SPF Beyond the Standard: Management and Operational Challenges in Practice and Practical Recommendations

Md. Ishtiaq Ashiq and Weitong Li, *Virginia Tech;* Tobias Fiebig, *Max-Planck-Institut für Informatik;* Taejoong Chung, *Virginia Tech*

https://www.usenix.org/conference/usenixsecurity24/presentation/ashiq

This paper is included in the Proceedings of the 33rd USENIX Security Symposium.

August 14–16, 2024 • Philadelphia, PA, USA

978-1-939133-44-1

Open access to the Proceedings of the 33rd USENIX Security Symposium is sponsored by USENIX.

SPF Beyond the Standard: Management and Operational Challenges in Practice and Practical Recommendations

Md. Ishtiaq Ashiq
Virginia Tech

Weitong Li
Virginia Tech

Tobias Fiebig
Max-Planck-Institut für Informatik

Taejoong Chung
Virginia Tech

Abstract

Since its inception in the 1970s, email has emerged as an irreplaceable medium for global communication. Despite its ubiquity, the system is plagued by security vulnerabilities, such as email spoofing. Among the various countermeasures, the Sender Policy Framework (SPF) remains a seminal and commonly deployed solution, working by specifying a list of authorized IP addresses for sending email.

While SPF might seem simple on the surface, the practical management of its records proves to be challenging; for example, although syntactical errors are uncommon (0.4%), evaluation-phase challenges are prevalent (7.7%), leading to potential disruptions in email delivery.

In our paper, we conduct a comprehensive study on the SPF extension, drawing from 17 months of weekly data snapshots that span 176 million domains across four top-level domains; we delve into the reasons behind such prevalent evaluation errors. Simultaneously, we undertake an ethical methodology to explore how SMTP servers *validate* SPF records and evaluate the effectiveness of widely-used software implementations. Our study unveils potential attack vectors that could be exploited for DNS amplification attacks or disrupt mail distribution; for instance, we demonstrate how an attacker could temporarily impede email reception by exploiting flaws in SPF validation mechanisms. We also conduct a qualitative study among email administrators to gain insights into the practical implementation and usage of SPF and SPF validators. Based on our findings, we provide recommendations designed to reconcile these discrepancies and bolster the SPF ecosystem's overall security.

1 Introduction

Email, often referred to as electronic correspondence, has been a dominant mode of global communication since its advent in the 1970s. Although there are now numerous other ways to communicate, the ubiquity of email continues to expand, offering cross-platform versatility.

However, despite its widespread use, email remains susceptible to security shortcomings. This is largely because its foundational protocol, the Simple Mail Transfer Protocol (SMTP), was designed in an era with different security norms [39] and thus lacks inherent features for verifying sender authenticity, making them vulnerable to a variety of security risks, such as email spoofing attacks [12, 14, 18].

To mitigate these threats, there have been a number of different email security protocols and extensions such as SPF [23], DKIM [10], DMARC [21], BIMI [9], etc. SPF, or Sender Policy Framework, is among the earliest solutions aimed at preventing email sender spoofing and was proposed in 2003, and first documented in 2006 [41]. Its operation is straightforward: the domain owner specifies a list of authorized IP addresses via DNS `TXT` records[1] so that a receiving mail server can verify if the source IP address of an incoming email is on that authorized list.

However, the apparent simplicity is misleading; managing `SPF` records can be intricate and misconfigured `SPF` records are quite common, exposing domains to phishing and spoofing, or limiting deliverability. An additional layer of complexity is added if `SPF` records depend on other `SPF` records not under the control of the domain owner, e.g., those belonging to hosting providers, as a domain owner must then also monitor those third-party records for changes.

Furthermore, over time, operational practice diverged from the original guidelines in RFC4408 [41] and the later updates in RFC7208 [23], increasing the potential for misconfigurations and vulnerabilities. For example, the `SPF` records of email hosting providers may point to multiple other `SPF` records to utilize their infrastructure, requiring the receiver to make more than 10 DNS queries, a practice that is actually prohibited by the RFC7208 [23]. Consequently, we observe a variety of implementation quirks in the wild. For example, our analysis reveals that certain email software comes configured with lenient DNS query limits by default, while some of their earlier versions lack query limitations

[1] For conciseness, we refer to these as `SPF` records in this paper.

altogether. Our experiment indicates that these outdated and vulnerable software might still be in existence and can be used as reflectors to launch stealthy DoS attacks.

In this paper, we present a comprehensive study of the SPF ecosystem to bridge the gap between operational practice and challenges in real-word SPF deployment. To examine the sender-side (i.e., the domain name owners), we employ 17 months of daily SPF record snapshots covering *all* second-level domains across the .com, .org, .net, and .se top-level domains. For assessing the receiver-side, we reduce the ethical impact associated with sending unsolicited emails by leveraging ambiguities in the RFC regarding *when* to initiate SPF record checks.

Our investigative approach is multi-faceted, aiming to unearth not just common misconfigurations but also to delve into the *why* behind them; this is achieved through a blend of quantitative analysis, based on our comprehensive scans, and qualitative insights, gathered through surveys of network operators. Our contributions are as follows:

- We conduct a large-scale longitudinal analysis of the SPF ecosystem both from sender side and recipient side. We find that the vast majority of SPF records are *syntactically* correct, with fewer than 0.4% errors. However, issues arise during the evaluation phase, where many records (6.5%) incur excessive DNS lookups, exceeding the RFC-defined limit of 10.
- We identify two primary sources of such misconfigurations: first, two popular hosting providers that incorporate too many domains in their SPF records, and second, oversights by SMTP administrators who neglect to remove obsolete SPF records after service migration.
- We outline an attack scenario that can temporarily prevent victims from receiving emails by causing their SPF validators to experience extended DNS resolution delays, all while the victims remain unaware due to the absence of emails in their inbox.
- We conduct a survey among mail operators to understand the SPF landscape in practice. Our survey reveals that even large email providers managing over 1,000 accounts often deviate from required limits in RFCs [23], highlighting a disconnect between outdated standards and current operational needs.
- Finally, we offer a set of community-focused recommendations aimed at mitigating potential abuse and bridging the gap between nearly two-decade-old standards and contemporary SPF usage.

On a constructive note, our study uncovers straightforward avenues for improving SPF validation to achieve its intended security goals. In the interest of collaborative research and actionable insights, we make our entire analysis code and dataset publicly available at

https://spf-measurement.github.io

2 Background

2.1 Simple Mail Transfer Protocol (SMTP)

Simple Mail Transfer Protocol (SMTP) is the protocol employed for email exchange over the Internet. The email process commences with the user composing an email in their Mail User Agent (MUA). This email is then transmitted to the sender's Mail Transfer Agent (MTA), either via SMTP or HTTP. The sending MTA subsequently identifies the recipient MTA's address through the Domain Name System (DNS), establishes a Transmission Control Protocol (TCP) session with the target host, and transfers the email using SMTP. The process involves the execution of several SMTP commands, including:

- HELO/EHLO commences communication with the server. They serve as greeting messages, wherein the sending server usually declares its name. The EHLO command is an extended version, prompting servers to notify the client about the extended features they support.
- MAIL FROM specifies the sender's email address, thereby informing the server about the origin of the email. It also sets the return path for delivery status notifications.
- RCPT TO indicates the recipient's email address. The server acknowledges each valid recipient with a positive response.
- DATA describes the actual email content. This command is followed by the email message, including headers and body. In the email message header, the sender can also specify one or more *From* address(es), which is generally shown as the from address in the display of the recipient's MUA [32]. For readability, we refer to this *From*–commonly know as the RFC5322.From–as Header From.

Subsequently, the Mail Delivery Agent (MDA) delivers the email to the recipient using HTTP, IMAP, or POP3 protocols [22].

2.2 Sender Authentication

SMTP has no built-in security mechanisms; theoretically, an attacker can manipulate the address in the MAIL command and in the Header From field to spoof the sender domain [21, 37]. To mitigate these attacks, various security extensions have been proposed.

2.2.1 Sender Policy Framework (SPF)

With SPF, domain owners can publish a policy with which a receiving mail server can assess whether a sending IPv4 or

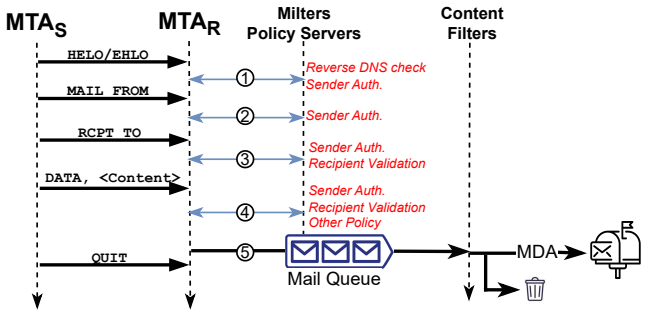

Figure 1: An overview of SPF validation flow at various stages of the SMTP transaction. Upon receipt of the `HELO/EHLO` command, either the MTA natively or through milters can initiate a reverse DNS lookup for the incoming IP address. The standard [23] requires that SPF validation must be performed on the `MAIL FROM` identity, but also suggests doing it on the `HELO/EHLO` as well. So, after the `MAIL` command, sender authentication can be executed at any point during the SMTP connection by milters or policy servers. Notably, SPF validation can even occur post-queue in the content filter stage (e.g., SpamAssassin [6]).

IPv6 address may deliver a message for a specific domain. This is done via a DNS `TXT` record at the sending domain. Policies are described in a `TXT` record (the previously used SPF DNS resource record (RR) has been discontinued [23]) using `mechanisms` with `qualifiers` and `modifiers` and SPF verifiers check policy of `HELO` identity (recommended) and `MAIL FROM` identity (must) against the connecting IP address. Furthermore, SPF supports the use of `macros` expanded based on characteristics of the message.

Mechanisms Mechanisms serve as the main method of defining policy; In total, eight mechanisms are defined: `A`, `ip4`, `ip6`, `mx`, `ptr` (discouraged as per RFC [23]), `exists`, `include`, and `all`. `ip4` and `ip6` allow specifying permitted sender addresses or networks for a domain. For example, the following configuration authorizes emails only from the IP address 10.0.0.1, while rejecting all others.

```
example.com   TXT   "v=spf1 ip4:10.0.0.1 -all"
```

In addition, `mx` allows the IP addresses of a domains' `MX` records, `ptr` allows all clients whose reverse DNS name is in the domain (easily forged and therefore discouraged), `a` if a given FQDN resolves to a connecting IP and `exists` if a given domain exists, which is mostly useful with `macros`. Finally, the `all` record sets a default policy for all addresses not matched by earlier mechanisms.

In this paper, we primarily focus on the `include` mechanism. This mechanism allows referring to `SPF` records of another FQDN. This is especially useful when domain name owners outsource their email service to a third-party provider, e.g., Google. A domain owner can then simply include `_spf.google.com` managed by Google to ensure `SPF` is appropriately configured.

```
example.com   TXT   "v=spf1 include:_spf.google.com -all"
```

This mechanism is also useful if domain owners require other third-party services for specialized tasks; for instance, a domain owner using QQ.com for email hosting may also wish to utilize a bulk email service like Mailchimp or SendGrid. Hence, email hosting providers like Google Workspace [5] or QQ.com [19] publish their SPF records to be included by domain owners through the `include` mechanism, and often explicitly document that users should do this.

Qualifiers Qualifiers specify the disposition if a mechanism matches. Absence of a qualifier defaults to `+ (pass)`.

- `+ (pass)`: Positive disposition.
- `- (fail)`: Negative disposition. Commonly used with the `all` mechanism to reject all senders not explicitly listed.
- `? (neutral)`: No disposition.
- `~(softfail)`: a negative disposition but not a strong one (i.e., hinting that sender is probably not authorized).

Modifiers Currently, the `exp=` and the `redirect=` modifier are specified in RFC4408 [41]. `exp=` should point to an FQDN whose TXT record explains the reason for a negative disposition. `redirect=` works similar to `include`, even though subtly different. As `redirect=` is rarely used in practice (apart from `gmail.com`), we focus in `include` in this work.

SPF Errors and Security Considerations SPF relies on DNS records for its operation. Evaluation can fail if the `SPF` records are inaccessible or malformed. Generally, two types of errors can arise during the evaluation of an `SPF` record, often resulting in email rejection:

- `temperrors` occur when the SPF validator encounters a transient issue, such as a DNS timeout. The validator may retry the operation later.
- `permerrors` occur when the SPF validator successfully retrieves a DNS record but fails to interpret it correctly.

A classic example that triggers a *permerror* is when the number of required DNS queries to evaluate an SPF record exceeds the standard limit of 10. This usually happens when multiple mechanisms refer to other DNS records, such as through `include` mechanisms. The standard imposes this limit to alleviate undue stress on DNS infrastructure.[2] In situations where the referenced SPF record does not resolve due to an empty DNS response or errors such as `NXDOMAIN`, this condition is termed a *void lookup*. The RFC states that the number of void lookups `should` be limited to two.

[2] It is worth highlighting that this is distinct from `CNAME` records, which map one domain name to another in DNS. `CNAME` expansion is usually the responsibility of DNS resolvers, while `include` mechanisms in SPF are expanded by the SPF validator.

TLD	Domains w/ MX	Domains with MX records		
		SPF	SPF include	SPF include with Ext.
.com	75.8 M	48 M (63.2%)	28.9 M (60.3%)	28.8 M (99.6%)
.net	6.5 M	3.5 M (53.8%)	2 M (57.1%)	1.98 M (99.6%)
.org	5.8 M	3.2 M (55.2%)	1.9 M (59.3%)	1.9 M (99.7%)
.se	845 K	439 K (52%)	365 K (83.1%)	365 K (99.7%)

Table 1: Overview of the datasets. The number and percentage of the domains that have SPF records are as-of March 27, 2023.

SPF Validator Ecosystem SPF validation occurs either natively within the recipient MTA using SPF libraries (e.g., libspf2 [45]) or via external extensions, which can fall into three categories based on the timing of the SPF check.

- Before Queue Milter (e.g., milter-greylist [46]) or Milter, in short, is a flexible framework for pre-queue email filtering and modification. Milters can reject emails *during the SMTP transaction*.
- Policy Server (e.g., iRedAPD [44]) makes routing decisions based on predefined policies, such as rejecting an email if SPF fails.
- Post-Queue Content Filter (e.g., SpamAssassin [6]) operates after the MTA queues the mail, focusing on content-based attributes like message body and attachments.

We explain how these extensions can make SPF queries in Figure 1.

2.2.2 Other Extensions for Sender Authentication

Although not the primary focus of this paper, there are additional security extensions designed to support sender authentication. Two notable examples are briefly explained below:

- `DomainKeys Identified Mail (DKIM)` enables an email receiver to confirm the integrity of a received message by embedding a digital signature within the email header. The receiver can then fetch the corresponding public key from the DNS to verify this signature.
- `Domain-based Message Authentication Reporting and Conformance (DMARC)` allows domain owners to publish a policy within the DNS that dictates certain actions for email receivers to follow when SPF and DKIM validation fails.

In summary, DKIM serves to provide both the integrity of the message and its authenticity, while DMARC furnishes a policy framework for email receivers to act upon in cases where SPF and DKIM validation fails.

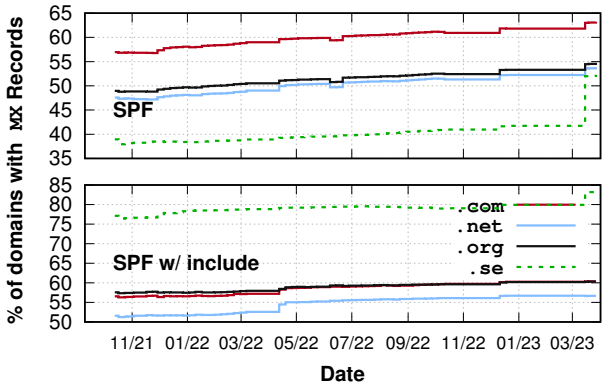

Figure 2: The percentage of domains with `SPF` records is increasing; the spike on March 15, 2023 for `.se` domains is because around 100K domains using loopia.se as their email hosting provider added an `SPF` record to their domain.

3 SPF for Outbound Emails

We first focus on domain name owners who *publish* `SPF` records and how they are deployed and managed.

3.1 Datasets

To scan a vast number of registered domains, we conduct DNS scans across four top-level domains (TLDs): the most widely used three generic TLDs (gTLDs) `.com`, `.org`, and `.net` gTLDs and a country-code TLD (ccTLD), `.se`, recognized for its aggressive deployment for email security protocols such as DANE [24] or DMARC [4]. Initially, we gather daily zone files from the respective registries for each of the four TLDs, namely Verisign for `.com` and `.net`, Public Internet Registry for `.org`, and Internetstiftelsen for `.se`. This allows us to acquire the Name Server (`NS`) information for *all second-level domains* (SLDs). For each of SLDs, we fetch `SPF` records for each domain. In total, our snapshots span 17 months of 176 M domains from October 13th, 2021 to March 27th, 2023, which is summarized in Table 1.

3.2 SPF Record Prevalence

We now first examine how `SPF` records are deployed by domain name owners. Figure 2 shows the percentage of domains that have `SPF` records during our measurement period. First of all, we see a stable increasing trend across all four TLDs; for example, the deployment rate in `.com` domains increased from 56.9% to 63.2% during our measurement period. Similarly, we also observe that a prevalent usage of `include`, which indicates that the domain name owners import other SPF policy for their domains; for example, in the case of `.se`, we find that 83% of domains with `MX` records use external SPF policies. We now examine whether the policy is from other parties; since it is not straightforward

TLD	Percentage Changes of Mechanism (Δ)							
	a	mx	include	ip4	ip6	ptr	exists	all
.com	-3.4	-3.8	+3.9	-2.0	-0.3	-0.8	-0.0	+0.2
.net	-2.9	-3.1	+5.1	-3.1	+0.0	-0.7	-0.0	+0.5
.org	-3.4	-3.9	+2.7	-1.8	-0.3	-1.1	-0.0	+0.3
.se	-9.1	-9.1	+6.2	-7.3	-0.9	-0.2	-0.0	+0.3

Table 2: Percentage change (%) in the usage of each mechanism between our initial snapshot on October 13, 2021, and our final snapshot on March 27, 2023; only `include` and `all` mechanisms have increased across the TLDs, as indicated by a green background.

to compare whether two domains (base domain and the domain in `include`) are managed by the same entity[3], thus we compare whether the domains share same organizational domain [21] or not.

Remarkably, across all TLDs, we find that 99.8% utilize included `SPF` records where at least one of the included domains has a different organizational domain than the base domain. This strongly suggests that the majority are employing the `include` mechanism to import SPF policies from external sources; Such a trend is further substantiated when examining the changes in mechanism usage over our measurement period; Table 2 reveals the evolution in the percentage distribution of each mechanism within SPF records. Notably, we observe that only the `include` and `all` mechanisms have seen an increase in usage across the TLDs, further emphasizing the dependence on external policies and the general move towards stricter email authentication. This phenomenon can be attributed to the prevalent use of third-party mail hosting providers. These providers frequently recommend that their clients employ the `include` mechanism to reference their `SPF` records, which may encompass a broad array of authorized IP addresses.

3.3 SPF Records Management

The management of `SPF` records is crucial due to their role in authorizing the SMTP server of the sender domain. When checking the incoming IP address against SPF records, receiving SMTP servers can encounter a variety of errors, which largely fall into two categories: syntax and evaluation errors.

Syntax Error: `SPF` records that do not follow the syntax of the standard [23] can be instantly detected during the parsing process. The syntax of the record is validated first before the evaluation, thus if there are any syntax errors anywhere in the record, it is determined as `permerror`, terminating the SPF evaluation process immediately without further evaluation.

[3]One possible approach is to leverage WHOIS record, but the infrastructure is heavily rate-limited and notoriously inconsistent [25] and many domains are registered through privacy-preserving services, making it challenging to identify the actual owner [11].

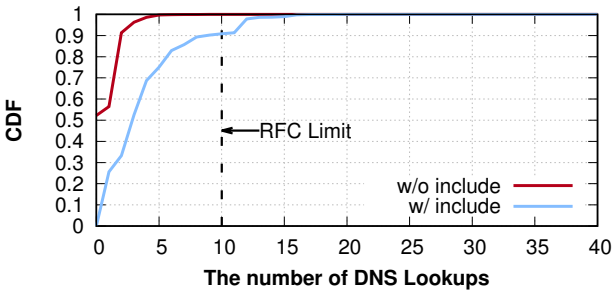

Figure 3: More than 9.2% of SPF records with `include` requires more than 10 DNS look ups. Note that the *x* axis extends to 175!

Evaluation Error: `SPF` records have the ability to reference other DNS records, such as `A` records, through mechanisms. This means that even if an `SPF` record appears valid, it can be deemed invalid during the evaluation process if, for instance, the domain name in the `include` or `a` does not exist. Furthermore, even if all mechanisms within an `SPF` record are successfully evaluated, the validation can still fail if it requires a large number of DNS lookups during the evaluation process since it can impose a significant performance overhead on the receiving server. To address this, the standard [23] mandates issuing a `permerror` when the total number of DNS lookups exceeds 10. One challenge is that domain owners may not promptly detect these errors, especially *when the `SPF` records referenced by mechanisms (e.g., `include`) are managed by other domains*; for example, if `SPF` records included from other domains cause extra look ups, the total number of DNS lookups may exceed 10, resulting in a `Permerror`.

3.3.1 Results

We use our latest scan and focus on 55 M domains that have `SPF` records to assess the accuracy of these `SPF` records by fully resolving all mechanisms contained within them. Table 3 shows the results. First of all, upon examining the syntax of `SPF` records, we find that they *look* well-managed; for example, only 0.003% of SPF records do miss some values for specific `a` or `mx` mechanisms.

However, when focusing on evaluation errors, more prevalent issues emerge. For instance, we find 690,554 domains (1.2%) with incorrectly configured mechanisms that are unresolvable. Even more concerning is the issue of excessive DNS lookups during SPF evaluation; specifically, 3,584,014 (6.5%) domains require more than 10 DNS lookups for evaluation. This is particularly significant because RFC7208 [23] mandates generating a `permerror` outcome in such cases, potentially disrupting email delivery.

Given that most domains in the `include` mechanisms originate from sources external to the base domain (Table 1), mainly due to hosting providers, this widespread occurrence is expected; for example, `SPF` records using `include` mechanisms require a considerably higher number of DNS

Type	Reasons	# of domains	Example
Syntax	Missing Values: ip4 and ip6	7,827 (0.01%)	"v=spf1 ip4: -all"
	Missing Values: a and mx	2,127 (0.003%)	"v=spf1 a: -all"
	Invalid Value	23,147 (0.04%)	"v=spf1 ip4:1.2.3 -all"
	Unknown Mechanism	53,466 (0.1%)	"v=spf1 mx -al"
	Recursive Value	55,907 (0.1%)	a.com TXT "v=spf1 include:a.com -all"
	Multiple SPF records	82,606 (0.15%)	a.com TXT "v=spf1 a -all" a.com TXT "v=spf1 mx -all"
Evaluation	Missing Records: TXT record	85,140 (0.16%)	a.com TXT "v=spf1 include:b.com -all" % *b.com does not exist.*
	Missing Records: MX record	324,524 (0.6%)	a.com TXT "v=spf1 mx:b.com -all" % *MX record of b.com does not exist.*
	Missing Records: A record	280,890 (0.5%)	a.com TXT "v=spf1 a -all" % *A record does not exist.*
	Too many DNS lookups	3,548,014 (6.5%)	a.com TXT "v=spf include:a1.com -all" % *a1.com includes a2.com, etc.*

Table 3: The most popular reasons and their examples of SPF record errors are shown.

lookups compared to those without such mechanisms. As depicted in Figure 3, 3,564,056 (9.2%) of these SPF records, with at least one include, require over 10 lookups.

To investigate if this is a result of popular hosting providers' misconfigurations, we evaluate each domain listed in a include mechanisms. We count their occurrences in SPF records; Figure 4 depicts that 50% of SPF records reference six other SPF records from well-known hosting providers[4]. Next, we also do so, but focusing on those that require more than 10 DNS lookups; interestingly, 2,983,114 (83.7%) of them reference just two SPF records[5]. These are associated with Hostgator and Bluehost, each of their customers requiring 10 and 13 lookups, respectively. Still, 616,581 (17.3%) of the domains that demand more than 10 lookups do not reference these two major hosting providers. We delve into this subset of domains in the following section.

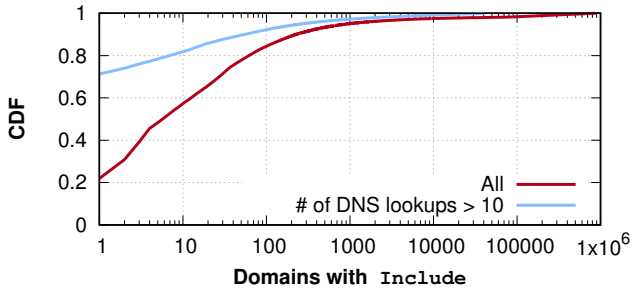

Figure 4: As of March 27th, 2023, only six SPF records appear in the include mechanisms for 50% of all SPF records. Remarkably, just two SPF records—managed by two specific hosting providers—account for 83.7% of the SPF records that require more than 10 DNS lookups for evaluation.

4 Superfluous SPF records

We have discovered that around 616K domains require more than 10 DNS lookups but they do not include either Hostgator or Bluehost. This raises an immediate question: why do these domains require many lookups? In this section, our goal is to determine the essentiality of all the IP addresses or prefixes specified as approved senders in their SPF records, and whether any redundant mechanisms are in place. For instance, if an email server transitions to a different host or opts for an alternative hosting provider, the administrator must update their DNS entries accordingly; for example, in 2023, Microsoft hotmail accounts encountered email outages due to their SPF records not being updated despite changes in their actual email server IP addresses [1]. Therefore, the prudent approach would involve retaining their historical include mechanism (i.e., those that might be considered superfluous) while simultaneously introducing new SPF records to ensure proper functionality.

[4] outlook.com (11.6%), google.com (10.3%), secureserver.net (9.0%), registrar-servers.com (8.4%), websitewelcome.com (6.3%), mailchannels.net (2.7%)

[5] websitewelcome.com and bluehost.com

The domains with superfluous SPF records does not seem to cause any operational problems; however, it may amplify the required DNS lookups causing delivery issue. In this section, we aim to find the domains having superfluous SPF records that persist despite alterations to the sending MTAs.

4.1 Methodology

To assess whether an SPF record contains superfluous entries, we first need to infer the ideal minimally covering SPF record for a domain. While, for a small set of major mail providers, these are published on their websites and can be assumed to be up-to-date, this is not the case for the average domain. Hence, while we can–and do–collect the documentation pages of large email providers to get ground-truth on their ideal SPF policy, we face a gap when it comes to smaller setups.

To bridge this gap, we work on the receiving mail servers communicated via MX records that are also responsible for sending emails for a domain. Even though this assumption is especially not true for large providers, we found it to be a common case also, e.g., signified by 350K domains in March 27, 2023 having mx as the only mechanism in their SPF policy. Nevertheless, such domains may also leverage external senders *in addition* to in-house sending, e.g., for newsletters.

Rank	MX Records	Inferred SPF Records	# of Domains	Prob.	SPF Updated	# of Flatten Domains	# of Domains w/ Stale SPF
1	*.google.com	_spf.google.com	4,357,014	95.9%	✗	2,836	–
2	*.outlook.com	spf.protection.outlook.com	3,974,407	93.4%	✓	1,324	182 (13.7%)
3	*.registrar-servers.com	spf.efwd.registrar-servers.com	3,782,396	98.4%	✗	201	–
4	*.googlemail.com	_spf.google.com	1,226,471	99.0%	✗	2,836	–
5	*.ovh.com	mx.ovh.com	571,484	95.2%	✓	48	10 (20.8%)
6	*.jellyfish.systems	spf.web-hosting.com	509,629	97.7%	✓	86	51 (59.3%)
7	*.hostinger.com	_spf.mail.hostinger.com	467,240	93.9%	✓	167	43 (26.3%)
8	*.qq.com	spf.mail.qq.com	465,717	97.9%	✓	151	44 (29.1%)
9	*.titan.email	spf.titan.email	427,060	99.2%	✓	38	9 (23.6%)
10	*.gandi.net	_mailcust.gandi.net	389,624	97.0%	✗	22	–

Table 4: The table lists the top 10 popular MX records, inferred SPF records, and their occurrence probabilities. Additionally, it indicates (1) if the IP addresses in the SPF records have been updated, (2) the number of domains that have flattened their SPF records, and (3) the domains with stale SPF records despite updates from their hosting provider.

4.1.1 Inferring SPF Records from MX Records

To infer the minimal SPF record of a domain, we first assume that domains utilizing third-party hosting services should have at least one include mechanism. Now, we compute the *likelihood* that a domain with a specific MX record, mx_x also includes an SPF record, spf_y in their include mechanism. To be precise, we follow these steps:

1. For a given domain name, we generate a list of tuples that consist of all possible combinations of its MX records and the domains in their include mechanisms; for example, if a domain name has a two MX records and three include mechanisms, we generate a total of 6 tuples.
2. We repeat this process for all domains and count the occurrence of domains for each tuple (mx_m, spf_n), which we define as $d(mx_m, spf_n)$.
3. For each tuple, we calculate a conditional probability, $P(spf_k|mx_m) = \frac{d(mx_m,spf_k)}{\sum_{i=1}^{n} d(mx_m,spf_i)}$ where $1 \leq k \leq n$ and n is the number of observed SPF records with mx_m.

Using this approach on our latest snapshot, we obtain 31,876,212 tuples along with their conditional probability. To evaluate our metric, we manually survey the top 10 email hosting providers to obtain the mapping of their MX records and SPF records for their customers.

As shown in Table 4, we find that all of their probabilities are above 93%, each of which aligns with our manual survey. It is worth noting that attaining an absolute 100% probability is practically unfeasible; even in the case of a domain featuring a highly prevalent MX record (e.g., smtp.google.com), the inclusion of superfluous SPF records can diminish the likelihood of achieving a perfect match with _spf.google.com to a value lower than 100%.

Since our objective is to establish a mapping of MX records to their corresponding SPF records for popular hosting providers, we consider only those cases where the probability of SPF record given MX record, $P(spf_k|mx_m)$ is above 90%, and the number of domains associated with such records is more than 100, which leaves us 3,141 tuples of MX records and their corresponding SPF records; clearly, this threshold introduces a trade-off between the size of the tuple list and accuracy. We delve into this topic in more detail in 8.2.

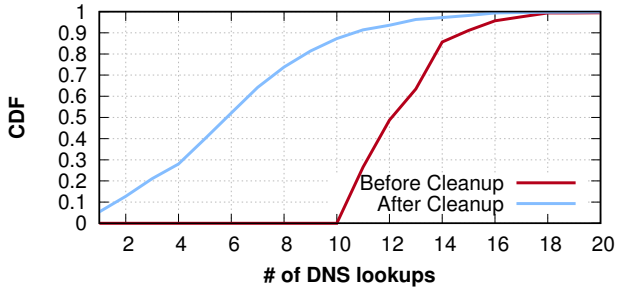

Figure 5: Removing superfluous SPF records can effectively reduce the number of required DNS lookups.

4.2 Domains with Superfluous SPF Records

Next, we identify the domains that have superfluous SPF records. To accurately pinpoint these redundant SPF records, we first only consider the domains for which we know all of the associated MX records for the SPF records in their include mechanisms using our compiled datasets, which leaves us with a total of 24,832 domains. Next, we delve into quantifying the number of domains that possess *superfluous* SPF records, ones that are not tied to any MX records; to be precise, we label an SPF record as superfluous if its corresponding MX records are not present in our dataset.

Remarkably, our analysis reveals that 20,124 (81.0%) of these domains are burdened with superfluous SPF records. Such redundant records could be problematic, raising the likelihood of mail delivery complications due to these extraneous SPF entries. The resolution of this concern lies in the removal of these superfluous SPF records; as shown in Figure 5, we can observe that 17,554 (87.2%) of these domains can effectively streamline their lookup within 10,

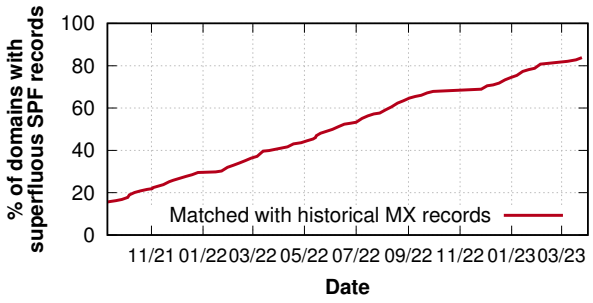

Figure 6: The percentage of domains with superfluous SPF records from our latest snapshot that match with historical MX records.

thereby alleviating the potential mail delivery problem.

4.3 Why Superfluous Records Exist

As elaborated in §4.1, a potential drawback of our methodology arises when sending MTAs are not represented in the MX records, leading to false-positives in our identification of superfluous SPF records.

We now investigate if these superfluous SPF records have any correlation with historical or outdated MX records that a domain might have previously used; if such a connection exists, it could strongly suggest that the presence of these superfluous records may be due to domain owners neglecting to update their SPF configurations in a timely manner. This scenario could arise when domains transition their email services from one hosting provider to another, inadvertently overlooking the removal of the former SPF records.

We examine if the MX records linked to these superfluous SPF records can be traced among the entire history of MX records used by the domain; in essence, we ascertain whether these superfluous SPF records can indeed be aligned with any of the domain's previous (and now outdated) MX records.

For the 20,124 domains that have superfluous SPF records in our latest snapshot, we found that 18,936 (94.1%) domains *have changed their MX records during our measurement period.* Now, for each historical snapshot, we examine if we can find any MX records that correspond to the superfluous SPF records, which is shown in Figure 6; we find that a substantial majority of these superfluous SPF records 16,884 (83.9%) stem from obsolete SPF entries that have not been removed following the migration of mail servers. Interestingly, for 15.3% of these superfluous SPF records, we can find their corresponding MX records in our initial snapshot, suggesting these redundant entries have persisted for at least 15 months.

4.4 Case study: SPF Flattening

To mitigate the recursive nature of SPF records, the concept of *SPF flattening* has been proposed [42]. This technique involves fully evaluating the SPF records and explicitly listing all the assessed IP addresses or prefixes, thereby eliminating the need for include mechanisms. While SPF flattening is effective in reducing the recursion introduced by include mechanisms, it comes with its own set of challenges. Specifically, this approach becomes less reliable when IP addresses within the nested SPF records undergo changes. If these flattened addresses are not updated in a timely manner to reflect such modifications, email delivery may face disruptions.

To evaluate this possible issue, we employ a two-step approach: first, for each domain, we examine if the set of allowed IP addresses can fully cover all the permitted IP addresses specified in the SPF record tied to its MX record. This is done by comparing with our pre-established groundtruth dataset. Next, we monitor for any changes in IP addresses within the SPF records of providers throughout our measurement period.

From the steps, we observe that six providers that have implemented such updates, which suggest that the domains using these hosting providers and relying on flattened SPF records could potentially face complications in email delivery; the results are presented in Table 4.

Interestingly, we discover that numerous domains which have transitioned to flattened domain structures have not yet updated their SPF records, rendering them stale; for instance, among the domains using JellyFish and adopted flattened SPF records, 59.3% of them have neglected to revise their SPF records, despite the fact that the SPF records of the hosting provider itself have been updated. This discrepancy could potentially lead to email rejections due to SPF validation failures.

5 SPF for Inbound Emails

In the previous section, we have found that 6.5% of domains serve SPF records that require more than 10 DNS lookups, which may cause the email receivers to reject incoming emails from the domains since it raises permerror; however, such a high percentage might indicate that *some receiving MTAs may allow more than that.*

Now, we shift our focus to the SMTP servers that check SPF records for incoming emails. In this section, we primarily focus on the DNS lookup limit specified in the standard for: (1) total DNS lookup and (2) void lookup.

5.1 Methodology

At first glance, it might seem straightforward to measure how MTAs impose limits on DNS lookups of SPF records; we could set up our own MTAs and serve an SPF record that incurs many DNS lookups by utilizing nested include mechanisms. Subsequently, we could send emails to selected MTAs and tally the number of incoming DNS queries; this measurement methodology has been employed in prior

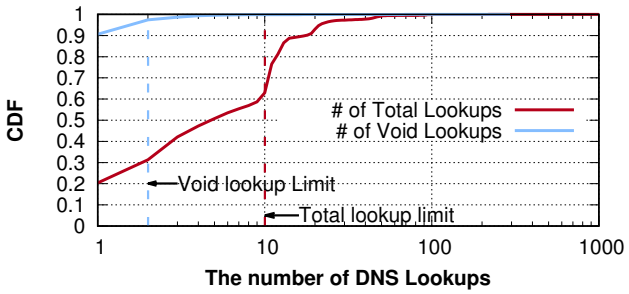

Figure 7: CDF of the number of DNS requests that our DNS authoritative server receives; 40% servers violate the total lookup limit recommendation in practice. Note that x axis extends to 25,117![6]

studies to gauge the deployment of SPF checks for incoming emails [14, 36]. However, we argue that this approach could elicit ethical dilemmas especially when the objective is to measure many MTAs; taking explicit permission from many recipients is practically not feasible, which may classify our action as unsolicited sending, thereby potentially causing inconvenience or even harm. Instead, we leverage an RFC ambiguity to trigger DNS queries from MTAs.

Alternative Approach: We note that RFC7208 [23] does not specify the SMTP command that should trigger the receiving MTAs to perform the resolution of the `SPF` record; given that the SPF check is performed on the domain name in the `MAIL FROM` header, we may expect that some SMTP servers may send DNS queries for `SPF` records once it receives the `MAIL FROM` command from the sender; for example, we find that MTA software such as `postfix` with the `iRedAPD` extension sends DNS queries for the `SPF` record after `RCPT TO` command. This implies that it may be possible to trigger DNS queries for SPF records *without sending emails* to the recipient's inbox if it looks up `SPF` records as soon as it receives the domain name; theoretically, as the `MAIL FROM` header contains the sender's domain name, it may be enough for them to lookup `SPF` records; thus we only send the `HELO`, `MAIL FROM`, and `RCPT TO` commands to determine if it triggers DNS queries, and then terminate our SMTP connection *without* executing the `DATA` command to avoid transferring email data.

However, we acknowledge that our methodology could potentially identify fewer vulnerable receiving MTAs because most email spam filters or SPF milters perform an `SPF` record check once the mail data is transferred. Nevertheless, we believe that this approach is appropriate for understanding the vulnerability empirically without adversely affecting the MTAs in the wild. Furthermore, we obtain a set of unique `MX` records by scanning zone files, and then resolve their IP addresses while removing those that share the same IP addresses to limit the number of MX servers.

Name	Type	Ver.	DNS # of Lookups N_{lim}	DNS # of Lookups V_{lim}	DNS Timeout (s) t_{lim}	DNS Timeout (s) T_{lim}
libspf2 [45]	Lib	latest	10	1	5	∞
Mail::SPF::Query [27]	Lib	< 1.8	∞	1	15	∞
Mail::SPF::Query [27]	Lib	> 1.8	10	1	15	∞
pyspf [49]	Lib	latest	10	2	2	20
milter-greylist [46]	M	latest	10	10	5	∞
spfmilter [50]	M	latest	10	2	5	20
mtpolicyd [47]	PS	latest	10	2	30	∞
policyd-spf [48]	PS	latest	20	1	5	20
iRedAPD [44]	PS	< 5.1	∞	∞	3	∞
iRedAPD [44]	PS	≥ 5.1	20	20	3	∞
SpamAssassin [6]	CF	latest	20	1	3	5
RSpamD [3]	CF	latest	30	30	1	∞

Table 5: The lists 10 popular SPF libraries (`Lib`), milters (`M`), policy servers (`PS`), and content filters (`CF`). Here, N_{lim} and V_{lim} represent the limits on DNS and void lookups, respectively; for example, Mail::SPF::Query and iRedAPD have no N_{lim}, resulting in endless DNS requests (highlighted in red). t_{lim} and T_{lim} indicate DNS timeout values for individual and total SPF evaluations, respectively. When parallel SPF validation is not supported, the SMTP server is vulnerable to DoS attack for inbound emails (highlighted in yellow).

Experiment Process:

We purchase a domain name, `a.com`, and configure our customized DNS authoritative server and SMTP server using Postfix [30]. From the latest snapshot of four zone files, we examine the `MX` records and their corresponding IP addresses to create a unique set of MTAs, yielding 1,886,825 `MX` records in total and then proceed as follows:

1. For each `MX` record, mx_a, we generate two unique subdomains: $mx_a.t.a.com$ and $mx_a.v.a.com$ to serve two types of `SPF` records to evaluate their DNS lookup and void lookup limits. To assess the DNS lookup limit, we serve `SPF` records featuring 50 `include` mechanisms, which incurs 51 DNS lookups. For the void lookup limit, the `SPF` record includes 50 `include` mechanisms, 36 of which lead to void lookups; we provide an explanation of the choice of parameters in §8.2.
2. We establish two SMTP connections to the target MX server: one for gauging the total lookup limit (Exp. #1) and another for probing the void lookup limit (Exp. #2). In each connection, we issue a `HELO` command featuring the unique subdomain.

[6] Along with this SMTP server, we found that two additional SMTP servers sent duplicate queries over 1,000 times; we attempted to contact them using common administrator email addresses (e.g., `postmaster@`) and the admin name in the `rname` field of the `SOA` record, but we were unable to reach them. This could be due to SPF validators that store `SPF` records in their database and periodically update the `SPF` records for security purposes (e.g., iRedAPD converts SPF records of specified domains to IP addresses on an hourly basis).

3. Subsequently, we transmit our `MAIL FROM` address with `noreply` as the username.
4. Next, we send a `RCPT TO` command with `postmaster` as the username.
5. We pause for 3 seconds to allow sufficient time for the `SPF` queries to reach our DNS authoritative servers, after which we terminate the SMTP connection.
6. All incoming DNS queries are logged for analysis.

We conducted these experiments from Feburary 28, 2024 to March 16, 2024. Along with this, we investigate 10 popular SPF libraries, milters, policy servers, and content filters, as detailed in Table 5, through source code analysis and testing.

5.1.1 Ethical Considerations

Our approach does not involve sending unsolicited emails; however, we intend to discuss the potential implications for email service providers explicitly before sharing our findings. Firstly, we correctly set up reverse DNS record and remove duplicates from the `MX` records and their resolved IP addresses to ensure that we communicate with no more than one server.

We also took extra care to ensure our SMTP scanner is fully compliant with the RFCs; for instance, we have programmed our scanner to disconnect following the `QUIT` command, mirroring the behavior of a well-behaved SMTP client. This attention to detail is crucial; failing to comply with this standard, such as by disconnecting without the proper command, could unnecessarily alert system administrators. It is worth noting that throughout our scanning period, we did not receive any automated abuse reports or personalized feedback from operators.

We acknowledge that we have not asked for any explicit consent to connect to the IP addresses mapped to `MX` records, which may violate one of the four principles of the Menlo Report [15], "Respect for Persons"; unfortunately, it is practically impossible for us to obtain an informed consent from all email operators and also, emailing them to ask for approval will be considered as unsolicited email. Thus, we decided not to ask for consent. However, it is important to note that the inability to obtain informed consent does not imply a disregard for respect towards individuals [35]. To ensure we follow the guidelines by the Menlo report [15], we did not send any emails and tried our best not to contact multiple email operators by deduplicating the adddresses. We believe in the value of publishing our findings, considering the potential benefits and the preventative aspect of our research in mitigating future harm by informing the community, thereby justifying our approach amidst ethical concerns.

5.2 Results

With our methodology, we successfully connect to 1.2M (64%) SMTP servers out of 1.89M ones in total and find that 81,843 (6.8% of 1.2M) initiate `SPF` queries prior to issuing the `DATA` command; this suggests that most SMTP servers opt for SPF validation after the `DATA` command, likely to exploit the policy flexibility available at this stage.

Now, we focus on the number of DNS queries we receive. Figure 7 shows the results. First, we find that 903 servers (1.1%) send more than 51 queries, but they do so by *issuing duplicated `SPF` queries*. This could be due to either (1) SPF validation taking place from multiple extensions, (2) the involvement of multiple DNS resolvers executing identical queries, or (3) the use of SPF validators that lack DNS lookup limits as elaborated in §2. We will further explore this in §6. Among them, we find that 195 SMTP servers queried for all 50 `include` in our `SPF` record; this strongly suggests that some SMTP servers use `SPF` validators that *do not have any total lookup limit*. Out of these 195 servers, 165 of them maintain a void lookup limit as they aborted `SPF` resolution right after querying the set number of domains that incur `NXDOMAIN` responses in Exp. #2; the rest (30) requested all 36 domains that result in void lookups, which indicates that these servers could potentially serve as reflectors for launching DNS queries against a targeted victim's authoritative server. Interestingly, even for those who made < 51 queries in Exp. # 1, we find that 39 SMTP servers made all 36 void lookups, which indicates that they do not have a void lookup limit even though they have a total lookup limit.

These findings corroborate the hypothesis that a substantial number of SMTP servers operate with lax or even unlimited lookup limits. As validated in Table 5, both `Mail::SPF::Query` and `iRedAPD` lack a DNS lookup cap. Given that the iRedAPD's patch was in August 2022, this signals potential vulnerabilities in numerous mail servers.

5.3 Threat Model

Each MTA establishes its own timeout limits for SPF validation. If the corresponding SPF validator does not support parallel processing of incoming SMTP connections, attackers can exploit this timeout leading to interruptions in processing valid incoming emails for a victim recipient MTA.

Preliminaries: To execute this attack, attackers should have (1) a domain name (2) a custom authoritative server for the domain to serve DNS records and manipulate DNS response time. More details on the execution is introduced in §5.4.

1. **DoS attack:** the attackers can disrupt the processing of benign incoming emails by deliberately slowing down DNS responses during SPF validation, causing timeouts and thus either delaying or outright denying email processing.
2. **Low economic barrier:** the attacker may (1) register a domain name using the TLDs that offer domain names at no cost (e.g., .ml [33]) (2) utilize a free-tier hosting service

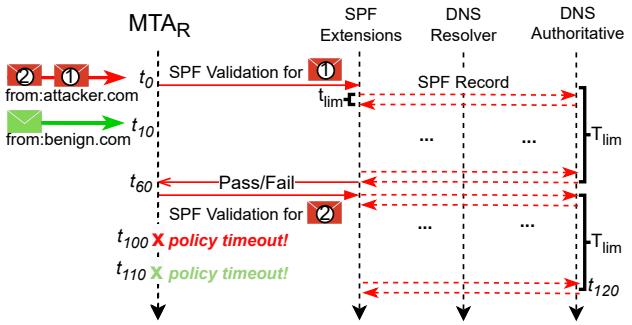

Figure 8: Certain SPF validators (e.g., iRedAPD or libspf2), do not support parallel SPF validation. In such cases, attackers can introduce delays up to either T_{lim} or $t_{\text{lim}} \times N_{\text{lim}}$ seconds; Each MTA has its own timeout limit for SPF validation, denoted as M_{lim}. Should the delay exceed this threshold, subsequent incoming emails may be rejected.

(e.g., AWS micro instance) to operate a customized DNS server designed to return DNS responses with extra delays.

5.4 DoS Attack for Inbound Emails

Each MTA sets individual timeout limits for SPF validation, leading to variations in the waiting time for process completion. For example, when a Policy Server is responsible for validation, the MTA may wait until a *policy service timeout* is triggered. Likewise, a *Milter command timeout* comes into play when Milter handles the validation. In the case of Postfix, the default settings specify a 100-second limit for policy service timeout and a 30-second limit for Milter timeout during SPF validation for incoming emails [31]. Should SPF validation exceed these predefined timeframes, the SMTP connection handling the email will be terminated.

This feature becomes a vulnerability when the SPF validators in use *do not support parallel SPF validation and dns lookups continue even after MTA aborts the connection*; as a result, if the validation process exceeds the *policy service timeout*, it can cause a cascade of timeouts, leading to the rejection of multiple incoming emails. This makes the emails undeliverable to the recipient MTA, effectively serving as a Denial-of-Service (DoS) attack vector.

For example, iRedAPD imposes a limit of 3 seconds (t_{lim}) for each DNS query and a maximum of 20 DNS lookups (N_{lim}); an attacker can exploit these constraints to induce delays. Specifically, by carefully crafting DNS responses that take exactly 3 seconds to resolve, the attacker can cause the SPF validation process to last for up to 60 seconds (3 seconds $\times$ 20). This can lead to a SPF validation exceeding the system's configured timeouts, effectively becoming a bottleneck and causing incoming emails to be rejected.

However, in practice, t_{lim} is also influenced by the DNS resolver's own timeout settings. Should the DNS resolver that the SPF validator employs have a shorter timeout value, the resolver could trigger its own timeout, potentially reducing the window for attack but also complicating the validation process. In such a scenario, attackers can take advantage of the fact that *CNAME record expansion is managed by the DNS resolvers, thus not contributing to* N_{lim}; this allows the attacker to induce the resolver into initiating a fresh DNS request, effectively resetting the resolver's timeout. For example, consider the attacker sets up the following SPF and DNS records:

```
a.com      TXT    "v=spf1 include:r1.a.com"
r1.a.com   CNAME  r2.a.com
r2.a.com   TXT    "v=spf1 include:r3.a.com"
r3.a.com   CNAME  r4.a.com
```

In this example, the attacker has purposely created a chain of `CNAME` records to prolong the DNS lookup process; when the DNS resolver reaches the `CNAME` record at `r1.a.com`, it is compelled to initiate a new DNS request for `r2.a.com`. Subsequently, the request for `r2.a.com` leads to another `CNAME` record (`r3.a.com`), triggering yet another DNS request and resetting the timeout. This can be extended further, thereby increasing the time needed for SPF validation but decreasing the chance to trigger t_{lim}.

By sending two emails simultaneously, the attacker can cause a cumulative delay of 120 seconds; given that Postfix's default policy service timeout is 100 seconds [31], all incoming emails in the 20-second window following the initial delay—from t_0 to t_{20}—would be rejected. This situation is illustrated in Figure 8. Specifically, the second email initiates its SPF validation at t_{60} and will hit the policy timeout, leading to termination of the SMTP connection at t_{100}, but blocking further SPF validation until t_{120}.

It is worth noting that SPF validation often occurs after the `RCPT TO` command; this allows attackers to potentially initiate a DoS attack without actually dispatching emails, making detection and mitigation more challenging.

5.4.1 Responsible Disclosure

We have reported a vulnerability concerning our proposed attack to the corresponding GitHub repository's issue tracker.[7] The previously identified issue with Mail::SPF::Query versions that were susceptible has been resolved. Furthermore, to assist users in assessing the security of their domains, we have incorporated a search feature on our website to let them inspect the status of their SMTP servers. Importantly, this feature is designed to prevent the scraping of entire lists.

6 SPF Validation in Practice

In this section, we augment our empirical data with a survey conducted in late 2022 to gain a more comprehensive view

[7] `https://github.com/iredmail/iRedAPD/issues/19`

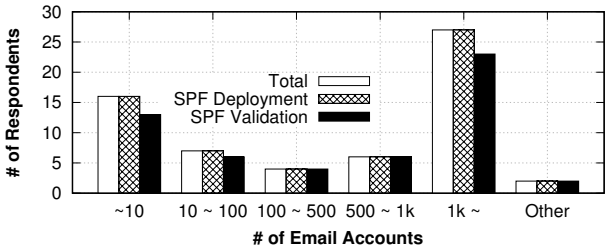

Figure 9: The figure shows the distribution of the number of email accounts managed by each of the 62 respondents who answered both questions regarding SPF deployment and SPF validation support; note that all respondents serving as SMTP administrators confirmed the deployment of `SPF` records.

of SPF validation in the email ecosystem. While our passive measurements and in-lab tests provide valuable quantitative insights, they are based on publicly accessible information and open-source software. The survey fills the knowledge gap, offering a qualitative look at operational practices and challenges in real-world SPF deployments.

6.1 Survey Methodology

We sourced participants for our survey from specialized mailing lists, including the Mail Operators' List (MailOP) [2], the North American Network Operators' Group (NANOG) [29], and Email Security Standards in the European Union (MESSEU) [28]. Of the initial 95 respondents, 74 engaged with at least one survey question; The participant demographics, summarized in Figure 9, reveal a wide range of mail-setup sizes: from 16 operators overseeing fewer than 10 accounts, to 25 operators responsible for more than 1,000 accounts. This variety suggests that our survey offers a broad view of the mail operator landscape. In summary, our diverse participant sample enhances our understanding of email infrastructure; also, the input from mail operators is particularly valuable as it plays a critical role in the standardization of protocols within the email ecosystem.

However, we also note that our survey has limitations worth noting for proper interpretation of the results. The small sample size offers a qualitative view rather than a basis for broader generalization. While we targeted knowledgeable operators, we did not verify their responses, which may introduce self-reporting and social desirability biases. Participation was voluntary, potentially leading to self-selection bias. Despite these limitations, our survey provides valuable qualitative insights into email operations, complementing our technical findings.

Ethical Considerations: Our survey focuses on organizations rather than individuals and collects factual data on system deployments. Our Institutional Review Board (IRB) confirmed that this does not qualify as human subject research, eliminating the need for studying protocol evaluations. Nonetheless, we adhered to best practices similar to those for human subject research, informing participants of their data access rights and the ability to withdraw at any time.

6.2 Operational Practice for SPF Validation

SPF Deployment and Validation: We first understand how they *deploy* email security protocols by publishing `SPF` records; interestingly, we observe that all of the participants (74) answered that they publish SPF for source validation; among them, 45 (60.8%) and 39 (52.7%) deployed DKIM and DMARC for the same purpose as well. In contrast, the adoption rates for other protocols aimed at encryption and receiver validation are somewhat lower; for example, MTA-STS is used by 13 participants (17.6%) and DANE by 19 (25.7%).

Regarding SPF validation, 62 participants responded to whether their SMTP server validates sender's SPF record or not and 54 participants (87.1%) answered yes, while only 8 explicitly indicated they do not.[8]; these findings underscore the critical importance of SPF in the realm of sender authentication. Firstly, among the 32 participants aware of when SPF validation occurs, we observe a multi-stage process: 3 perform validation after the `HELO/EHLO` command, 7 after the `MAIL FROM` command, 10 after the `RCPT TO` command, and 12 (37.5%) proceed after the `DATA` command. This variability underscores the feasibility of the attack vectors we propose, making it challenging for receivers to detect their unwitting role as reflectors when they check before the `DATA` command. Secondly, 14 participants (26%) report utilizing *multiple* extensions for SPF validation, potentially increasing the volume of DNS requests for each validation process. This aligns with our experiments, showing that 1.1% of SMTP servers issue more than 51 queries during SPF validation.

RFC Recommendation vs. Practice: Among the 14 participants who were aware of their SPF lookup limits, three have opted to increase these limits, and one has additionally raised the void lookup limit, suggesting that the current RFC guidelines may not align with practical operational needs; interestingly, *all four operators serve customer bases exceeding 1,000 accounts*, suggesting that the existing limits in RFC7208 for lookup limits may not adequately serve large-scale email operators. We also note that four of these participants have extended their total SPF resolution timeout.

Software Update and Others: Of the 61 participants who responded to questions about software update frequency, 13

[8] The remaining 12 respondents did not answer the question.

disclose that they *do not* update their SPF validation software periodically, with one explicitly stating they never do so. This raises concerns about the potential vulnerabilities in their SPF validation process.

SPF Validator Usage: Out of 54 participants who responded to questions about which SPF validator they use, 19 (35%) reported to use SpamAssassin [6], 14 (26%) use policyd-spf [48], 6 (11%) use libspf2 [45], 14 (26%) use Rspamd [3]. 14 (26%) participants use MTAs that natively support SPF validation while 4 (7.4%) of them did not know which validator they used. Out of the remaining 4 participants, one reported to use mtpolicyd [47], one iRedAPD [44], one amavisd-new [43], and one participant use Google's in-house solution. Note that 18 (33%) participants reported to have multiple SPF validators.

In summary, our survey reveals diverse practices in SPF validation among operators, suggesting both potential vulnerabilities and adaptability, with notable deviations from RFC7208 and varying degrees of operational vigilance.

7 Related Work

SPF Deployment for Outbound Emails: There have been a few studies that have focused on the deployment of `SPF` *records*; for example, Mori et al. [26] investigated the top 500 Alexa domains in 2011, finding that half had deployed `SPF` records but also noting common syntactic errors in their SPF policies. Foster et al. [17] extended this work by examining the Alexa 1M domains in 2015, revealing that 40% of these domains had valid `SPF` records. Hu et al. [18] and Wang et al. [40] continued this trend by showing increasing deployment rates, 44.9% and 54.1% respectively, in subsequent years. Recently, in 2023, Czybik et al. [13] conducted a scan of 12 million domains, revealing that 56.5% have implemented `SPF` records and Blechschmidt et al. [7] reported a 41% `SPF` deployment ratio among the DomCop Top 10M domains.

SPF Validation for Inbound Emails: The evaluation of SPF validation policies for inbound emails presents unique challenges. Earlier studies have primarily resorted to one of three methods: (1) analyzing popular email server logs, such as Gmail, (2) registering with popular email service providers and testing their services, or (3) sending emails to assess validation protocols. For example, Durumeric et al. [16] examined SMTP handshake logs for emails sent to and from Gmail between 2014 and 2015; their results showed that Gmail successfully validated 92% of incoming emails using SPF. Similarly, Foster et al. [17] evaluated popular email providers and found that 15 out of 22 had `SPF` records for outgoing emails. Moreover, 20 out of these 22 providers performed SPF validation for incoming emails, although only 10 actively enforced the SPF results.

More recent research by Hu et al. [18] revealed that 31 out of 35 top email providers validated `SPF` records for inbound messages while Blechschmidt et al. [7] revealed 46 of 47 providers did so. This underscores the widespread adoption of SPF validation in the wild.

In 2021, Casey et al. [14] *sent* 26K *legitimate* emails to measure SPF, DKIM, and DMARC validation rates, discovering that up to 85% of domains performed SPF validation for incoming emails. For a separate set of domains, they also measured SPF behavior by making an SMTP connection and then aborting the connection right *after* sending the `DATA` command but before sending any email content, which allowed them to measure SPF validation behavior of 1,574 MTAs. For our study, we intentionally disconnect *before* the `DATA` command since this command is intended for sending emails and might potentially activate spam extensions, leading to unnecessary computational resource consumption.

SPF Misconfiguration: Misconfiguration of SPF policies is a significant concern, as highlighted by numerous studies; for example, Durumeric et al. [16] reported that 29% of mail servers had overly permissive SPF policies that included more than 10,000 addresses. Hu et al. [18] indicated that 0.1% of domains in the Alexa 1M dataset had an `SPF` record that allowed all IP addresses to pass.

Tatang et al. [38] performed a graph-based analysis on the Alexa 1M dataset and observed that many domains included overly broad SPF policies from large providers like Amazon or SendGrid, making them vulnerable to spoofing attacks. Moreover, they found domains that permitted IP addresses from as many as 9,000 different ASes, marking them as potential targets for attackers. Scheffler et al. [36] discovered that several email servers did not adhere to RFC limits on the number of SPF referrals by connecting to *the entire IPv4 address space and sending emails using popular usernames*, thereby exposing a gap in compliance with standards.

Recent studies have even also pointed to potential security vulnerabilities in SPF implementations; Jeitner et al. [20] discovered possible injection and buffer overflow vulnerabilities in two popular SPF implementations, namely `policyd-spf` and `libspf2`.

While these previous works offer valuable insights into SPF misconfigurations, our study distinguishes itself by aiming for a comprehensive evaluation of the current SPF landscape. Our study extends these prior works in two ways. First, we undertake what is, to our knowledge, the most comprehensive assessment of SPF deployment, covering 176 million domains and their respective misconfigurations across a 15-month period. Second, our investigation is not solely confined to descriptive statistics; we also delve into the root causes

of prevalent misconfigurations. We employ a mixture of quantitative measurement data and qualitative analysis to explore the underlying mechanisms contributing to these issues. Specifically, we examine the security implications of exceeding RFC-recommended DNS lookup limits and investigate the rationale behind why email operators may choose to deviate from these guidelines.

8 Concluding Discussion

We presented a multi-faceted, deep-dive investigation into SPF records and their management, encompassing both a quantitative analysis based on comprehensive scans and qualitative insights obtained through operator surveys. We found that a significant majority (60.2%) of SPF records use the include mechanisms, largely depending on external domains. Most SPF records are syntactically correct, however, they falter in the evaluation phase—primarily due to excessive DNS lookups that surpass RFC-imposed limits; our study points to two major culprits behind these misconfigurations: popular hosting providers with large number of include in their SPF records and SMTP administrators failing to update obsolete records post-migration. We demonstrate a novel attack vector causing disruptions in email receipt without alerting the victim. Our operator surveys reveal that many large-scale email providers deviate from RFC best practices, underscoring a significant gap between existing standards and current operational requirements.

In summary, our work serves as a comprehensive resource for understanding the state of SPF management, its security implications, and avenues for improvement. It is our hope that this paper will act as a catalyst for more secure and effective email systems moving forward.

8.1 Recommendations

Altogether, our findings illuminate the present landscape of SPF deployment and management. We offer insights on revisions needed for existing recommendations and RFCs.

- Given the increase in (nested) SPF policies over time, we suggest increasing the current lookup limit of 10 to, e.g., 20 as it is already handled in several implementations.
- Domain owners and email operators should make use of subdomains for (e.g., newsletter) sending or other tasks delegated to an external email service provider to avoid the 'too many lookups' issue. These subdomains can then have a smaller SPF policy, only including, e.g., the service used for newsletters (news.example.com) or the CRM software (support.example.com).
- Email service providers on-boarding a new client should not only validate that client's SPF policy for the correct inclusion of *the provider's records*, but should also check whether their customers have superfluous records.
- Non-parallel email receipt/milter execution should be avoided, and milter operators should more clearly document it if their implementation or certain configurations for their software leads to non-parallel email receipt.
- Depending on the preferences of an email operator, they might choose to perform SPF queries after the remote server indicates the end-of-data ('.\n\n'), but before accepting a message; this approach can be advantageous for protecting against the attack we proposed. However, this strategy might result in missing the opportunity to reject connections from unauthorized sources early, leading to memory consumption (to load the message in the DATA command). Therefore, email operators must carefully configure their systems according to their needs.

Furthermore, we have to acknowledge that especially milter libraries for established protocols like SPF are a part of the often forgotten and regularly maintained by but a few essential building blocks of modern technology such as NTP [34]. They hardly change, thus often hardly require maintenance–unless vulnerabilities occur [8]–and hence it becomes easy for such milter projects to slide into obscurity and abandonment.

Finally, the common truth of operating digital infrastructure also remains true; operators should regularly check for updates to components of their setups.

8.2 Limitations

Our study has several limitations. First, our analysis of superfluous SPF records based on MX records is susceptible to false positives. This inaccuracy can be attributed to two factors: (1) Domains not listed in MX records might still be legitimate email senders, such as newsletter services. (2) Our 90% threshold heuristic may not universally hold true. To overcome these limitations, we confined our analysis to domains with more than 100 associated senders and corroborated our results with ground-truth data from the top 10 email providers.

Second, in §5, we have classified SMTP servers that perform more than 50 record lookups as vulnerable, which may introduce false positives. However, as elaborated in Table 5, our analysis indicates the maximum lookup limit identified across all evaluated software is 30, suggesting a lower likelihood of false positives.

Third, SPF queries were received from 81,843 (6.8%) recipient MTAs we successfully connected to. Despite appearing modest, we believe that this number represents a well-balanced trade-off compared to the numbers obtained by sending emails (e.g. 26K SPF validating MTAs in [14]). Furthermore, we evaluated 10 popular open-source SPF validators; although there is a chance that some less widely used software may have been missed, we have still covered validators of 49 (91%) administrators who participated in our survey.

Fourth, we used `postmaster` as the recipient username. Out of 1.2M servers we were able to successfully connect to, we got a `negative` response to the `RCPT TO` command for (679K) 56% of them, which indicates the absence of this user; MTAs are also often set up to whitelist `postmaster` address. Thus, even if `RCPT TO` command succeeds, sender validation might not take place.

Fifth, it is important to note that our study intentionally excludes the consideration of additional lookup limits resulting from `MX` or `PTR` records within an `SPF` record. This decision was made to avoid the ethical implications associated with initiating four connections to the same SMTP server.

Sixth, when identifying superfluous `SPF` records, we seek out the `SPF` records and their corresponding `MX` records to create a *pseudo-ground truth dataset* by focusing on the popular `SPF` records associated with more than 100 domains. Aiming for a high level of confidence in mapping to pinpoint superfluous records, we adopt a conservative threshold of 100 domains; for example, reducing this threshold to 20 domains notably expands the coverage, from 20,124 to 39,632 domains with superfluous records. However, we believe this adjustment leads to a high number of false positives.

Lastly, our analysis covers all second-level domains within four TLDs, thus not accounting for other domains that operate under subdomains.

Acknowledgments

We thank anonymous reviewers and our shepherd for their helpful comments. We are also thankful to Laura Atkins and Grant Taylor for giving valuable feedback on our work and Marteen Aertsen for his assistance with the survey to MEESEU, as well as to the email operators in the MailOP, NANOG, and MEESEU mailing lists who took part in our survey. This research was supported in part by NSF grant CNS-2323137, CNS-2247306, and the Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) [RS-2023-00215700, Trustworthy Metaverse: blockchain-enabled convergence research].

References

[1] Hotmail SPF outage. https://answers.microsoft.com/en-us/outlook_com/forum/all/my-hotmail-account-is-getting-errors-when-sending/b6b485dc-916b-486c-a084-5eb4be534efe.

[2] Mail Operators' List. https://www.mailop.org/.

[3] Rspamd. https://rspamd.com/.

[4] M. I. Ashiq, W. Li, T. Fiebig, and T. Chung. You've Got Report: Measurement and Security Implications of DMARC Reporting. *USENIX Security*, 2023.

[5] Add your SPF record at your domain provider. https://support.google.com/a/answer/10684623?hl=en.

[6] Apache SpamAssassin Project. https://spamassassin.apache.org/.

[7] B. Blechschmidt and B. Stock. Extended Hell(o): A Comprehensive Large-Scale Study on Email Confidentiality and Integrity Mechanisms in the Wild. *USENIX Security*, 2023.

[8] N. Bennett, R. Sowards, and C. D. SPFail: Discovering, Measuring, and Remediating Vulnerabilities in Email Sender Validation. *IMC*, 2022.

[9] S. Blank, P. Goldsten, T. Loder, T. Zinkn, and M. Bradshaw. Brand Indicators for Message Identification (BIMI). IETF, 2021.

[10] D. Crocker, T. Hansen, and M. Kucherawy. DomainKeys Identified Mail (DKIM) Signatures. RFC 6376, IETF, 2011. http://www.ietf.org/rfc/rfc6376.txt.

[11] F. Cangialosi, T. Chung, D. Choffnes, D. Levin, B. M. Maggs, A. Mislove, and C. Wilson. Measurement and Analysis of Private Key Sharing in the HTTPS Ecosystem. *CCS*, 2016.

[12] J. Chen, V. Paxson, and J. Jiang. Composition kills: a case study of email sender authentication. *USENIX Security*, 2020.

[13] S. Czybik, M. Horlboge, and K. Rieck. Lazy Gatekeepers: A Large-Scale Study on SPF Configuration in the Wild. *IMC*, 2023.

[14] C. Deccio, T. Yadav, N. Bennett, A. Hilton, M. Howe, T. Norton, J. Rohde, E. Tan, and B. Taylor. Measuring Email Sender Validation in the Wild. *CoNEXT*, 2021.

[15] D. Dittrich and E. Kenneally. The Menlo Report: Ethical Principles Guiding Information and Communication Technology Research. 2012. https://www.dhs.gov/sites/default/files/publications/CSD-MenloPrinciplesCORE-20120803_1.pdf.

[16] Z. Durumeric, D. Adrian, A. Mirian, J. Kasten, E. Bursztein, N. Lidzborski, K. Thomas, V. Eranti, M. Bailey, and J. A. Halderman. Neither Snow Nor Rain Nor MITM ... An Empirical Analysis of Email Delivery Security. *IMC*, 2015.

[17] I. Foster, J. Larson, M. Masich, A. C. Snoeren, S. Savage, and K. Levchenko. Security by Any Other Name: On the Effectiveness of Provider Based Email Security. *CCS*, 2015.

[18] H. Hu and G. Wang. End-to-End Measurements of Email Spoofing Attacks. *USENIX Security*, 2018.

[19] How to Set Up SPF for QQ Mail? https://www.tencentcloud.com/document/product/1097/44886.

[20] P. Jeitner and H. Shulman. Injection Attacks Reloaded: Tunnelling Malicious Payloads over DNS. *USENIX Security*, 2021.

[21] M. Kucherawy and E. Zwicky. Domain-based Message Authentication, Reporting, and Conformance (DMARC). RFC 7489, IETF, 2015. https://tools.ietf.org/html/rfc7489.

[22] P. Krumviede, R. Catoe, and D. J. C. Klensin. IMAP/POP AUTHorize Extension for Simple Challenge/Response. RFC 2195, 2195, RFC Editor, 1997.

[23] S. Kitterman. Sender Policy Framework (SPF) for Authorizing Use of Domains in Email. RFC 7208, IETF, 2014. https://tools.ietf.org/html/rfc7208.

[24] H. Lee, A. Girish, R. van Rijswijk-Deij, T. T. Kwon, and T. Chung. A Longitudinal and Comprehensive Study of the DANE Ecosystem in Email. *USENIX Security*, 2020.

[25] S. Liu, I. Foster, S. Savage, G. M. Voelker, and L. K. Saul. Who is .com? Learning to Parse WHOIS Records. *IMC*, 2015.

[26] T. Mori, K. Sato, Y. Takahashi, and K. Ishibashi. How is E-Mail Sender Authentication Used and Misused? *Proceedings of the 8th Annual Collaboration, Electronic Messaging, Anti-Abuse and Spam Conference*, Association for Computing Machinery, 2011.

[27] Mail::SPF::Query. https://metacpan.org/pod/Mail::SPF::Query.

[28] Modern Email Security Standards for EU (MESSEU). messeu@sys4.de.

[29] North American Network Operators' Group. https://www.nanog.org/.

[30] Postfix. http://www.postfix.org/.

[31] Postfix Configuration Parameters. https://www.postfix.org/postconf.5.html.

[32] P. Resnick. Internet Message Format. RFC 5322, IETF, 2008. https://www.rfc-editor.org/info/rfc5322.

[33] R. Roberts, Y. Goldschlag, R. Walter, T. Chung, A. Mislove, and D. Levin. You Are Who You Appear to Be: A Longitudinal Study of Domain Impersonation in TLS Certificates. *CCS*, 2019.

[34] F. Y. Rashid. Time is running out for NTP. https://www.infoworld.com/article/3144546/time-is-running-out-for-ntp.html.

[35] M. Salganik. Bit by Bit: Social Research for the Digital Age. 2016.

[36] S. Scheffler, S. Smith, Y. Gilad, and S. Goldberg. The Unintended Consequences of Email Spam Prevention. *PAM*, 2018.

[37] Sabotage! Coping with the Joe Job. https://www.sitepoint.com/sabotage-coping-joe-job/.

[38] D. Tatang, F. Zettl, and T. Holz. The Evolution of DNS-Based Email Authentication: Measuring Adoption and Finding Flaws. *RAID*, 2021.

[39] F. Tobias, L. Franziska, S. Florian, K. Thorben, L. Pieter, B. Randy, and F. Anja. Learning from the past: designing secure network protocols. *Cybersecurity Best Practices: Lösungen zur Erhöhung der Cyberresilienz für Unternehmen und Behörden*, 2018.

[40] C. Wang, K. Shen, M. Guo, Y. Zhao, M. Zhang, J. Chen, B. Liu, X. Zheng, H. Duan, Y. Lin, and Q. Pan. A Large-scale and Longitudinal Measurement Study of DKIM Deployment. *USENIX Security*, 2022.

[41] M. Wong and W. Schlitt. Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1. RFC 4408, IETF, 2006.

[42] What is SPF Flattening? https://acme.com/software/spfmilter/.

[43] amavisd-new. https://www.ijs.si/software/amavisd/.

[44] iRedMail - Open Source Mail Server Solution. https://www.iredmail.org/.

[45] libspf2. https://www.libspf2.org/.

[46] milter-greylist. http://hcpnet.free.fr/milter-greylist/.

[47] mtpolicyd. https://mtpolicyd.org/index.html.

[48] policyd-spf. https://manpages.debian.org/testing/postfix-policyd-spf-python/policyd-spf.1.en.html.

[49] pyspf. https://github.com/sdgathman/pyspf/tree/master.

[50] spfmilter. https://acme.com/software/spfmilter/.

9 Appendix

9.1 Survey Questionnaire

All the questions except in §9.1.1 were optional. In questions where we had `Other (please specify)` as the last option, a textbox was there for the participants to specify their answer. SCQ denotes single choice question, MCQ denotes multiple choice question, and YN denotes yes no question.

9.1.1 Page 1: Consent Form

Participants were presented with the following two mandatory consent questions:

- I consent voluntarily to be a participant in this study and understand that I can refuse to answer questions and I can withdraw from the study at any time, without having to give a reason.
- I understand that information I provide will be used for scientific reports and publications.

If the participant answered `No` to any of the above questions, the survey would end with no further input.

9.1.2 Page 2: Email Service

Both the questions came with a textbox.

- Would you be willing to provide the name of the organization whose e-mail service you manage? (If you don't want to, please enter `No`.)
- Would you mind sharing the name of the domain whose e-mail service you manage? We'd like the main domain name, not the domain name of your mail server. For example, example.com, not mx.example.com. (If you don't want to give the name, please enter `No`.)

9.1.3 Page 3: SMTP Service

SCQ What is the name of your email server software? Or, what is the name of your MTA? Options were `Postfix`, `Exim`, `Sendmail`, `Qmail`, `Exchange`, `Haraka`, `MDaemon`, `hMailServer`, and `Other (please specify)`.

SCQ Are you using any framework, setup tool, or how to for your mail server? Options were `iRedMail`, `Mailcow`, `OpenExchange`, `Plesk`, `Webmin`, `Mail-in-a-box`, `(ISP) Mail Server Howto`, and `Other (please specify)`.

SCQ How many email accounts exist under your operated infrastructure? Options were `< 10`, `10 - 50`, `50 100`, `100 500`, `> 500`, and `Other (please specify)`. This was a single choice question.

SCQ How many emails does your system receive on average per day? Options were `< 10`, `10 - 100`, `101 - 1000`, `> 1000`, and `Other (please specify)`.

SCQ How many concurrent connections can your mail server/ MTA handle on average? Options were `< 10`, `10 - 100`, `101 - 1000`, `> 1000`, and `Other (please specify)`.

SCQ Do you frequently update your MTA or associated plugins and mail filters? By update, we mean following the releases for the associated software and installing them regularly. Options were `Always`, `Often Sometimes`, `Rarely`, and `Never`.

Date If yes, approximately when did you last update your MTA or any of the plugins and mail filters? This had a datepicker.

MCQ What security email protocols do you use? Options were `SPF`, `Often DKIM`, `DMARC`, `BIMI`, `MTA-STS`, `DANE`, `STARTTLS`, `ARC`, and `Other (please specify)`.

9.1.4 Page 4: SPF Validation

YN Does your email server validate the sender's SPF record when it receives an email? Options were `Yes` and `No`. If the answer to this question is `No`, survey ends with no further input.

9.1.5 Page 5: SPF Validator

MCQ What is the name of your SPF validator? In other words, which plugin (i.e. milter, policy server, content filter) is doing the SPF validation when an email comes to your server? If your MTA natively supports SPF validation, please mark "MTA Native". Options were `SpamAssassin`, `policyd-spf`, `iRedAPD`, `libspf/libspf2`, `RSpamd`, `MTA Native`, `Don't Know`, and `Other (please specify)`.

YN Did you set up the SPF validator yourself? If the answer to this question is `No`, survey ends with no further input.

9.1.6 Page 6: Self-managed SPF Validator

MCQ When does your SPF validator perform the SPF check of the sender domain? Options were `After the HELO/EHLO command`, `After the MAIL command`, `After the RCPT command`, `After the DATA command`, and `After the email is received`.

YN Did you change any of the default values in your SPF validator? (Like changing the default DNS timeout of SPF validation). If the answer to this question is `No`, survey ends with no further input.

9.1.7 Page 7: SPF Validator Default Value Change

- Did you change any of the following default values in your SPF validator? If yes, please indicate how did you change them. Otherwise, please select the "leave as-is" option. FYI, the RFC recommendation for total DNS lookup and void lookup limits is 10, and 2 respectively. This question had three options: `Number of permitted DNS lookups per SPF check before an SPF permanent error is raised`, `Number of void DNS lookups per SPF check before an SPF permanent error is raised`, and `DNS timeout per SPF check before an SPF permanent error is raised`. Each option had three single choice radio buttons: `Increase`, `Leave as-is`, and `Decrease`.
- Please provide any rationales behind such updates (if you did so). This question had a textbox.