5.2 Minimal Impact on MTAs

In connection with our ethics requirement of not causing any degradation of service to any MTA, we took several measures to minimize the impact of our activity on any given MTA or domain. First, while we communicated with every **NotifyMX** and **TwoWeekMX** MTA to carry out 39 validation tests, we shuffled the order in which MTAs were analyzed to decrease the chance that MTAs for the same domain were analyzed at the same time. Second, as mentioned in Section 4.6, we inserted a 15-second "sleep" between each test, rather than issuing requests in parallel or back-to-back, to minimize our footprint. Finally, as explained in Section 4.2, we experimented with every MTA designated by `MX` and, by association, `A` and `AAAA` records, for **NotifyMX** and **TwoWeekMX**. In the case where multiple domains in a given data set designated the same IP address as a destination MTA, only one of the domains was selected as a recipient domain. This means that a given MTA would only be analyzed once—except in the case where multiple IP addresses (including IPv4 and IPv6) were associated with the same MTA. Our analysis of that MTA was then applied to all domains that designate that MTA. In the case of **NotifyEmail** domains, we typically interacted with only one MTA per email address, although email addresses with a common suffix might result in multiple communications with the same MTA, as noted in Section 4.2.

5.3 Contact and Experiment Attribution

We created several ways by which we might be contacted by those responsible for the MTAs that were the targets of our measurement activity. With regard to the **NotifyEmail** activity, the primary purpose of the interactions was to communicate a message to the individual(s) at the email addresses—even if each came from a unique **From** address (the **From** email header matched the envelope **From** address, so DMARC would pass). Although the **From** address for any notification email uses a domain distinct from that of all other notification emails, we included a **Reply-To** header in the message body, so they could correspond with us, whether it was with respect to the notification itself or for questions about the abnormal **From** domain.

The **NotifyMX** and **TwoWeekMX** communications did not use a **Reply-To** header because there was no email message transmitted at all; the only identifying information associated with this part of the experiment are a client MTA address and an envelope **From** address. In this case, if an entity wanted to contact us because of anomalous SMTP traffic, they would need to review the MTA's logs and send a message to the **From** address therein, or use the IP address to look up contact information.

In the case that, for either experiment, an entity looks up contact information using the DNS, we published a contact email address in the `RNAME` (responsible name) field of the `SOA` record for the common domain we used. We also published that email address in the DMARC record associated with every **From** domain. Finally, the IP address from which we initiated our SMTP connections ran a Web server with more detailed information on the experiment itself and how to opt out. We were not contacted by any entity regarding our activity.

5.4 Ethics Board Review

In addition to the other measures that we took to abide by ethical principles with our experiment design, we consulted with individuals within our institution for an assessment of our study. Institutional Review Boards (IRBs) are often used for this purpose. We met with our institution's IRB, and they deemed our study exempt from formal IRB evaluation because our research did not involve human subjects. Therefore, we asked constituents of the IRB for assess our study informally as an ad-hoc ethical review board. Their consensus was that there were no concerns with our experiment design.

6 SPF, DKIM, DMARC VALIDATION ASSESSMENT

We now answer one of the most important questions driving our research: how many domains and servers are performing SPF, DKIM, and DMARC validation on incoming email messages?

For the purposes of this paper, we refer to an MTA as *SPF-validating*, *DKIM-validating*, or *DMARC-validating* if it issues at least one DNS query in conjunction with SPF, DKIM, or DMARC validation, respectively, for any one or more of our test policies. Similarly, we refer to a domain as SPF-validating if it has one or more SPF-validating MTAs.

6.1 NotifyEmail Results

We first consider the **NotifyEmail** data set, which is composed of domains to which legitimate email messages were sent, and validation of all types was expected to pass. Of the 26,695 domains to which email was issued, 22,703 (85%) were found to be SPF-validating, accounting for 15,323 (81%) of the receiving MTAs. Additionally 21,814 (82%) domains were found to be DKIM-validating, and 14,436 (54%) were found to be DMARC-validating.

A complete breakdown of SPF, DKIM, and DMARC validation for **NotifyEmail** is found in Table 4, and a summary of SPF validation for all experiments is found in Table 5.

We note that over half (53%) of domains employed validation of SPF, DKIM, and DMARC. With this particular experiment (**NotifyEmail**) we have no way of knowing if the MTAs associated with these domains would actually flag or discard an email for which SPF and/or DKIM validation failed—because the emails we sent were designed to pass validation. Nonetheless, the DNS lookups we observed are a good sign that the domains are validating. Nearly one quarter of the domains (24%) appear to be in "trial" mode with SPF and DKIM—that is, they are willing to validate the sender with both technologies but not enforce the validity with DMARC. Just under one in five domains (17%) do not perform any sort of sender validation; this is perhaps the most alarming of the combinations.

SPF	DKIM	DMARC	NotifyEmail Domains
✓	✓	✓	14,056 (53%)
✓	✓	×	6,322 (24%)
×	×	×	4,456 (17%)
✓	×	×	2,156 (8.1%)
×	✓	×	1,436 (5.4%)
×	×	✓	211 (0.79%)
✓	×	✓	169 (0.63%)
×	✓	✓	0 (0.0%)

Table 4: A breakdown of SPF, DKIM, and DMARC support exhibited by domains in the NotifyEmail experiment.

	Total		SPF-Validating	
	Domains	MTAs	Domains	MTAs
NotifyEmail	26,695	18,851	22,703 (85%)	15,323 (81%)
NotifyMX	26,390	28,896	13,538 (51%)	14,560 (50%)
TwoWeekMX All	22,548	11,137	2,949 (13%)	1,574 (14%)
TwoWeekMX Decile 1	2248	2424	290 (13%)	345 (14%)
TwoWeekMX Decile 2	2248	1926	334 (15%)	381 (20%)
TwoWeekMX Decile 3	2248	1986	309 (14%)	355 (18%)
TwoWeekMX Decile 4	2248	1663	275 (12%)	272 (16%)
TwoWeekMX Decile 5	2248	1540	286 (13%)	233 (15%)
TwoWeekMX Decile 6	2247	1815	374 (17%)	337 (19%)
TwoWeekMX Decile 7	2248	1617	235 (10%)	280 (17%)
TwoWeekMX Decile 8	2248	1563	292 (13%)	305 (20%)
TwoWeekMX Decile 9	2248	1589	287 (13%)	287 (18%)
TwoWeekMX Decile 10	2247	1656	263 (12%)	275 (17%)

Table 5: SPF-validating domains and MTAs observed in the experiments.

Table 6 shows the validation status of 19 of the 22 popular mail providers examined by Foster, et al., in 2015 [7]. The level of SPF among these providers is consistent with that of all domains in

Domain	SPF	DKIM	DMARC
hotmail.com	✓	✓	✓
gmail.com	✓	✓	✓
yahoo.com	✓	✓	✓
aol.com	✓	✓	✓
gmx.de	✓	✓	×
mail.ru	✓	✓	✓
yahoo.co.in	✓	✓	✓
comcast.net	✓	✓	✓
web.de	✓	✓	×
qq.com	×	×	×
yahoo.co.jp	✓	✓	✓
naver.com	✓	✓	✓
163.com	×	×	×
libero.it	✓	✓	✓
yandex.ru	✓	✓	✓
daum.net	✓	✓	×
cox.net	✓	✓	✓
att.net	×	×	×
wp.pl	✓	✓	✓

Table 6: A breakdown of SPF, DKIM, and DMARC support exhibited by popular mail providers as observed by the NotifyEmail experiment. This list is composed of 19 of the 22 popular mail providers examined in previous work [7].

	NotifyEmail Domains		
		In Alexa	In Alexa
	All	Top 1M	Top 1K
All	26,695	2,953	87
SPF-Validating	22,701 (82%)	2,608 (88%)	81 (93%)
DKIM-Validating	21,812 (82%)	2,473 (84%)	78 (90%)
DMARC-Validating	14,434 (54%)	1,973 (67%)	69 (79%)

Table 7: A breakdown of SPF, DKIM, and DMARC support exhibited by all NotifyEmail domains, as well as the subsets that are in the Alexa Top 1M and Alexa Top 1K.

the **NotifyEmail** data set: 16 of 19 (84%) of the mail providers performed a DNS lookup to retrieve an SPF policy. Interestingly, however, some of the providers analyzed by Foster, et al., to be SPF-validating did not issue DNS queries in our study: `qq.com`, `163.com`, and `att.net`. Encouragingly 13 of 19 (68%) mail providers performed SPF, DKIM, and DMARC validation, up from 23% in the 2015 study [7].

We also consider the domains in our data set that are from the Alexa Top list [1], as it was on October 12, 2020. While we chose not to use the Alexa list for our target set of domains to analyze (see Section 4.1), we believe that there is merit in assessing those domains from the **NotifyEmail** data set that are also in the Alexa Top list. The trend shown in Table 7 is that domains in the Alexa Top 1M exhibit higher validation rates in SPF, DKIM, and DMARC than the general data set, and domains in the Alexa Top 1K exhibit higher validation rates in all three categories than domains in the Alexa Top 1M.

We also observed several inconsistencies with MTA validation behavior. As indicated by Table 4, MTAs for 169 domains looked up DMARC policies but did not look up either SPF policy or DKIM public key. The discrepancy here is that DMARC requires at least SPF or DKIM to validate. While this represents less than 1% of domains, the fact that the behavior exists in the wild is bewildering.

Additionally, 690 (3.0%) of the 22,703 domains that exhibit SPF validation don't actually *finish* SPF validation. To explain how we came to this conclusion, we briefly describe the test policy used with the **NotifyEmail** experiment. While the policy certifies our client as a valid sender (see Section 4.3.1), the receiving MTA must perform several DNS queries to come to that conclusion. The policy contains an "a" mechanism, and the domain name corresponding to that "a" mechanism resolves to the IPv4 and IPv6 address of the server that sent the notification emails. Without actually performing the `A` or `AAAA` lookups necessary to resolve the name and learn the address, there is no way to know that the sender address is valid. In the case of the 690 partial validators, a `TXT` query for the SPF policy was observed but no `A` or `AAAA` query was observed. In 86 (12%) of those cases, SPF is relied on exclusively; no DNS query for the DKIM public key was observed. However, of the 86, only 3 show signs of possible enforcement, as evidenced by a DMARC query.

6.2 NotifyMX Results

We now consider the **NotifyMX** data set, which has a nearly identical set of domains but is viewed from a different perspective. Of the 28,896 MTAs we probed, only 14,560 (50%) show evidence of SPF validation. These account for only 13,538 (51%) of the domains in the data set—much lower than the 85% that exhibited validation behavior when an email was actually sent. We observed a contrast in validation behavior for a total of 15,316 domains—which is 58% of the domains common to both experiments. In 14,584 (95%) of cases of inconsistency, the domain was shown to be SPF-validating in the **NotifyEmail** experiment, but not in the **NotifyMX** experiment, nine months later—that is, only 65% of those observed to be SPF-validating with **NotifyEmail** exhibited SPF validation in **NotifyMX**. A study of this difference is important for better understanding the limitations of the methodology we employed for the **NotifyMX** and **TwoWeekMX** experiments.

One hypothesis to explain the difference in SPF validation rate is that not all SPF-validating MTAs initiate SPF validation before the email message is actually transmitted (i.e., after the `DATA` command). To test this hypothesis, we compare the timestamp of every email sent in the **NotifyEmail** experiment with that of the corresponding DNS query for the SPF policy, which amounted to 36,812 emails. Because there might be multiple, identical DNS queries for a given policy, we consider the query with the *earliest* timestamp. We then subtract the difference between the email delivery timestamp and the SPF lookup timestamp. Because our Exim4 setup uses a timestamp with a granularity of seconds and not sub-seconds, we cannot accurately compare timestamps with a difference between 0 and 1 second. We filter out the 3,169 (8.6%) emails with timestamps in that window. We also remove 7 emails with a timestamp difference of several days (e.g., because of an earlier delivery attempt that triggered SPF validation and a later delivery that did not). At this point, we have 33,634 email addresses representing 21,091 domains.

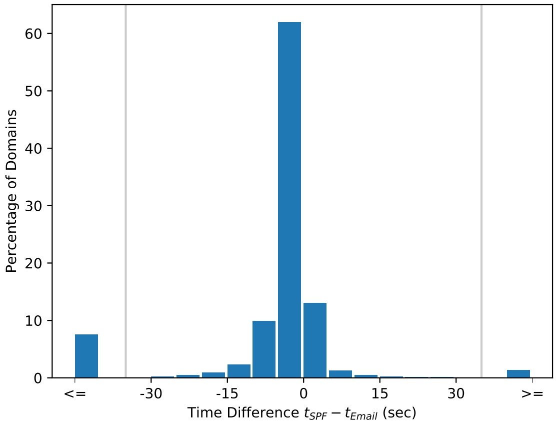

Figure 2: Distribution of the difference in timestamp between when the DNS query for an SPF policy was received (t_{SPF}) and when the corresponding email message was delivered (t_{Email}) in the NotifyEmail experiment.

We check the consistency in the timestamp difference (i.e., positive or negative) across emails for a given domain. Only 25 (less than 1%) of domains have inconsistent timestamp differences. For those domains with consistent timestamp differences across all emails, we average the differences and plot the distribution in Figure 2. The plot shows that for 17,550 (83%) domains the difference in timestamps was negative, indicating that the DNS query for the SPF policy was received *before* email delivery was complete; for the remaining 17%, the SPF policy was not looked up until *after* email delivery. As a side note, for 91% of results the timestamp difference is between -30 seconds and 30 seconds.

Our analysis of timestamps supports our hypothesis that not all SPF validation happens real-time, during the SMTP communication; for nearly 1 in 5 domains in **NotifyEmail**, SPF validation occured only after email delivery. This offers a partial explanation for a lower validation rate in the **NotifyMX** experiments: no email was actually accepted for delivery. However, there is still a discrepancy in that roughly 35% of SPF-validating domains did not exhibit validation behavior in the **NotifyMX** experiment, but the observed rate of SPF-policy lookup after mail delivery is only 17%. In looking closer at the data, 7,803 (27%) MTAs returned an error message containing the (case-insensitive) string "spam", and 872 (3.0%) issued an error message referencing "blacklist", before our client issued the `DATA` command. Together, these account for 8,143 (28%) of the total MTAs assessed. Our experimental activity related to **NotifyMX** landed our client on several blacklists, which was a inhibitor for some MTAs involved.

6.3 TwoWeekMX Results

The difference in results between the **NotifyEmail** experiment and the **NotifyMX** experiment (see Section 6.2) show the challenges that come with trying to infer validation behaviors without legitimate recipient email addresses. Without this legitimacy, the **NotifyMX** and **TwoWeekMX** experiments cannot be relied on to accurately quantify the presence of SPF, DKIM, and DMARC validation in MTAs. However, they can be used as a heuristic for quantifying a lower bound of SPF deployment in a more general data set, in compliance with the representation requirement. They can also be used for analyzing the validation behavior of a subset of MTAs in accordance with the perspective requirement.

With that scope and purpose defined, we begin our analysis with a high-level report of SPF validation, summarized in Table 5. Of the 11,137 **TwoWeekMX** MTAs we analyzed, we only observed SPF policy lookups for 1,574 (14%) of MTAs, accounting for 2,949 (13%) of domains in the data set. These percentages are significantly lower than even those in **NotifyMX**. We addressed some of the challenges in comparing the analysis of the **NotifyMX** with **NotifyEmail** experiments in Section 6.1. However, **TwoWeekMX** has additional considerations, the most significant of which are that it has a very different data set and that we don't have the email address of an actual user—the **To** addresses are simply guesses.

We first further evaluate the **TwoWeekMX** data set itself, looking at both its uniqueness and the SPF validation consistency within. Only 748 (3.3%) of the **TwoWeekMX** domains were also present in the **NotifyEmail** set. That quantifies the overall uniqueness of the set. However, we hypothesized that perhaps the higher-demand domains were better provisioned, such that the less domains in less demand domains were bringing down the overall average. To test that theory, we divided the domains into deciles (10 percentiles), ordered by the number of queries issued for each domain during the collection period. Decile 1 consists of the 10% of domains with the most queries, Decile 2 the next most queried 10% of domains, etc. For this part of the experiment, we excluded the "local" (i.e., `byu.edu`) domains; although they generally represented only 0.12% of all **TwoWeekMX** domains (see Section 4.1), their query counts were disproportionately higher than those of other domains, so they consistently showed up in the top Decile, potentially biasing these results. The results are shown in Table 5. Contrary to our hypothesis, the frequency of SPF validation amongst different demand of the **TwoWeekMX** data set were very consistent, with means of 13% of domains and 17% MTAs validating. The standard deviation was 1.7% and 1.8% for domains and MTAs, respectively. The take-away is that the overall fraction of SPF-validating resolvers observed is consistent across the data set, indicating that other contributors are at play for the low percentage of SPF-validating resolvers.

The second consideration is the **To** address used. The first major concern with guessing a recipient address is that it might not be valid. In this case, the destination MTA might simply refrain from performing SPF validation, even if they already had the domain from the **From** address from the `MAIL` command. For example, 716 (6.4%) MTAs from the **TwoWeekMX** set returned an SMTP error related to invalid recipient.

A more challenging concern is that when an email is received for `postmaster`, many MTAs are configured to bypass any sender validation mechanisms, such as SPF. Such "whitelisting" would result in a decrease in observed SPF validation. This is one of the unfortunate side effects of our methodology, which we have chosen in adherence to our ethics requirement. The `postmaster` account was ultimately used as the recipient for 69% of all MTAs with which we communicated. This was because our SMTP communications using

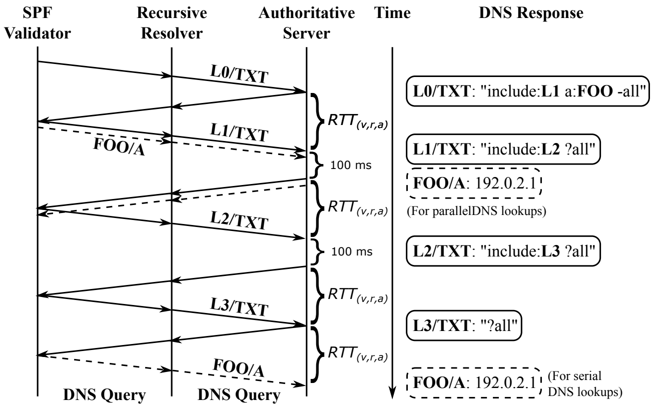

Figure 3: A timeline of DNS communications involving an SPF validator, a DNS recursive resolver, and a DNS authoritative server, in conjunction with an SPF test policy. `FOO` **and** `192.0.2.1` **represent a domain name and corresponding IP address, respectively. The DNS query that might be performed either serially or in parallel is represented by dashed lines.**

other user names resulted in rejection before the `DATA` command was issued.

7 SPF VALIDATION BEHAVIORS

We now examine SPF validation behaviors other than simply whether they look up a policy (SPF-validating). We use the 39 test policies to answer questions about SPF validation behavior and better understand compliance, consistency, and configuration. We consider the most interesting and salient behaviors for our discussion. The **TwoWeekMX** data is used exclusively, unless specified otherwise.

We begin by establishing some terminology, with regard to test policy descriptions. We refer to the policy associated with the **From** domain as the "Level 0" (L0) policy. Every "include" mechanism induces an additional policy lookup. Any policy corresponding to an "include" mechanism that is part of a Level N (LN) policy is referred to as a Level $N + 1$ policy.

7.1 Serial vs. Parallel DNS Lookups

One test policy was designed to determine whether the DNS lookups associated with SPF validation are performed serially or in parallel: are DNS lookups a) made only on demand, as an IP address is being matched against each respective SPF policy mechanism, or b) executed in parallel, in advance of their potential use? The answer to this question might help implementors understand how policies might be optimized for efficiency, considering their evaluation.

As illustrated in Figure 3, the L0 policy for the test contains an "include" mechanism before an "a" mechanism. In the case of **NotifyEmail**, the domain associated with the "a" mechanism resolves to the IP address of the email sender, i.e., is *valid*; in the case of **NotifyMX** and **TwoWeekMX**, it resolves to an unaffiliated IP address (`192.0.2.1`), i.e., is *invalid.* The L1 policy is composed of an "include" for a single L2 policy, and the L2 policy similarly includes only a single L3 policy. The L3 policy contains only "?all". The right side of Figure 3 shows a simplified representation of these embedded policies.

The outcome of the test policy can now be determined by observing the order of the DNS lookups observed at the authoritative DNS server. A child policy can never be looked up before a parent policy because the child policy is not known before the parent policy is retrieved. Thus, the L1 policy must be looked up before the L2 policy and L2 before L3. Let the round-trip time (RTT) between the SPF validator and its DNS resolver be $RTT_{(v,r)}$, the RTT between the DNS resolver and the authoritative DNS servers be $RTT_{(r,a)}$, and the combined RTT be $RTT_{(v,r,a)} = RTT_{(v,r)} + RTT_{(r,a)}$. We additionally insert a 100ms delay before responding to the L1 and L2 queries. Thus, the minimum time elapsed between the arrival of the query for the L0 policy and the arrival of the query for the L3 policy is $3RTT_{(v,r,a)} + 200ms$. If DNS lookups are performed serially, then the `A`/`AAAA` query associated with the "a" mechanism will not be observed until at least $4RTT_{(v,r,a)} + 200ms$ after the query for the L0 policy is observed. If, however, the SPF validator performs DNS lookups in parallel, then the query will almost certainly come before the L3 policy query, and as early as $RTT_{(v,r,a)}$ after the L0 policy query. The difference in minimum elapsed time between the two strategies is therefore $3RTT_{(v,r,a)} + 200ms$. This is illustrated in Figure 3.

Because of the relatively high difference between minimum time elapsed for the two strategies—parallel vs. serial—we can infer with high confidence that DNS lookups are performed serially if the `A`/`AAAA` query arrives after the L3 `TXT` lookup and in parallel otherwise. In our experimentation, 1,432 MTAs were tested, and of those, 1,392 (97%) performed DNS lookups serially.

A general recommendation is difficult to make, as to which methodology is the more effective strategy. With parallel DNS lookups, an SPF validator might save time in evaluating more complex policies because the DNS lookups associated with validation will already have been performed—as much as possible. On the other hand, serial lookups are more conservative in terms of resources and complexity, but might result in worse performance. However, because 97% of MTAs perform DNS lookups serially, we recommend that organizations create their policy in such a way that the most frequently used addresses come first.

7.2 DNS Lookup Limits

The SPF specification sets various limits for MTAs performing SPF validation. For example, it suggests (using RFC 2119 "SHOULD" [2]) that a timeout be set; while no preferred value is given, it should be more than 20 seconds [9]. If the timeout value is exceeded, then "temperror" should be returned. The number of mechanisms that trigger DNS lookups should also be limited to 10, "to avoid unreasonable load on the DNS" [9]. If this limit is exceeded, then "permerror" MUST [2] be returned.

To test MTA compliance with the SPF specification for both DNS lookups and timeouts, one test policy included a total of 30 "include" mechanisms, illustrated in Figure 4. Each box in the figure represents a policy which is evaluated in the order corresponding to the number in the box, and the policy levels are written on the far left. An 800ms delay is imposed on every DNS response for query names in L1 through L5. Thus, the test policy resulted in a maximum

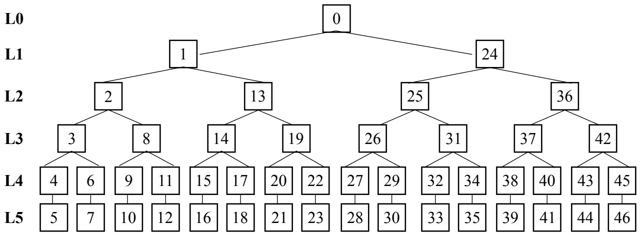

Figure 4: A test policy designed to test the SPF limits of an MTA.

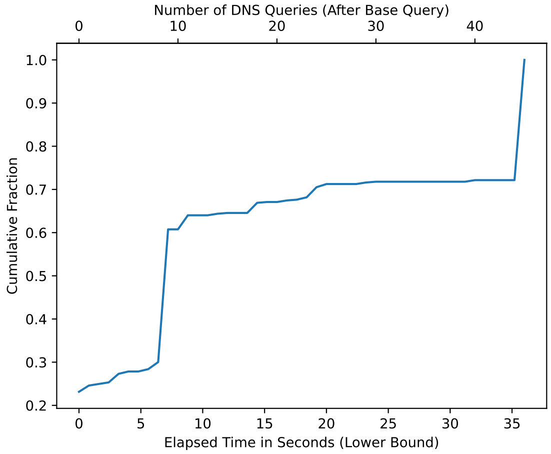

Figure 5: Results of testing DNS limits of both DNS lookups and time elapsed.

of 46 DNS lookups for any SPF-validating MTA—aside from the initial L0 policy lookup—and up to 36 seconds of validation time (i.e., 45lookups × 800ms). We note that Scheffler, et al., investigated the DNS lookup limit in their study, in similar fashion. However, their results only used aggregate numbers to infer violations; they did not have insight into the behaviors exhibited by individual MTAs [14].

By looking at the query name for the *last* query made by an MTA during its evaluation of the policy, we can determine 1) the number of DNS queries that were issued in connection with this policy evaluation and 2) the lower bound for the elapsed validation time. For example, if we observed that the last query from an MTA was the L4 query in box 6, then we would know that 1) six DNS queries had been issued, in addition to the base L0 query, and 2) at least 4 seconds (800ms × 5 queries) had elapsed since beginning validation.

The results of our analysis of both elapsed time (lower X-axis) and DNS lookup count (upper X-axis) are shown as a Cumulative Distribution Function (CDF), in Figure 5. We evaluated a total of 553 MTAs for this test because they looked up an SPF policy for the corresponding test. Of those MTAs, 336 (61%) halted their analysis before 10 DNS queries were issued. However, 154 (28%) MTAs executed all 46 DNS queries, with SPF validation taking more than 36 seconds!

7.3 Other Behaviors

The SPF specification recommends that MTAs look up the policy for the domain specified in the HELO command: "if a conclusive determination about the message can be made based on a check of 'HELO', then the use of DNS resources to process the typically more complex 'MAIL' can be avoided" [9]. To that end, one test policy includes a simple reject all (i.e., "-all") policy for the domain specified with HELO. However, of the 1,473 MTAs that performed SPF validation for this policy, only 73 (5.0%) looked up the TXT record associated with the policy for the HELO domain. *Every one* of those MTAs proceeded to look up the policy for the MAIL domain—effectively ignoring the earlier validation failure.

A number of MTAs ignored SPF syntax errors. When validating against a test policy with that had syntax errors in the main policy ("ipv4" instead of "ip4"), 79 (5.5%) of 1,444 validating MTAs continued validating, as evidenced by DNS lookups for mechanisms to the right of the errors. The results were slightly worse when the syntax errors were in a child policy instead of the main policy: 170 (12.3%) of 1,377 continued validation in the parent policy, despite the errors in the child policy. This behavior is in direct violation of specification, which indicates syntax errors should fail with "permerror, without further interpretation or evaluation" [9].

The SPF specification uses the term "void lookup" to describe a DNS lookup that results in no DNS records. It recommends ("SHOULD" [2]) that validators permit only two void lookups. While the limit may be configurable, a policy that exceeds the limit should result in "permerror" [9]. One test policy includes five "a" mechanisms, none of which resolve to an IP address. Of the 1,229 MTAs validating this policy, 1,193 (97%) exceeded the recommended two void lookups, and 788 (64%) looked up all five names!

When an MTA is determining where mail for a domain should be sent, and an MX lookup has failed, it follows up with an A/AAAA query [10]. That follow-up query is explicitly disallowed by the SPF specification when following an "mx" mechanism. One test policy tests this behavior by including an "mx" mechanism for which the domain does not resolve. Of the 1,338 MTAs validating this policy, 189 (14%) performed an A/AAAA lookup, against specification.

While multiple DNS records of type TXT might exist at a given domain name, only one may contain an SPF policy for that domain, according to the SPF specification. The existence of multiple SPF policies for a domain name creates ambiguity as to which policy should be adhered to. Thus, when multiple SPF policies are detected in the set of TXT records for a domain, the SPF validator should fail with "permerror". [9]. One test policy included two TXT records, each with a valid SPF policy; each included an "a" mechanism with a distinct domain name to resolve. Of the 1,368 MTAs tested, 1,058 (77%) followed the expected, specified behavior, by following *neither* of the policies; there were no queries for either policy-specific name. However, in 310 (23%) cases, MTAs followed *one* of the policies. Fortunately, in no case did an MTA follow *both* of the policies.

One test policy requires a DNS resolver to retrieve an SPF policy from an authoritative server over TCP. This is done by having the authoritative server return a truncated response for any UDP query received, eliciting a TCP follow-up. Fortunately, nearly all MTAs have resolvers that are able to speak TCP; just 2 of the 1336 resolvers that queried for the truncated resource over UDP did not attempt TCP after its UDP query.

One test policy requires a DNS resolver to retrieve an SPF policy over IPv6; the authoritative servers for the domains related to this policy only have `AAAA` records with IPv6 addresses. Of the 1,370 MTAs that performed SPF validation with this test, only 675 (49%) were able to retrieve the policy over IPv6. However, these MTAs only represent 599 (29%) of 2,055 domains involved in SPF-validating this test policy.

The SPF specification strictly limits the number of `A` or `AAAA` lookups associated with the `MX` record set resulting from an "mx" mechanism. After 10 lookups, the validator MUST return "permerror" [9]. One test policy included an "mx" mechanism yielding 20 `MX` records. Of the 1,057 MTAs that validated this test policy only 81 (7.7%) stopped after 10 `A`/`AAAA` lookups or fewer; the rest were in violation of the specification. Nearly two-thirds of all validating MTAs (679 or 64%) issued queries for all 20 `MX` records!

8 DISCUSSION AND FUTURE WORK

We now consider some of the major takeaways of our study, including limitations, open suggestions, and future work.

First, our methodology has limitations, as noted throughout our analysis. For example guessing email addresses (e.g., `postmaster`) for recipient domains was largely ineffective. To account for this shortcoming, we complemented our general analysis of in-demand domains with an analysis of domains that were the subject of a mass email communication. However, as noted previously, there is some bias in this additional data set. An idea for strengthening the methodology would be to make a Web-based tool available for comprehensively assessing SPF, DKIM, and DMARC and invite users with legitimate addresses to try the tool. Then the benefit becomes mutual.

Second, we observed a relatively high rate of SPF validation at 85%, with full sender validation (i.e., SPF, DKIM, and DMARC) employed by over half of the domains we analyzed. The rate of SPF validation, as evidenced by DNS lookups, is lower than the 91% exhibited by popular mail providers, reported in previous work [7]. However, the fraction of domains that we observe performing full sender validation is an improvement over results shown in previous work, which indicated that only 23% of popular mail providers were performing DNS lookups related to SPF, DKIM, and DMARC validation [7]. We also observed that SPF validation happens real-time with mail delivery most of the time. However, some MTAs don't observe the limits imposed by SPF specification, which, combined with the real-time validation, might degrade performance.

Among our planned future work is to more fully analyze the results of each individual test policy that we designed. The collective set of behaviors might be used to classify and even fingerprint an SPF validator implementation, to learn how many distinct implementations are deployed.

9 CONCLUSION

In this paper, we have analyzed the deployment of SPF and DMARC, which were developed to counter fraudulent messages transmitted over SMTP. We analyzed two different data sets with three different experiments, to better understand the behaviors of sender-validating MTAs when they encounter a legitimate email message or one that is illegitimate. We developed a methodology to elicit SPF, DKIM, and DMARC validation activity in both of these scenarios and executed it at scale, assessing SPF and DMARC validation by MTAs for the over 40,000 domains. We observed SPF validation for a high percentage of domains, upwards of 85%. We also observed that just over half of the domains we observed have deployed SPF, DKIM, and DMARC. Another 24% have deployed SPF and DKIM without DMARC. Finally, we noted many behavioral nuances of sender-validating MTAs—some in direct violation of specification.

Our study shows that advances are being made in email security and sender validation. Yet there is still room for improvement, both in terms of the quantity and the quality of sender validation. We hope that our analysis of SPF, DKIM, and DMARC can serve as a baseline for improvement, for secure and reliable email communications.

ACKNOWLEDGMENTS

We gratefully acknowledge Michael Briggs and Kimball Leavitt for their helpful reviews of this paper prior to submission, Dallin Warne for acquiring the data used in our analysis, and Daniel Zappala, kc claffy, and Matthew Luckie for their helpful discussions of experiment design. We also acknowledge the anonymous CoNEXT 2021 reviewers and Gareth Tyson, our paper shepherd, for their kind and constructive feedback in preparing this paper for publication.

REFERENCES

[1] Amazon. Alexa top sites. https://aws.amazon.com/alexa-top-sites/.

[2] S. Bradner. RFC 2119: Key words for use in RFCs to Indicate Requirement Levels, March 1997.

[3] D. Crocker, T. Hansen, and M. Kucherawy. RFC 6376: DomainKeys Identified Mail (DKIM) Signatures, September 2011.

[4] C. Deccio, A. Hilton, M. Briggs, T. Avery, and R. Richardson. Behind Closed Doors: A Network Tale of Spoofing, Intrusion, and False DNS Security. In *Proceedings of the ACM Internet Measurement Conference*, IMC '20, page 65–77, New York, NY, USA, 2020. Association for Computing Machinery.

[5] Zakir Durumeric, David Adrian, Ariana Mirian, James Kasten, Elie Bursztein, Nicolas Lidzborski, Kurt Thomas, Vijay Eranti, Michael Bailey, and J. Alex Halderman. Neither snow nor rain nor MITM...: An empirical analysis of email delivery security. In *IMC '15 Internet Measurement Conference*, October 2015.

[6] Center for Applied Internet Data Analysis. Routeviews Prefix to AS mappings Dataset for IPv4 and IPv6, 2021.

[7] Ian D. Foster, Jon Larson, Max Masich, Alex C. Snoeren, Stefan Savage, and Kirill Levchenko. Security by any other name: On the effectiveness of provider based email security. In *CCS '15 Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, October 2015.

[8] Ivan Gojmerac, Patrick Zwickl, Gabriel Kovacs, and Christoph Steindl. Large-scale active measurements of dns entries related to e-mail system security. In *Communications (ICC), 2015 IEEE International Conference on*, June 2015.

[9] S. Kitterman. RFC 7208: Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1, April 2014.

[10] J. Klensin. RFC 5321: Simple Mail Transfer Protocol, October 2008.

[11] P. Mockapetris. RFC 1034: DOMAIN NAMES - CONCEPTS AND FACILITIES, November 1987.

[12] P. Mockapetris. RFC 1035: DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION, November 1987.

[13] Tatsuya Mori, Kazumichi Sato, Yousuke Takahashi, and Keisuke Ishibashi. How is e-mail sender authentication used and misused? In *CEAS '11 Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*, September 2011.

[14] Sarah Scheffler, Sean Smith, Yossi Gilad, and Sharon Goldberg. The unintended consequences of email spam prevention. In *Passive and Active Measurement*. Springer International Publishing, 2018.

[15] Kaiwen Shen, Chuhan Wang, Minglei Guo, Xiaofeng Zheng, Chaoyi Lu, Baojun Liu, Yuxuan Zhao, Shuang Hao, Haixin Duan, Qingfeng Pan, and Min Yang. Weak links in authentication chains: A large-scale analysis of email sender spoofing

attacks. In *30th USENIX Security Symposium (USENIX Security 21)*, pages 3201–3217. USENIX Association, August 2021.