Lazy Gatekeepers: A Large-Scale Study on SPF Configuration in the Wild

Stefan Czybik
Technische Universität Berlin
Machine Learning and Security
Berlin, Germany

Micha Horlboge
Technische Universität Berlin
Machine Learning and Security
Berlin, Germany

Konrad Rieck
Technische Universität Berlin
Machine Learning and Security
Berlin, Germany

ABSTRACT

The Sender Policy Framework (SPF) is a basic mechanism for authorizing the use of domains in email. In combination with other mechanisms, it serves as a cornerstone for protecting users from forged senders. In this paper, we investigate the configuration of SPF across the Internet. To this end, we analyze SPF records from 12 million domains in the wild. Our analysis shows a growing adoption, with 56.5 % of the domains providing SPF records. However, we also uncover notable security issues: First, 2.9 % of the SPF records have errors, undefined content or ineffective rules, undermining the intended protection. Second, we observe a large number of very lax configurations. For example, 34.7 % of the domains allow emails to be sent from over 100 000 IP addresses. We explore the reasons for these loose policies and demonstrate that they facilitate email forgery. As a remedy, we derive recommendations for an adequate configuration and notify all operators of domains with misconfigured SPF records.

CCS CONCEPTS

• **Networks → Application layer protocols**.

KEYWORDS

Email, SPF, Authorization, Email Forgery

1 INTRODUCTION

Email still represents the prime form of communication on the Internet today. Despite several weaknesses of the protocol, billions of users regularly use email messages for business and personal exchange [10]. Due to its popularity, email is a constant magnet for cybercrime, serving as a vehicle for transporting unsolicited, fraudulent and malicious content, which ranges from spam and phishing attempts to targeted attacks and malware distribution [e.g., 8, 12, 14, 30]. These activities benefit from the lack of security mechanisms in the original protocols that cannot establish the authenticity of senders and content by itself.

Several extensions have been proposed over the last years to counter the misuse of email, including security mechanisms for the transport layer [11, 21], email headers [19], and message data [25, 27]. One of the oldest mechanisms to mitigate the spoofing of email senders is the *Sender Policy Framework* (SPF) [17]. Instead of retrieving emails from any network host, the receiving server can request an SPF record from the sender's domain and check whether the connecting IP address is authorized to send emails. In concert with other mechanisms, such as DKIM [19] and DMARC [18], SPF forms one central pillar for mitigating forged emails.

Despite this important role, however, the configuration of SPF in the wild and its weak spots are still an open field of research. The study by Gojmerac et al. [9] from 2014 indicates a moderate adoption of the mechanism and a tendency towards coarse authorization. Further studies in the following years show an increasing number of domains using SPF. In this work, we expand this view on SPF and present a detailed analysis of its configuration on the Internet. In particular, we use the Tranco list [22] to collect SPF records from 12 million domains over a period of 5 months. Based on this collection, we analyze the adoption, validity and permissiveness of SPF policies to learn how servers use this mechanism.

Our study reveals a growing adoption of SPF in practice. While Wang et al. report in 2022 that 54.1 % of the domains contain valid SPF records, we observe an adoption of 60.2 % for the top 1 million and 56.5 % for all 12 millions domains in our study. Unfortunately, we also uncover persisting security issues: First, 2.9 % of the SPF records suffer from errors, undefined content, or ineffective rules, undermining the intended protection. Second, we observe a large number of *very* lax configurations. For example, 34.7 % of the domains allow emails to be sent from over 100 000 IP addresses. We demonstrate in a case study that these coarse configurations give rise to spoofing email senders and thus unnecessarily weaken the protection of SPF in the Internet.

To mitigate this situation, we investigate the reasons for the lax policies and derive guidelines for a more restrictive configuration. Moreover, we have launched a notification campaign for all SPF records with invalid policies. In total, we have contacted 111 951 operators by email and informed them about incorrect or insufficient configurations. Feedback on these reports has been positive, and several operators promised to fix the reported problems. Two weeks after our notification, a scan of the domains shows that 6 931 (3.3 %) of the entries have already been corrected, and we expect further improvement over the next months.

Stefan Czybik, Micha Horlboge, and Konrad Rieck

Roadmap. We review the background of SPF in Section 2 and discuss related studies in Section 3. Afterward, we describe the methodology of our study in Section 4. Our findings are presented in Sections 5 and 6, where we first investigate invalid configurations and then explore the coarse use of SPF authorization. Our guidelines are presented in Section 7, before we conclude in Section 8.

2 BACKGROUND

The sending and receiving of email is realized on top of the classic Simple Mail Transfer Protocol (SMTP) [24]. Standardized in 1982, this protocol has been designed without built-in mechanisms to ensure the confidentiality of transmitted messages or to verify the authenticity of senders. As the importance and ubiquity of email has grown over time, the need for enhanced security measures has become increasingly apparent.

One significant security concern is the propagation of emails with forged sender addresses, for example, as part of spam and phishing campaigns. These forged emails exploit the lack of authenticity in SMTP and are a notorious threat to users. In response to this security gap, the Sender Policy Framework (SPF) [28] was introduced in 2003 as a standard to define approved sending servers of emails for a specific domain. To this end, a domain owner can configure a Domain Name System (DNS) record, which specifies a list of authenticated hostnames or IP addresses that are permitted to send emails on behalf of the domain.

While the introduction of SPF appears reasonable at first glance, it does face certain limitations. Primarily, SPF only addresses the authenticity problem by extending it to hostnames and IP addresses. This means that users have to trust their network provider or managed email service to accurately handle this aspect of email security. Furthermore, SPF introduces new problems when it comes to email redirection, which becomes problematic for mailing lists.

As another protocol to improve the authenticity of emails, DomainKeys Identified Mail (DKIM) requires the sending server to add a cryptographic signature to all outgoing emails. These signatures can then be verified by the receiving mail server, providing an additional layer of authenticity. On top of DKIM and SPF, Domain-based Message Authentication, Reporting and Conformance (DMARC) [18] adds a descriptive record to the DNS. This entry describes the behavior that a receiving mail server should adopt when an email is received and there are issues with SPF or DKIM authentication.

Note that SPF as well as DKIM, and DMARC are not able to provide reliable confidentiality, integrity, and authenticity for end-to-end communications like S/MIME [27] and OpenPGP [25]. These mechanisms, however, are not widely adopted yet and suffer from their own problems [23]. Unlike SPF, which is implemented in the application layer, these mechanisms are implemented on top of email and do not affect email servers. Consequently, large email providers, such as Google and Microsoft, recommend and enforce the use of SPF as a basic element of email security.

2.1 A Primer on SPF

The security mechanism SPF operates through DNS records that store a configuration of permitted IP addresses, networks or hostnames. This configuration is controlled by the domain owner and is publicly accessible. When a server receives an incoming email, it can perform a DNS lookup to retrieve the corresponding SPF record associated with the sender's domain. While processing the configuration, the server validates whether the email originates from an authorized source. In the following, we use the term *SPF record* to refer to the string in a DNS request of type *TXT* that starts with `v=spf1` and defines the configuration. The deprecated DNS type *SPF* is not considered in this work.

Technically, an SPF record is composed of different policy terms. These terms are either directives containing *mechanisms* with qualifiers, or *modifiers*. While modifiers provide additional information for the configuration of the policy, a mechanism defines a way to determine allowed IP addresses, networks or hostnames. Once there is a match between the sending host and a mechanism directive, the processing of the SPF record ends and the mechanism's qualifier is returned as the result of the authorization.

Mechanisms. We first take a look at the different mechanisms and their qualifiers.

`a`
> This mechanism matches if the sending IP addresses match the specified A or AAAA DNS records.

`mx`
> If an email originates from any of the hostnames or IP addresses specified in an MX DNS record, this mechanism matches.

`ip4, ip6`
> It is also possible to set allowed IP addresses. If the sender IP is listed here, this mechanism matches.

`all`
> As the name says, this mechanism matches all sender IPs. Everything after this term is ignored.

`exists`
> This mechanism can check if a specific domain or hostname exists in the DNS. If the hostname exists, this mechanism matches.

`include`
> The `include` mechanism allows a domain to include another domain's permitted sender IPs from its SPF record. This is useful to cross administrative borders at email delivery. The receiving server evaluates the content of the included SPF record as usual, but this mechanism only matches if the sender IP is explicitly allowed by it. Otherwise, and also in case of an error, the processing of the including record continues. Therefore, it is not possible to deny any or all IP addresses with the `include` mechanism.

`ptr`
> The last one, the `ptr` mechanism, checks if a reverse DNS entry for the sending IP address exists. This mechanism matches if the IP addresses of the sending host and of the domain name retrieved by the reverse lookup are equal. Since this is a slow mechanism that causes a high DNS load, using this mechanism is generally not recommended.

Except for `all`, the mechanisms can be specified by arguments. If no argument is given, the domain or IP address to be checked is used. The `a`, `mx` and both `ip` mechanisms additionally allow specifying

a CIDR prefix length to specify a complete network. If no CIDR prefix length is given, it will refer to a single host.

A qualifier can be placed in front of each mechanism to define the outcome in case the IP address of the sending email server matches. If a mechanism is specified without a qualifier, `pass` is implied.

+ (pass)

   The email server is authorized to send emails for the domain.

- (fail)

   The email server is explicitly not authorized to send emails for the domain.

? (neutral)

   There is no assertion about the email server.

~ (softfail)

   The email server is neither explicitly denied nor allowed to send emails for the domain. It is not authorized, but not strong enough to create a strict policy.

If the evaluation has found a match between the sending IP address and a mechanism, the qualifier is returned as a result and gives information about the authenticity of the sender. Note that the default result for SPF is not `fail`. If there is no explicit `fail` or `softfail` qualifier for the `all` mechanism, the SPF result for all hosts without another match is always the default value `pass`. If no mechanism matches, for example, because the IP address is not listed as approved sender and there is no `all` mechanism set, the result is `neutral`.

Modifiers. In addition to the mechanisms, there are also modifiers, of which for our work only the `redirect` modifier is relevant. This modifier allows a domain to delegate its SPF record to another domain. Like the `include` mechanism, this can be used to cross administrative borders, but in contrast to that mechanism, the complete evaluation process is performed on the redirected domain. Any statements after a `redirect` modifier are ignored.

Additionally, the evaluation of an SPF record at the receiving site can provide further return values. In particular, `none` is returned when there is no valid domain from the SMTP session or no SPF record. In the event of a transient error like a DNS error, a `temperror` is raised. If a DNS error is permanent, such as NXDOMAIN, a `permerror` is returned. The `permerror` is further used when the SPF record can not be evaluated correctly. In Section 5, we investigate the occurrence of these errors in the wild.

Example. Let us investigate the following SPF record:

```
v=spf1 +mx a:puffin.example.com/28 -all
```

In this example, we have multiple mechanisms: `mx`, `a` and `all`. The term `+mx` with the explicit `pass` qualifier specifies that the domain's MX servers are authorized to send mails. The next directive specifies an IP address range, namely the IP address of `puffin.example.com` with a /28 CIDR notation. As no explicit qualifier is given, the default `pass` is used, and all addresses in this range are authorized. The last directive, `-all`, enforces to reject emails from all other sources.

3 PREVIOUS STUDIES

Since email security is an important topic, we are not the first to measure the prevalence of sender authentication mechanisms. Over the last decade, the adoption of these techniques and possible vulnerabilities have been studied several times. In the following, we briefly review this work.

Studies from 2014 to 2018. In 2014, Gojmerac et al. [9] scanned the top 1 million of the Alexa ranking for DNS entries such as SPF and DMARC. Only about 37 % of the domains provided an SPF configuration at that time. In addition, Gojmerac et al. found several common syntactic errors in SPF records in their study, such as missing values for matching mechanisms like `ip4`, but did not quantify them further. In 2015, Durumeric et al. [4] gave a broad overview of the adoption rates of security extensions for SMTP. Besides protocols like STARTTLS and DKIM, they also investigated SPF for the Alexa top 1 million list but ignored sites without an MX record. They found that 47 % of the domains had published an SPF policy, indicating a growing adoption.

In the same year, Foster et al. [6] evaluated the security provided against network attacks by such extensions from a theoretical and practical view. As part of their study, they also scanned the Alexa list and, additionally, the top million mail domains from a leaked set of user data from Adobe. The result here was that 42.26 % of the Alexa domains and 43.60 % of the Adobe domains were using SPF. A few years later, Hu and Wang [12] investigated how email providers handle spoofed emails and if such could reach the inbox of the users. In this context, they searched the top 1 million domains from the Alexa ranking for SPF records and reported a slightly increased adoption rate of 44.9 %.

Studies from 2020 to 2023. Tatang et al. [33] measured the adoption rate of SPF, DKIM and DMARC and analyzed the relationship between different domains through included SPF entries as well as the domains and the autonomous systems belonging to the allowed IP addresses. Therefore, they scanned in 2020 over 2 million domains from different top lists, of which 50.7 % had published SPF records. Moreover, they reported 13 % invalid entries and, as the most common error, too many DNS lookups. The authors also mentioned that many records used different includes, and that sometimes large IP subnets are trusted.

In the same year, Kahraman [13] analyzed the usage of SPF on a dataset of about 168 million domains. In this very large dataset, 25 % of the domains had SPF configured and were further analyzed in terms of the used mechanisms and syntactic as well as DNS lookup limit errors. Trost [34] crawled, also in 2020, about 8.3 million domains from different top lists for SPF records to analyze trust relationships. The analysis showed that some domains allowed very large networks to send emails on behalf of them, raising concerns about possible attack vectors. The measurements in these three papers are close to ours, yet we provide a detailed analysis of the SPF records themselves, which allows us to characterize the security risks and notify the affected operators.

Two years later, Wang et al. [35] measured the deployment of DKIM and issues of the management. In their work, they also reported an SPF adoption rate of 54,1 % in the Alexa top 1 million domains. This result continued the trend from previous work, that has shown an increasing number of domains with such a policy.

Attacks on SPF. Deccio et al. [3] analyzed how email servers process and validate SPF entries. They observed that several servers ignore syntax errors and ambiguities of the specification, which could lead to various forms of attacks. In a similar vein, Shen et al. [29] investigated several security protocols, including SPF and developed attacks for the authentication by systematically exploiting details in the standards that are often implemented inconsistently. They proposed more accurate protocol descriptions to eliminate the ambiguous definitions, which in the end could also decrease the number of errors in DNS records. Another attack vector was described by Liu et al. [20]: In their work, they investigated different types of email forwarding and how these change header fields. In the end, the implementation of some forwarding techniques enabled the authors to circumvent methods like SPF and to send spoofed emails without detection.

Finally, the implementation of sender validation libraries itself can be a point of attack. Bennett et al. [1] demonstrated this using *libSPF2* as an example. They found multiple bugs in it and developed a technique to detect vulnerable servers remotely, revealing the widespread use of this library version.

Difference to our study. Our study continues the line of previous research and extends it with additional perspectives: We base our study on a larger dataset than most previous studies, except for the work by Kahraman [13]. This gives us a broader picture of the use of SPF in the wild. As a result, we are able to perform a detailed analysis of the flaws and weaknesses in SPF configurations, showing where and why authentication fails. This combination of a large dataset and detailed analysis provides valuable insights into common problems when applying the SPF framework. Moreover, we conduct a case study demonstrating that overly coarse authorization policies weaken the security mechanism and make it easier to forge emails with spoofed senders.

4 METHODOLOGY

Next, we introduce our methodology for investigating SPF records, their errors and potential threats. The goal of our study is threefold: First, with a large-scale measurement, we want to determine the prevalence of SPF across a wide range of domains. Second, we aim to shed light on how often and why SPF entries are flawed and thus only provide inadequate protection. Finally, we want to assess the occurrence and impact of overly coarse authorizations in SPF configurations.

4.1 Measuring SPF in the Wild

From a technical perspective, we have two options to measure the configuration of SPF: As the first strategy, we can collect a representative set of emails and extract all sender and recipient email addresses. From these addresses, we could generate a list of domains of email providers and examine their SPF records, similar to the study of Durumeric et al. [4]. Second, we can use a list of domains and retrieve all available SPF records from them, even if they are not intended to ever be used to send emails. While the first strategy helps to understand how SPF is used relative to the distribution of email providers, the latter one provides a less biased view of SPF configuration in the wild. Consequently, we pursue this strategy for our large-scale study.

Data source. We use the Tranco list of domains [22] for our measurements. This list is a research ranking of well-known and frequently used websites. We use the full lists of the first of the months from January until May 2023 and merge them to get a bigger amount of domains.

Crawler. We develop a crawler for collecting and parsing SPF records using the *checkdmarc library*[1]. The crawler retrieves the SPF record for a given domain using the function `query_spf_record()`. This function sends DNS requests of type TXT and SPF, but only returns the first SPF record from the type TXT request. The record is then parsed using `parse_spf_record()`. To analyze different weaknesses, flaws and misconfigurations, we modify the library. Our modified version returns all necessary values, such as the number of DNS lookups, permitted IP addresses and a parsed version of the SPF record. Warnings and errors in the SPF syntax are reported, and our modified version continues with the parsing afterward. Due to the scale of our study, we implement a cache to reduce the DNS load by not sending the same request twice. If an SPF record already exists in the database, the cached object is used instead of requesting and analyzing it again. This reduces the load from include mechanisms of large providers significant. Only for the first domain the include mechanism is processed, all others hit the cache. Moreover, we distribute and rate limit the DNS requests across 150 servers. The same procedure is applied for DMARC using `query_dmarc_record()` and `parse_dmarc_record()`. In the end, we collect the following information per domain:

- SPF record
- DMARC record
- MX record

Note that this information is publicly available and therefore no confidential or private data is collected in our study, see also Appendix A.

We then analyze the collected records by checking for errors and misconfigurations. Moreover, we evaluate the matching mechanisms of SPF and investigate the resulting authorization policy. For example, we determine the amount and type of authorized senders by recursively analyzing the `include` mechanism.

Measurement focus. The main focus of our study is to understand the configuration of SPF entries and the role of authorized hosts in the underlying policies. By analyzing the collected data, we can examine these properties in detail. However, there are also limitations resulting from our study design: First, we can analyze all SPF mechanisms except for `exist`. This can only be done with the first measurement strategy and a dataset of representative emails. Second, we restrict our study to IPv4 hosts. Durumeric et al. report that only 1.13 % of the mechanisms in SPF are `ip6` terms. In our scan, we find an even lower adoption rate. Only 0.5 % of the domains use IPv6 directly, which is why we refrain from a detailed analysis.

5 SPF ADOPTION AND ERRORS

We begin our examination of the collected data by first analyzing the adoption of SPF in the wild and comparing it to previous work. We then proceed with a detailed analysis of the uncovered errors and misconfigurations, expanding the scope of previous studies.

---

[1]Available at https://github.com/domainaware/checkdmarc

Table 1: SPF and DMARC usage in the wild.

Study	Year	List	Size	SPF	DM.
Gojmerac et al. [9]	2014	Alexa	1M	36.7 %	0.5 %
Foster et al. [6]	2015	Alexa	1M	42.2 %	1.0 %
Foster et al. [6]	2015	Adobe	1M	43.6 %	0.9 %
Durumeric et al. [4][1]	2015	Alexa	1M	47.0 %	1.1 %
Hu and Wang [12]	2018	Alexa	1M	49.2 %	5.1 %
Kahraman [13]	2020	Alexa	1M	73.6 %	—
Wang et al. [35]	2022	Alexa	1M	54.1 %	11.9 %
Our study	2023	Tranco	1M	60.2 %	22.6 %
Tatang et al. [33]	2020	Other[2]	2M	50.7 %	11.5 %
Kahraman [13]	2020	None	168M	25.0 %	—
Our study	2023	Tranco	12M	56.5 %	13.6 %

[1] Only domains with MX record are considered in the evaluation
[2] Union of Alexa, Majestic and Tranco top 1M lists

5.1 SPF Usage

In total, we have scanned 12 823 598 domains for this study. While this expanded scan provides insights on the general configuration of SPF, it is not directly comparable with previous studies that have considered smaller sets from the top 1 million domains of the Alexa and other rankings. However, in our measurement, the result for the top 1 million domains is included. Thereby, it is comparable in terms of size and the fact that the domains are ranked. Therefore, we first focus on the top 1 million domains of the Tranco list[2] [22] generated on 01 May 2023.

Using this focus, we observe that the usage of SPF per domain has grown to 60.2 % of all scanned domains and 79.3 % for domains with MX record. A detailed comparison of our results with past measurements is shown in Table 1. We find that domains within the first 1 million use SPF and DMARC more frequently. But also for the complete 12 million domains, a clear increase in SPF usage to 56.5 % can be observed from our scan. Every second domain in our measurement is now employing this security mechanism. Additionally, Figure 1 provides an overview of all scanned domains and their adoption of SPF and DMARC.

We also observe an interesting phenomenon: 10.4 % of the domains without an MX record return an SPF record. At first glance, this may seem counterintuitive, since these domains specify which
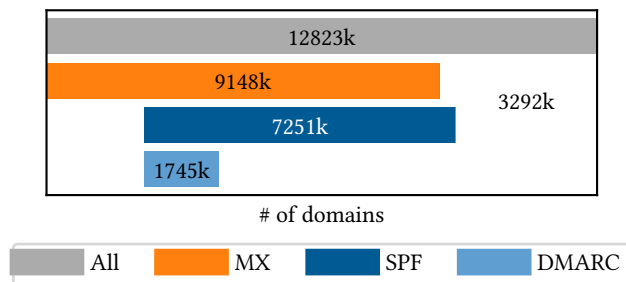

Figure 1: Implementation of email and security mechanisms and their overlaps.

senders are authorized through SPF but cannot receive email themselves. In several cases, these domains are not intended to send or receive email, and so the SPF record is used to deny sending email in general. We find that 53.1 % of the domains without an MX but an SPF record have SPF configurations containing `v=spf1 -all` (202 198) or `v=spf1 ~all` (1 143). However, the remaining half of these domains are likely misconfigured because they specify a sending policy but cannot receive bounces or other error messages from the transport, making them unsuitable for reliable email communication.

5.2 DMARC

In addition to SPF, we have also scanned for DMARC records using the *checkdmarc library*. We have done this to measure the increment from previous studies on email security. As shown in Table 1, DMARCs started with a low value of about 1 % in 2015 and is now at 22.6 % for the top 1 million domains and 13.6 % for all domains. The increasing usage of DMARC is likely due to the recommendations of large email providers[3]. Durumeric et al. already mentioned, that major email providers, such as Google and Microsoft, heavily skew the apparent adoption of security mechanisms.

5.3 SPF Errors

In our analysis of SPF, we observe a variety of errors in 2.9 % (211 018) of the domains, some of which are trivial typos while others are rather subtle misconfigurations. We hence explore these errors in more detail and count all issues as errors that affect the correct functionality of SPF. This includes all records that might result in a `permerror`. Figure 2 provides a general overview of all types of errors found.
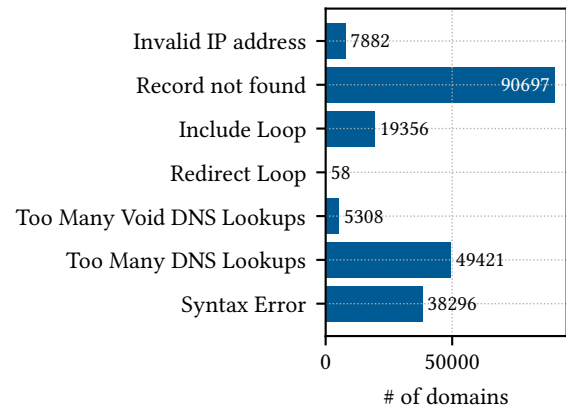

Figure 2: Appearance of different error types.

Note that during our scan, we received 1 179 DNS errors. This means that a domain that was supposed to be resolved when parsing the SPF record was not resolvable at that time. Since this may change on subsequent scans, we exclude these errors from the following analysis.

Record not found. First, we consider record-not-found errors indicating that no SPF record was found for a given domain name. These errors are the most common in our study with 42.98 %. They can be caused by either the `include` mechanism or the `redirect` modifier, since the SPF record of another domain must be parsed.

If we look at this error in detail, we see in Figure 3 that there are different causes. The most common cause with 53.8 % (48 824) of this error type is, that the requested domains has no SPF record. In contrast, there are 2.5 % (2 263) that provide more than one SPF record, which is no valid SPF record by the specification. An interesting finding here is that 75.6 % (1 711) of these errors are due to an `include` of the provider *cafe24.com*, which is a hosting provider for business customers. Another common record-not-found error is that the requested domain is not found (NXDOMAIN), as it happens 36 743 (40.5 %) times. This error could become critical if the domain is not registered and is taken over by an attacker. Other DNS related errors like a timeout, what is a `temperror` or an empty result are less common. The three other errors are one each of a DNS label is > 63 octets long, a DNS name is > 255 octets long, and one utf-8 decode error.
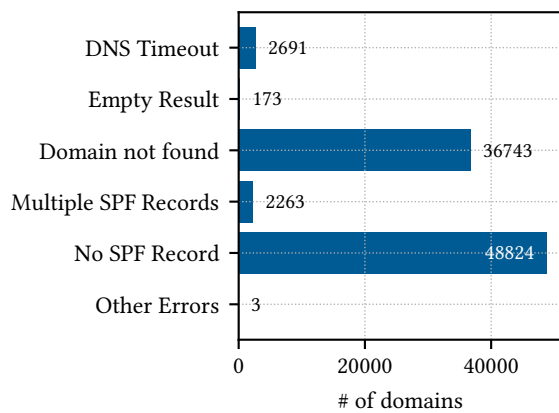

Figure 3: Distribution of record-not-found errors.

Too many DNS lookups. To prevent denial-of-service attacks, the number of DNS lookups that an SPF record may trigger is limited to 10 requests. It is the second commonest error with 23.42 %. As the specification is not totally clear here, we need to discuss this error in detail. RFC7208 says:

> The following terms cause DNS queries: the "include", "a", "mx", "ptr", and "exists" mechanisms, and the "redirect" modifier. SPF implementations MUST limit the total number of those terms to 10 during SPF evaluation, to avoid unreasonable load on the DNS.

The problem here is that for the `include` mechanism, there is no further description of how recursive DNS requests should be handled. As for `mx` and `ptr` mechanisms they are within the overall limit of 10, we assume this holds for the `include` mechanism too. In the *checkdmarc* library, this is implemented by counting the mechanism-related lookups during recursion. Another important fact is that this error does not have to lead directly to a `permerror`

in the SPF check. The SPF check can be successful if a result is returned within the first 10 lookups.

Now that we have discussed this type of error, let us get back to the underlying causes. As there is only a limited set of mechanisms that could create these errors, we find that the `include` mechanism is the main cause of this issue. Reasons for this include but are not limited to recommendations by email or web hosting providers. As an example, *bluehost.com*[4] is a provider that recommends customers to add an invalid SPF record that causes 14 DNS lookups. In Figure 4 we see a scatter plot of the includes where each dot represents an include. We zoom into the interesting part with more than 10 includes. In total, there are 2 408 included SPF records exceeding the DNS lookup limit directly, affecting 85 915 domains. 68 347 (79.6 %) of them are from *bluehost.com*.
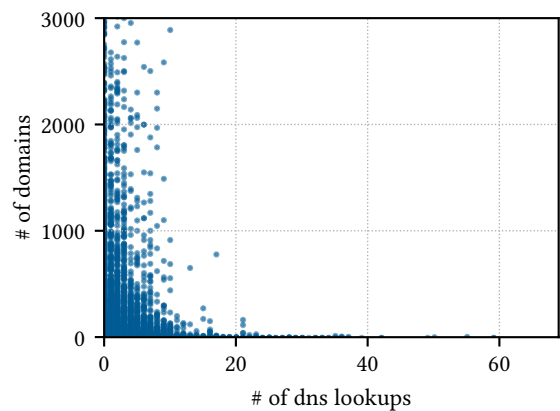

Figure 4: Cutout of the number of domains using a specific include depending on the DNS lookup count.

Too many void DNS lookups. This error is raised when there are two DNS errors during the evaluation of the SPF record. DNS errors are empty results or NXDOMAIN. With 23.42 % this error is less common. Since this error refers to a DNS lookup limit like *Too many DNS lookups* and the reasons for exceeding them are almost the same, we will refrain from explaining them in detail again.

Syntax Errors. A more interesting group of errors are syntax errors. These are caused by different oversights and shortcomings when creating an SPF record. Common errors are typos, wrong mechanism names or concatenating up different DNS records. With 18.15 % it is the third-largest error group and the most diverse one in our study.

In our manual investigation, the first common mistake in this group is that mechanisms are misspelled. We find that 11.0 % (4 216) of the syntax errors are using `ipv4` instead of `ip4` and 0.8 % (289) are using `ipv6` instead of `ip6`. 7.7 % (2 946) are just using `ip` as the wrong mechanism. Similarly, merging DNS entries also leads to errors. We observe that 7.0 % (2 699) of the errors are concatenations of the SPF record and a site verification string. When measuring the appearances of `v=spf1` in the SPF records, the result is that 15.3 % (5 847) of the records with invalid syntax contain more than one,

---

[4]https://www.bluehost.com/

which could be caused by combining multiple recommendations. A mechanism can have an argument that is placed directly after a `:`, however a whitespace in this position is causing 16.6 % (6 344) of the errors. Even though this group of errors is more diverse, they are typically easier to fix than other errors in SPF entries.

Include loops. An include loop is created when an `include` mechanism refers back to itself, either directly or at a deeper level of recursion. It is a less common mistake with 9.17 %. A direct inclusion of the domain happens in 71.6 % (13 850) of the cases. We assume that knowledge about SPF is not correct in these cases. When the error occurs at lower recursion levels, it is not obvious to detect and the cause is more intelligible.

Redirect loops. Loops can also occur with `redirect` mechanisms, representing 0.03 % of the errors. The causes are similar to the *include loops.*

Invalid IP address. Because IP addresses have a well-defined representation, they can be easily written incorrectly. In our analysis, this issue causes 3.74 % of all errors. In particular, we observe the following four types of errors:

- No IP at all
- Wrong number of octets
- A domain instead of IP address
- Wrong IP version

Overall, our analysis of the errors shows that one of their main causes is insufficient attention to detail when creating SPF entries. Although SPF is a simple mechanism, the development of configurations is non-trivial and sometimes fraught with small details. For example, DNS lookup limits are challenging to inspect, inclusions and redirections can cause different loops, and the syntax of some SPF mechanisms must also be carefully considered.

5.4 Notification

Our detailed analysis of errors puts us in a unique position: We become able to run a notification campaign, informing domain operators about the discovered problems in their SPF configurations. To this end, we follow the recommendations developed by Stock et al. [31, 32] for large-scale notification and contact each operator via email. To reach as many operators as possible, we use the general addresses `postmaster@` and `security@` as defined in RFC2142 [2] for our campaign. In addition, we send an email to the contact named in `security.txt` [7], if available.

Sending out notifications. In total, we sent 111 951 mails to notify domain operators with erroneous SPF records, except for record-not-found errors. We used a dedicated email server to deliver these huge amounts of emails. To avoid being blacklisted, we throttled the transfer rate to 1 mail per second. Based on this limit, we sent out all notifications in the second week of May 2023.

For each notification email, we follow a fixed template: First, we introduce ourselves and the scope of our study. Then, we list the identified problems for the particular domain, along with examples and recommendations on how to fix them. We are aware that our campaign causes additional work for the operators, and therefore strive to provide actionable items for each error type. Further details

on ethical considerations arising from this notification campaign are discussed in Appendix A.

Returned emails and feedback. It is clear that a notification campaign targeting hundreds of thousands of domains results in a large number of bounces and error messages. Nevertheless, we obtained a notable amount of positive feedback with thank-you notes, further questions and recommendations for future activities. By the time of the paper submission, we had received 300 grateful emails from domain operators. Only 3 responses were negative, and considered our notifications to be spam. We added the respective domains to an opt-out list so that they would not receive further security notifications from us.

Impact of notification. To learn about the practical impact of our notification campaign, we rescanned the domains with errors on May 24, 2023, that is, two weeks after the notification. We observe that 6 931 errors have been fixed by that time. In the same period 1 030 of the domains with errors disappeared, so there are no errors anymore. Table 2 shows detailed results for the different errors. The highest success rate is achieved with syntax errors and invalid IP addresses, as these can be easily fixed and do not require a deep understanding of SPF record evaluation. The errors with the lowest success rate are those related to DNS lookup limits. We assume that these are often non-trivial to fix, as they depend on the inclusion of external providers in the respective SPF configurations.

Table 2: SPF errors before and after our notification.

Error	Before	After	Change
Syntax Error	38 296	36 103	-5.73 %
Too Many DNS Lookups	49 421	48 630	-1.60 %
Too Many Void DNS Lookups	5 308	5 127	-3.41 %
Redirect Loop	58	56	-3.45 %
Include Loop	19 356	18 617	-3.82 %
Invalid IP address	7 882	7 498	-4.87 %
Total Errors	211 018	204 087	-3.28 %

Overall, our campaign achieves similar performance to notifications performed in previous work. Stock et al. [31], for example, report a 4.1% success rate in reporting web vulnerabilities via email. Our campaign achieves a success rate of 3.3 % just two weeks after sending the notifications, thus providing a similar effectiveness.

5.5 Additional Findings

We conclude our examination of SPF configurations with a discussion of further and curious findings discovered during the processing of our dataset.

Permissive all policies. For 5.9 % (427 767) of the domains, the SPF configuration is missing a restrictive all policy, which harms the effectiveness of SPF. The SPF evaluation will then just end without a matching mechanism and therefore return a `neutral` result. This may be intentional, as we will present in Section 6.2 for a few domains, but it leads to a reduction in protection. In most cases, we notice that a final deny directive is missing, such as `-all`. Here, we often spot typos as the reason for the problem, such as the invalid terms `-al` or `-all;` in the SPF entries.

Not recommended records. Over time, the SPF extension has evolved from the experimental RFC4408 [28] to a proposed standard in RFC7208 [17]. Due to this evolution, the DNS record type `SPF` has been deprecated since 2014. The `PTR` mechanism is not recommended anymore, as it is slow, not reliable due to DNS errors and produces a high DNS load. In our dataset, we find 107 646 domains still using this DNS record type and 233 167 domains using the `PTR` mechanism. As these versions still provide protection, we do not count them as errors in our analysis.

Implementation of abuse reporting. With RFC6652 [16], SPF was extended in 2012 with three new modifiers: `ra`, `rp` and `rr`. These modifiers allow the operator of a domain to be notified when an unauthorized email is rejected at an email server. Although this is a helpful extension, we notice only 14 domains implementing it in our dataset.

XSS attacks over SPF. Finally, we observe a cross-site scripting attack packaged in an SPF record of a domain. The attack looks as follows:

```
v=spf1 xss=<script>alert('SPF')</script> ~all
```

Since SPF parsers in email servers generally do not interpret JavaScript code, they should not be vulnerable to this type of attack. However, as soon as software displays SPF records in a web browser, there is a risk that the attack will succeed. This is, for example, the case for web services that check and validate SPF entries. Given the harmless payload of the attack, however, we assume that it is meant for testing purposes.

6 SPF INCLUDES AND SPOOFING

During our analysis of SPF entries, we observe several entries that authorize a very large number of IP addresses. In the following, we analyze these lax configurations in detail, investigate the underlying reasons and outline attacks that become possible through such configurations.

6.1 Number of Authorized IP Addresses

Since the goal of SPF is to authorize senders for a given domain, the number of allowed sending IP addresses should be minimal to reduce the attack surface. While the actual number of sending hosts for a domain is generally unknown, we can use the number of receiving servers to get at least an intuition of the general magnitude. Ruohonen [26] reports that domains listed in the Alexa rankings generally have fewer than 20 MX records. Therefore, we conjecture that the scale of sending servers is not significantly larger. However, we find that many domains in our study authorize *orders of magnitude* more addresses to send emails.

Figure 5 shows the distribution of the number of allowed IPv4 addresses as Cumulative Distribution Function (CDF). In line with our assumption, one out of three domains has fewer than 20 allowed hosts for sending emails. By contrast, we also find that almost the same number of domains authorizes more than 100 000 IPv4 addresses. The largest rise in the CDF is between 400 000 and 700 000 IPv4 addresses, mainly caused by including huge providers. In general, there are two ways a huge number can arise in the SPF record. First, we want to look at large IP ranges and later at includes.
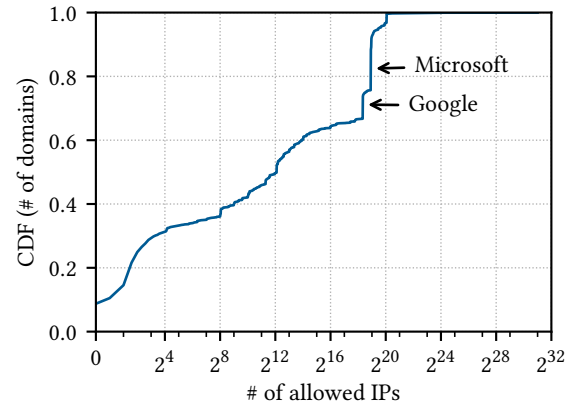

Figure 5: CDF of authorized IPv4 addresses.

6.2 Large IP Ranges

In general, the reason for intentionally allowing large IP ranges is not clear to us. To investigate, we take a look at SPF records that have more than 100 000 IP addresses allowed. Possible mechanisms are `a`, `mx` and `ip4`. We observed that 9 994 domains have their large number of IP addresses through these mechanisms.

Table 3: Type and amount of SPF mechanisms with large IP ranges.

CIDR	SPF: `ip4`, `a`, `mx`	SPF: `include`
/0	54	0
/1	29	2
/2	47	10
/3	16	7
/4	7	3
/5	6	0
/6	4	0
/7	4	0
/8	2 162	110
/9	23	3
/10	131	27
/11	44	50
/12	313	137
/13	228	210
/14	1 178	5 419
/15	1 145	5 389
/16	11 126	14 243

In Table 3 we see an overview about the appearance of very large IP ranges. At the hugest possible network /0 we found 39 domains explicit allowing `0.0.0.0/0`, what looks intentional. In contrast, there are 15 domains that have a specific IPv4 address with a tailing /0, what rather appears to be a misunderstanding of CIDR prefixes. Going ahead, the huge includes /1 and /2 appear to be typos that should refer to /16 and /24 respectively. Continuing the rows in Table 3 we can see that the includes are lower than the direct mechanisms, both at a very low level.

Therefore, these few large IP ranges in SPF Records cannot explain the huge amount of allowed IP addresses we see in Figure

5 around 2^{19}. In the next section, we will have a detailed look at the `include` mechanism and its impact on the number of allowed IP addresses. This is more promising to be the reason, as we observe that 2 507 097 domains authorize a large number of IP addresses through the `include` mechanism.

6.3 Usage of Includes

To get a better understanding of why so many IP addresses are included in the SPF entries, we analyze the use of the `include` mechanism. This mechanism is designed to cross administrative borders and is used by 67.0 % of the domains. Providers often recommend to their customers that they add a specific `include` mechanism to their SPF record when using their services. As already mentioned, 2 507 097 domains have a coarse policy with a huge number of allowed IP addresses from an include.

Trust relationships. In general, including a configuration from another domain involves a certain trust related risk. The owner has no control over possible changes in the inherited addresses, which could lead to spoofed email addresses with valid SPF check. Therefore, the domain owner has to trust the party they include from. While it should not be a big problem to trust the own provider in this case, things change if there are multiple inclusion levels and thus several administrators involved. As shown in Figure 6, most configurations have not more than one include, which seems reasonable in the described scenario. Nevertheless, we also observed 10 recursive includes and more, which raises the question if the domain owners are aware of everything they include and trust all involved parties.
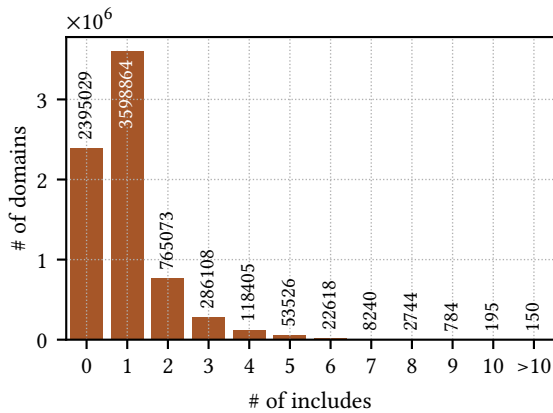

Figure 6: Number of includes in the top level record.

Included network size. Another interesting part of the includes are the networks allowed by them. Figure 7 shows the distribution of the used network sizes coming from the included SPF records. While most entries only include one IP address (/32 network), there is also a second notable peak for /24 networks. In the context of larger providers, load balancing and scaling, these sizes are understandable to share the load between multiple servers. Surprisingly, there are also SPF entries, which allow very huge networks, larger than /16. Even though large providers like Google might need a large number of servers sending mail for their customers, there

are obviously limits. We could not find a specific reason for these includes. Especially for less common domains, at the end of the Tranco list, we observe many /8 inclusions. Malicious senders could use these domains to send emails from many hosts within this list. What is interesting here is that most of the domains we observe in this context come from the ".top" top-level domain.
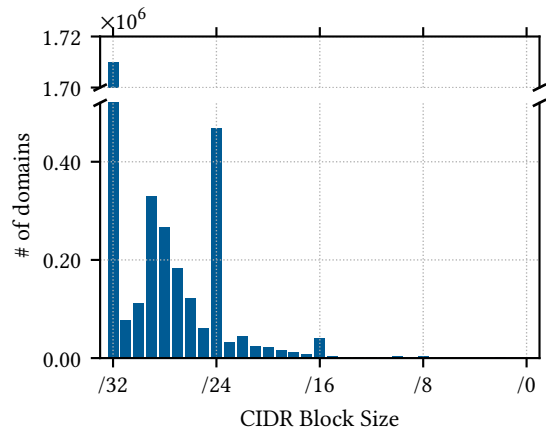

Figure 7: Distribution of subnet sizes in includes.

Number of IP addresses per include. We now take a closer look at how many times an `include` is used, depending on the number of allowed IP addresses. Figure 8 is a heatmap that shows the density of includes within a pixel of the plot, representing a logarithmic scale of the number of allowed IPs and how often the include is used. We can see that there is a huge concentration, up to around 2^{20} allowed IPs. Recalling Figure 5, we see that this correlates with the steep rise there at the same number of domains. From this observation, we can conclude that the large numbers of allowed IPs are typically from includes.

In Table 4, we report the top 20 includes we discover in our scan. As the first two, namely Microsoft and Google, include a huge amount of IP addresses. Similarly, other includes of providers are
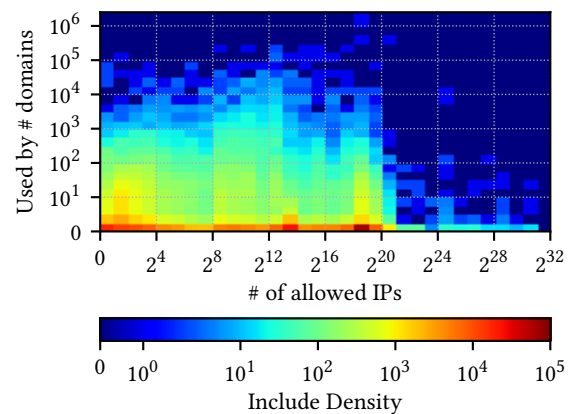

Figure 8: Heatmap of domains that are using a specific include, depending on the allowed IPs for the include.

larger than 100 000 and likely too coarse for proper protection. In contrast, there are also providers, like OVH[5] and Xserver[6], which include only a few sending email servers, demonstrating a restrictive authorization policy.

Table 4: Top 20 included domains with their number of allowed IPs.

Include	Used by	Allowed IPs
spf.protection.outlook.com	2 456 916	491 520
_spf.google.com	1 418 705	328 960
websitewelcome.com	414 695	1 088 784
secureserver.net	374 986	505 104
relay.mailchannels.net	289 112	4 358
servers.mcsv.net	263 343	22 528
spf.mandrillapp.com	236 293	4 608
sendgrid.net	215 497	220 672
_spf.mailspamprotection.com	212 418	1 049
spf.efwd.registrar-servers.com	196 465	264
amazonses.com	183 184	64 512
mx.ovh.com[1]	176 191	2
mailgun.org	172 499	36 312
_spf.mail.hostinger.com	139 423	4 358
zoho.com	138 227	6 209
mail.zendesk.com	114 026	26 112
spf.mailjet.com	111 760	5 120
spf.web-hosting.com	111 405	10 492
spf.sendinblue.com	102 004	87 040
spf.sender.xserver.jp	92 411	15

[1] Uses not recommended PTR mechanism

6.4 Case Study on Web Hosting

Our analysis shows that coarse authorization is a common practice in SPF configurations. From a theoretical point of view, it is obvious that overly permissive policies weaken the intended protection. However, whether these loose configurations can really help attackers in practice is not immediately clear. We therefore set out to investigate the risk of this practice in a case study on web hosting providers.

In particular, we focus on common web hosting providers that usually offer web space along with email support. Since these providers manage thousands of domains for their customers, they represent an essential part of the Tranco list and the collected SPF records. Moreover, as most web hosting providers support active content, such as PHP scripts, we are able to test the sending and authorization of SPF with permissive configurations.

As the basis for the case study, we search for recommended providers worldwide using the review website *hostings.info*[7]. From each country in the overview, we search for a recommended SPF record at the top 10 recommended web hosters. Due to the size of our measurements, we found 79 providers. Since many of the providers require national residency or the purchase of a national domain, we rent web space from 5 providers that do not impose any

[5]https://www.ovhcloud.com
[6]https://www.xserver.ne.jp/
[7]https://hostings.info/

constraints. We especially choose providers that offer PHP support, allow the use of own domains, and provide a short contract period. These providers are located in 4 countries (2xDE, FR, US, UK).

Imitating spoofing. To investigate the risk of spoofing due to lax SPF configuration, we sent emails from the selected providers to ourselves. In these emails, we spoof the sending domain by picking one that authorizes the IP address of the web hosting provider due to the recommended SPF record. Technically, we use two methods to realize this strategy: First, we try to send an email directly via SMTP from the web space via a corresponding PHP script. Second, we use the PHP function `mail()` to send it via the local Mail Transfer Agent (MTA) of the provider. We then examine how the emails are received on our site and whether they pass the SPF checks. A spoofing attempt is considered successful if one of the two methods succeeds in transferring a valid email.

Note that we only send emails from our rented web space to our own email addresses. Thus, spoofed senders in these mails do not cause any harm. Also, we have informed the vulnerable web hosting providers about their lax configurations in the hope that they will enforce stricter policies. For more details on ethical considerations, see Appendix A.

Results. We find that 4 of the 5 web hosting providers enable us to send emails with spoofed senders due to overly coarse authorization, as we can see in Table 5. In particular, there are two providers that enable sending emails with the PHP `mail()` function over their MTA. Among these two providers, we find 264 and 24 959 authorized domains, respectively. In addition, there is one provider that allows us to send emails via SMTP for 159 affected domains. With the fourth provider, we can even send emails via the SMTP method and `mail()` on behalf of 713 domains.

In summary, we are able to send emails with valid SPF entries from 26 095 domains, simply by renting web hosting space for about 30 Euro. Even worse, we can pick from a wide range of domains for the spoofing, including lobby organizations, political parties, health insurances companies, and even banks. Although our case study focuses on a small group of web hosting providers, it shows the potential for phishing campaigns and spam when lax SPF configurations are exploited by attackers.

Table 5: Results of the providers case study.

Provider	Success	# Domains	# Allowed IPs
1	MTA	24 959	177 168
2	SMTP, MTA	713	514
3	MTA	264	2 052
4	SMTP	159	3 074
5	None	0	672

7 LESSON LEARNED

After examining the prevalence of flaws in SPF implementations, we work out some recommendations on how to improve the use of SPF. We first discuss possible actions for domain owners before we take a look at the side of the web hosting providers.

7.1 Domain Owners

Domain owners are often dependent on other parties to operate particular services. This especially applies to email servers, where an external provider is often responsible for the security of the service. Nevertheless, in most cases, the domain owner needs to take care of all DNS records and therefore must provide the correct SPF configuration.

If the email server is operated by another provider, the domain owner should, in general, follow their recommendations for SPF records. As they might change IP addresses from time to time, this is usually a scenario for which the `include` mechanism is intended. However, we recommend checking the included addresses. As can be seen from our analysis, often only a single inclusion is necessary, making such a test technically feasible.

On the other hand, one of our findings is that providers may recommend including large IP ranges or additional includes in order to use their service. In these cases, we strongly recommend to check if the included ranges are only email servers used for the domain, potentially by contacting the provider and requesting a description of the includes. A further risk is an `a` mechanism in the SPF record of a shared web space. Every user on the same server that the A record points to could use this server to send an email on behalf of this domain. Ultimately, the recommended SPF entry can be taken as a rough indicator of whether a provider takes email security seriously and therefore serves as a decision-making aid for choosing a provider.

If the domain owners manage their DNS records themselves, they are fully responsible for the content. Our work shows that there are many entries with syntax errors, which could easily be prevented in advance. We therefore recommend validating SPF records with a tool to check for errors and undefined parts. In case of a self-hosted email infrastructure, administrators should ensure that they only add the hosts needed to send emails.

7.2 Web Hosting Providers

Web hosting providers generally want to offer their customers a well-functioning and user-friendly service, yet this sometimes conflicts with providing the best possible security. A well-designed setup can help avoid difficulties.

Customers should usually not be able to open SMTP connections directly to email servers on a shared web space, especially if this host is included in the recommended SPF entry. Therefore, it is a recommended practice to block outgoing connections to port 25 and related services. Nevertheless, users should be able to send emails within their application. Therefore, a local MTAs with proper authentication should be used to verify that the authenticated account is allowed to send emails on behalf of the specified domain.

If a customer needs to send emails directly using their own MTA, this should not be done using shared IP addresses. We recommend providing a user documentation to manually add an IP address to the entry, instead of automatically including it in the default SPF record. To prevent further problems with SPF, providers should enable their customers to understand and properly use this framework. Therefore, they should explain how it works or link to relevant material, and also point out potential risks associated with setting certain SPF mechanisms.

8 CONCLUSIONS

With our analysis, we shed light on the state of SPF in the wild. We observe an increasing adoption of this security mechanism; at the same time, we find flawed and overly coarse authorization policies in numerous cases. We demonstrate that these lax practices increase the attack surface of SPF and make spoofing senders possible with little effort. It is enough to identify web hosting providers that manage thousands of domains with permissive configurations to send spoofed emails at a large scale.

Fortunately, we can conclude from our notification campaign that several of the configurations were not intentionally malfunctioning. Shortly after our notifications, we could already observe thousands of fixed SPF entries. In general, SPF faces a tradeoff between security and usability. Although a minimal authorization policy would be desirable, operators often relax their configurations for practical reasons, for instance, because it is inconvenient to identify all sending hosts or because they try not to interfere with their clients' activities. Our analysis shows that the compromises made by operators are far from adequate, and we therefore strongly recommend using more validated and restrictive SPF policies in practice, for example, by following the guidelines presented in this paper.

REFERENCES

[1] N. Bennett, R. Sowards, and C. Deccio. Spfail: Discovering, measuring, and remediating vulnerabilities in email sender validation. In *Proceedings of the 22nd ACM Internet Measurement Conference*. ACM, 2022.

[2] D. Crocker. Mailbox Names for Common Services, Roles and Functions. RFC 2142, 1997. URL https://www.rfc-editor.org/info/rfc2142.

[3] 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. In *Proc. of the 17th International Conference on emerging Networking EXperiments and Technologies*. ACM, 2021.

[4] 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... In *Proc. of the 2015 Internet Measurement Conference*. ACM, 2015.

[5] European Commission. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation), 2016. URL https://eur-lex.europa.eu/eli/reg/2016/679/oj.

[6] I. D. Foster, J. Larson, M. Masich, A. C. Snoeren, S. Savage, and K. Levchenko. Security by any other name. In *Proc. of the 22nd ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2015.

[7] E. Foudil and Y. Shafranovich. A File Format to Aid in Security Vulnerability Disclosure. RFC 9116, 2022. URL https://www.rfc-editor.org/info/rfc9116.

[8] H. Gascon, S. Ullrich, B. Stritter, and K. Rieck. Reading between the lines: Content-agnostic detection of spear-phishing emails. In *Proc. of the International Symposium on Research in Attacks, Intrusions and Defenses (RAID)*. Springer Cham, 2018.

[9] I. Gojmerac, P. Zwickl, G. Kovacs, and C. Steindl. Large-scale active measurements of dns entries related to e-mail system security. In *2015 IEEE International Conference on Communications (ICC)*. IEEE Computer Society, 2015.

[10] T. R. Group. Number of e-mail users worldwide from 2017 to 2025 (in millions) [graph]. URL https://www.statista.com/statistics/255080/number-of-e-mail-users-worldwide/.

[11] P. E. Hoffman. SMTP Service Extension for Secure SMTP over Transport Layer Security. RFC 3207, 2002. URL https://www.rfc-editor.org/info/rfc3207.

[12] H. Hu and G. Wang. End-to-end measurements of email spoofing attacks. In *Proc. of the USENIX Security Symposium*. USENIX Association, 2018.

[13] G. Kahraman. Characterizing sender policy framework configurations at scale. Master's thesis, 2020. URL http://essay.utwente.nl/83315/.

[14] C. Kanich, C. Kreibich, K. Levchenko, B. Enright, G. M. Voelker, V. Paxson, and S. Savage. Spamalytics: an empirical analysis of spam marketing conversion. In *Proc. of the ACM Conference on Computer and Communications Security (CCS)*. ACM, 2008.

[15] E. Kenneally and D. Dittrich. The Menlo report: Ethical principles guiding information and communication technology research. Technical report, U.S. Department of Homeland Security, 2012.

[16] S. Kitterman. Sender Policy Framework (SPF) Authentication Failure Reporting Using the Abuse Reporting Format. RFC 6652, 2012. URL https://www.rfc-editor.org/info/rfc6652.

[17] S. Kitterman. Sender Policy Framework (SPF) for Authorizing Use of Domains in Email, Version 1. RFC 7208, 2014. URL https://www.rfc-editor.org/info/rfc7208.

[18] M. Kucherawy and E. Zwicky. Domain-based Message Authentication, Reporting, and Conformance (DMARC). RFC 7489, 2015. URL https://www.rfc-editor.org/info/rfc7489.

[19] M. Kucherawy, D. Crocker, and T. Hansen. DomainKeys Identified Mail (DKIM) Signatures. RFC 6376, 2011. URL https://www.rfc-editor.org/info/rfc6376.

[20] E. Liu, G. Akiwate, M. Jonker, A. Mirian, G. Ho, G. M. Voelker, and S. Savage. Forward pass: On the security implications of email forwarding mechanism and policy. In *2023 IEEE 8th European Symposium on Security and Privacy (EuroS&P)*. IEEE Computer Society, 2023.

[21] C. Newman. Using TLS with IMAP, POP3 and ACAP. RFC 2595, 1999. URL https://www.rfc-editor.org/info/rfc2595.

[22] V. L. Pochat, T. V. Goethem, S. Tajalizadehkhoob, M. Korczynski, and W. Joosen. Tranco: A research-oriented top sites ranking hardened against manipulation. In *Proc. of the 2019 Network and Distributed System Security Symposium*. Internet Society, 2019.

[23] D. Poddebniak, C. Dresen, J. Müller, F. Ising, S. Schinzel, S. Friedberger, J. Somorovsky, and J. Schwenk. Efail: Breaking S/MIME and OpenPGP email encryption using exfiltration channels. In *27th USENIX Security Symposium (USENIX Security 18)*. USENIX Association, 2018.

[24] J. B. Postel. Simple Mail Transfer Protocol. RFC 821, 1982. URL https://www.rfc-editor.org/info/rfc821.

[25] T. Roessler, M. Elkins, R. Levien, and D. D. Torto. MIME Security with OpenPGP. RFC 3156, 2001. URL https://www.rfc-editor.org/info/rfc3156.

[26] J. Ruohonen. Measuring basic load-balancing and fail-over setups for email delivery via dns mx records. In *2020 IFIP Networking Conference (Networking)*. IEEE Computer Society, 2020.

[27] J. Schaad, B. C. Ramsdell, and S. Turner. Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification. RFC 8551, 2019. URL https://www.rfc-editor.org/info/rfc8551.

[28] W. Schlitt and M. W. Wong. Sender Policy Framework (SPF) for Authorizing Use of Domains in E-Mail, Version 1. RFC 4408, 2006. URL https://www.rfc-editor.org/info/rfc4408.

[29] K. Shen, C. Wang, M. Guo, X. Zheng, C. Lu, B. Liu, Y. Zhao, S. Hao, H. Duan, Q. Pan, et al. Weak links in authentication chains: A large-scale analysis of email sender spoofing attacks. In *USENIX Security Symposium*. USENIX Association, 2021.

[30] C. Simoiu, A. Zand, K. Thomas, and E. Bursztein. Who is targeted by email-based phishing and malware?: Measuring factors that differentiate risk. In *Proc. of the Internet Measurement Conference (IMC)*. ACM, 2020.

[31] B. Stock, G. Pellegrino, C. Rossow, M. Johns, and M. Backes. Hey, you have a problem: On the feasibility of Large-Scale web vulnerability notification. In *Proc. of the 25th USENIX Security Symposium (USENIX Security 16)*. USENIX Association, 2016.

[32] B. Stock, G. Pellegrino, F. Li, M. Backes, and C. Rossow. Didn't you hear me? - towards more successful web vulnerability notifications. In *Proc. of the 2018 Network and Distributed System Security Symposium*. Internet Society, 2018.

[33] D. Tatang, F. Zettl, and T. Holz. The evolution of dns-based email authentication: Measuring adoption and finding flaws. In *Proc. of the 24th International Symposium on Research in Attacks, Intrusions and Defenses*. ACM, 2021.

[34] J. Trost. All your spf belong to us: Exploring trust relationships through global scale spf mining, 2020. URL http://www.covert.io/all-your-spf-are-belong-to-us-exploring-trust-relationships-through-gloabl-scale-spf-mining/. Accessed: 2023-09-13.

[35] 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. In *31st USENIX Security Symposium (USENIX Security 22)*. USENIX Association, 2022.

A ETHICS

Our university does not implement a formal IRB process for the conducted study. Still, we have designed all experiments in accordance with ethical best practices outlined in the Menlo report [15] and legal regulations of the European GDPR [5]. First, the collection of SPF and DMARC records is fully automated and does not involve human subjects. By design, the collected data is openly available and does not contain any private or sensitive information. Furthermore, we have taken measures to keep the load on DNS servers as low as possible. To this end, we implemented a cache as described in Section 4.1. Second, we have notified all operators of domains with misconfigured SPF records via email, providing detailed descriptions of the identified issues. Although this notification caused extra work for the operators, we argue that informing them about the misconfigurations and thereby improving email protection outweighs this disadvantage.