2018-10-01

A Quantitative Study of the Deployment of the Sender Policy Framework

Eunice Zsu Tan
Brigham Young University

A Quantitative Study of the Deployment of the Sender Policy Framework

Eunice Zsu-Chnn Tan

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of

Master of Science

Casey Deccio, Chair
Kent Seamons
Michael Jones

Department of Computer Science

Brigham Young University

ABSTRACT

A Quantitative Study of the Deployment of the Sender Policy Framework

Eunice Zsu-Chnn Tan
Department of Computer Science, BYU
Master of Science

Email has become a standard form of communication between businesses. With the prevalent use of email as a form of communication between businesses and customers, phishing emails have emerged as a popular social engineering approach. With phishing, attackers trick users into divulging their personal information through email spoofing. Thus, it is imperative to verify the sender of an email. Anti-spoofing mechanisms such as the Sender Policy Framework (SPF) have been developed as the first line of defense against spoofing by validating the source of an email as well as the presenting options of how to handle emails that fail to validate. However, deployment of SPF policies and SPF validation remains low. To understand the cost and benefit of deploying SPF, we have developed metrics to quantify its deployment and maintenance complexity through modeling. Our approach provides a way to visualize the SPF record of a given domain through the use of a graph. Using the developed model, we applied the metrics to both the current and historical SPF policy for the Alexa Top Sites for empirical study and historical trend analysis.

Keywords: SPF, Spoofing, DNS, SMTP

ACKNOWLEDGMENTS

I am grateful for my advisor, Dr. Casey Deccio, for his guidance and encouragement throughout this thesis. To Jennifer Bonnett for her willingness to help, as well as my family, who is always there to support me. I would also like to acknowledge Farsight Security for providing access to DNSDB, along with their partners, that made possible the historical SPF analysis from our study. Last, but not least, my gratitude goes out to Jack Demke for his contributions to the code used to analyze the data in the paper.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Email has become one of the standard tools for interaction between businesses and their customers. The number of email accounts is expected to grow from 3.1 billion in 2011 to 4.1 billion in 2015, with an estimated of 215 billion email messages were sent per day in 2016 [11]. With the prevalent use of email as a form of communication between businesses and their customers, phishing is naturally one of the popular social engineering approaches used by attackers to trick users into revealing important information, such as credentials. In a typical phishing setting, an attacker sends an email appearing to be from a legitimate organization, seeking to lure receivers into clicking a hyperlink in the email that brings them to a counterfeit website designed to steal their credentials. Phishing has since evolved to a more devilish variation in which users systems are infected with malware through opening an email attachment [4]. Malware, which used to be merely a nuisance that caused system malfunction, has now adopted a more malicious role of going into stealth mode with the intent of causing either financial damage to the users or turning their systems into "bots".

According to the phishing activity trends reports by the Anti-Phishing Working Group (APWG) APW [1], there was a 397% increase in the number of unique reported phishing cases from 284,445 in 2011 to 1,413,978 in 2015. An average of 415 brands per month from various industry sectors, such as ISP, financial and retail, were hijacked by phishing campaigns in the same year. The 2015 Verizon Data Breach Investigations Report [14] stated that as many as 23% of the recipients opened phishing messages, and 11% clicked on an email attachment. If trends continue, an estimated 325,214 (23% of 1,413,978) users will be tricked into opening a phishing email, and approximately 155,537 (11% of 1,413,978) will open phishing attachments, causing their systems to be infected with malware.

Phishing remains a successful form of social engineering attack since users cannot differentiate between a phishing and a genuine email. Both appear in the users mailbox though normal communication infrastructure [13]. To exacerbate the problem, Simple Mail Transfer Protocol (SMTP), the protocol designed to carry email around the globe, does not inherently prevent false impersonation or detect malicious intent. Another contributing factor to the rise in phishing is the role of humans: it is easier to breach a system by penetrating the weakest defense, humans, by exploiting trust [3].

The Sender Policy Framework (SPF) is an open standard anti-spoofing [12] protocol developed in 2003. With SPF, a domain designates authorized email senders such that email recipients can detect illegitimate senders. Major email providers (Gmail, Hotmail, Yahoo!, etc.) have implemented SPF, yet only 47% of the Alexa Top Million[1] deployed SPF [6] in spite of SPF being the de facto standard [15] for email source authentication.

Does the cost of implementing and maintaining an SPF record outweigh the benefits of domain protection? Can SPF be effectively deployed by organizations that use third-party email providers such as Google or Hotmail? Does the use of a third-party email provider complicate the deployment of SPF? We attempt to answer these critical questions by developing a model to analyze an SPF policy and quantify its complexity by applying metrics associated with that model. We apply our methodology to over 500,000 of the Alexa Top Sites, from both a current and a historical perspective. We observed an increase in the use of third-party organizations contributing to SPF policies in the past seven years. We also observed over this period an increase in IP addresses authorized to send email for a given domain. IPv4 remained the dominant protocol used among the SPF policies we analyzed, both in the present and historically. We also learned that 15.28% of the SPF policies we

[1] List of popular Internet websites rank according to the Alexa Traffic Rank algorithms.

analyzed can be evaluated without incurring additional DNS lookup beyond that required for the initial SPF policy.

The remainder of this thesis is organized as follows. Chapter 2 includes a brief description of SPF. Chapter 3 gives a summary of previous work in this area. Chapter 4 describes the methodology in model and metrics development. Chapter 5 presents the results of the empirical studying. Chapter 6 makes recommendation and concludes the research.

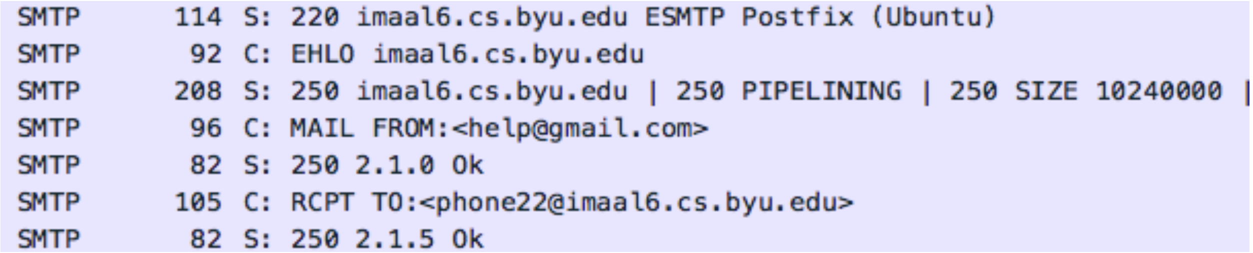
Chapter 2

Background

The main purpose of SMTP is to provide a simple, yet realizable transport protocol to act as an avenue for an email to be electronically transferred from sender to a receiver. SMTP gives the sender the discretion to identify itself by specifying their email address using the MAIL FROM identity section. This discretion is useful under certain scenarios, such as creating a mailing list where emails are sent by an authorized third-party vendor on behalf of a legitimate user. However, this flexibility also creates opportunity for abuse.

Attackers can exploit the insecurity of the MAIL FROM identity by impersonating someone else. Figure 2.1 shows an example of such exploitation; an email was sent via SMTP from a server within BYUs computer science network, yet the MAIL FROM indicates that the email is from `gmail.com`.

To counter email spoofing, SPF was developed to verify the legitimacy of the domain specified in the MAIL FROM identity by having the mail receiver validate the IP address of the received mail against a list of authorized hosts obtained from the domain in MAIL FROM identity. For SPF to work effectively, 1) the domain must publish the list of IP addresses



```
SMTP      114  S: 220 imaal6.cs.byu.edu ESMTP Postfix (Ubuntu)
SMTP       92  C: EHLO imaal6.cs.byu.edu
SMTP      208  S: 250 imaal6.cs.byu.edu | 250 PIPELINING | 250 SIZE 10240000 |
SMTP       96  C: MAIL FROM:<help@gmail.com>
SMTP       82  S: 250 2.1.0 Ok
SMTP      105  C: RCPT TO:<phone22@imaal6.cs.byu.edu>
SMTP       82  S: 250 2.1.5 Ok
```

Figure 2.1: Sample Capture of a Typical SMTP Communication

authorized to send via its SPF policy, 2) the policy must be validated by the SPF validator of the mail receiver, and 3) the SPF policy must adhere to the SPF proposed standard.

When properly deployed, SPF helps detect email spoofing. In the case of the example from Figure 2.1, the SPF validator extracts the domain portion of the MAIL FROM identity — `gmail.com`. It then retrieves the SPF policy for `gmail.com`, parses and evaluates the policy, then checks the sender's IP address against those specified by the policy. Because the IP address of the sender doesn't match any of those listed by the policy, the email is marked as suspicious.

To deploy SPF as specified in RFC 7208 [10], the system administrator for a domain creates an SPF policy and publishes it as a Domain Name System (DNS) record of type TXT (See Section 2.1 for background information on the DNS). However, SPF is not simply a list of IP addresses. It is composed of various terms, known as *mechanisms* and *modifiers*[1], this constitute a recipe for obtaining a complete list of authorized hosts. RFC 7208 is the proposed SPF standard describing the guidelines for operational processing, evaluation, and error-handling of these mechanisms and modifiers. A published SPF policy follows this syntax:

```
"v=spf1 qualifier [mechanism] |  modifier"
```

where `v=spf1` stands for SPF version one. There are altogether four qualifiers, eight mechanisms, and two modifiers. The mechanisms of the SPF policy provide instructions to the SPF validator to take actions such as fetching mail exchange (`MX`) records to ultimately derive the set of authorized IP addresses via additional DNS lookups. A match is found when a derived IP address matches the sender's IP address.

RFC 7208 dictates that the mechanisms be evaluated in turn from left to right, until the evaluation either returns a match or an exception. A mechanism can be prefixed with a qualifier and the qualifier value acts as the determining factor for the final return result to the SPF validator; e.g. a qualifier value of *"pass"* with a match condition will deem the

[1] See www.openspf.org/SPF_Record_Syntax for a comprehensive SPF syntax listing

sender's IP address as valid. The default qualifier for any mechanism not having one explicit stated is *"pass."* A modifier is an optional term that provides additional information to the SPF validator, such as fetching an SPF policy from another domain for further processing or tailoring its rejection message after policy evaluation. See section 4 for further information on qualifiers, mechanisms and modifiers.

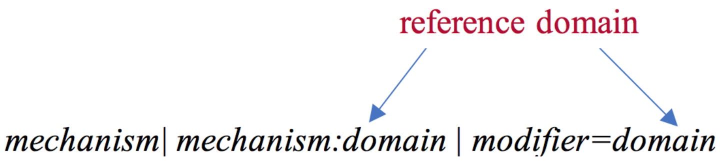


Figure 2.2: Reference Domain

Mechanisms `a`, `mx` and `ptr` can be either paired with or without a domain. On the other hand, the `include` mechanism and `redirect` modifier must always be suffixed by a domain. Any DNS query issues that come from the processing of mechanisms or modifiers suffixed with a domain must use the reference domain for its lookup. Mechanisms without the suffix will default to the current domain which is the base domain until it is replaced by the occurrence of a reference domain.

2.1 DNS

The architectural design of SPF is based largely on the DNS infrastructure. Thus, in this section, we will introduce the role and functionality of the DNS to increase the understanding of SPF and its policy processing.

DNS is the phone book for the Internet. The DNS is used to resolve human-friendly domain names to computer-readable IP addresses. The DNS infrastructure consists of: 1) the authoritative nameservers, which contain the domain-to-IP address mapping in the form of a DNS record; 2) the clients, which request the IP address of a particular domain name; and 3) the recursive resolver that queries the authoritative nameservers responsible for the domain name that is being translated.

There are many types of DNS records. We discuss four primary types that are related to our research: 1) the `A` record, which contains the IPv4 address for a domain name; 2) the `AAAA` record, which contains the IPv6 address for a domain name; 3) the `MX` record, which contains the domain names of mail servers for a given domain; and 4) the `TXT` record, which contains arbitrary text and, among other things, is used for declaring an SPF policy. A typical DNS lookup consists of a domain name and the record of interest: e.g., `example.com/AAAA`.

Even though the DNS works behind the scenes, the productivity of the entire Internet is dependent on the performance of the DNS infrastructure [9]. A simple DNS lookup might turn into a recursive query requiring multiple queries by the recursive resolver. Thus, any excessive or unnecessary DNS lookups have the potential to slow down the application relying on the lookups. To conserve DNS resources, caution should be exercised in the design of an SPF policy, especially those policies comprising of mechanisms and modifiers that use DNS lookups to derive the underlying IP addresses for the SPF validation process.

2.2 Common SPF Policies

For the purpose of understanding the work herein proposed, we will discuss some of the commonly used SPF mechanisms using the following two policies as an example.

```
imaal6.com.  IN TXT "v=spf1 a mx include:imaal3.com ~all"
```

- `a` - Expands to all IP addresses found in the `A` or `AAAA` DNS records corresponding to the domain. Since no domain name is explicitly referenced here, the domain name is implicitly `imaal6.com`. The A record contains IPv4 addresses, and the `AAAA` record contains IPv6 addresses.
- `mx` - Expands to the IP addresses corresponding to the `MX` records that are associated with the domain. Since no domain name is explicitly referenced here, the domain name is implicitly `imaal6.com`.
- `include:imaal3.com` - Indicates that the SPF policy for the reference domain `imaal3.com` should be looked up and included for further evaluation. Refer to section 2.3 for the policy processing related to include mechanism.
- `~all` - Used as the default mechanism at the end of the policy. "∼" is the qualifier that stands for soft fail; if validation fails, the mail should still be accepted but flagged.

```
imaal7.com. IN TXT "v=spf1 -all"
```

- `-all` - When used with the - qualifier (fail), no IP addresses are designated as authorized senders. Thus, no email should be sent from this domain.

2.3 SPF Policy Processing

To achieve our goals, we must understand the underlying procedure involved in processing an SPF policy, as outlined in RFC 7208. To illustrate the complexity of processing an SPF policy, we have constructed a hypothetical spoofing scenario where an attacker sends an email to company B, spoofing company A's domain. The following example details company A's SPF policy and the steps taken by company B to verify the sender IP against company A by using its SPF policy:

```
companyA.com. IN TXT "v=spf1 include:a.companyA.com
                        include:b.companyA.com -all"
a.companyA.com.  IN TXT  "v = spf1 a -all"
b.companyA.com.  IN TXT  "v = spf1 ip4:192.168.23.5 -all"
```

1. An attacker sends an email to company B, spoofing company A's domain.
2. Company B invokes its SPF validator to validate the received email.
3. The SPF validator issues a DNS lookup of type `TXT` to query for company A's SPF policy and begins to evaluate the returned policy from left to right.
4. The SPF validator issues a DNS lookup of type `TXT` to query for the SPF policy of reference domain `a.companyA.com.`
5. The SPF validator processes the returned SPF policy and issues a DNS query for the `A` record corresponding to `a.companyA.com.`
6. The SPF validator checks the sender IP against the returned IP address. Pass if matched, otherwise move to step 7.
7. The SPF validator issues a type `TXT` DNS lookup to query for the SPF policy of reference domain `b.companyA.com.`
8. The SPF validator checks the sender's IP address against the IPv4: 192.168.23.5. Pass if matched, otherwise return hard fail.

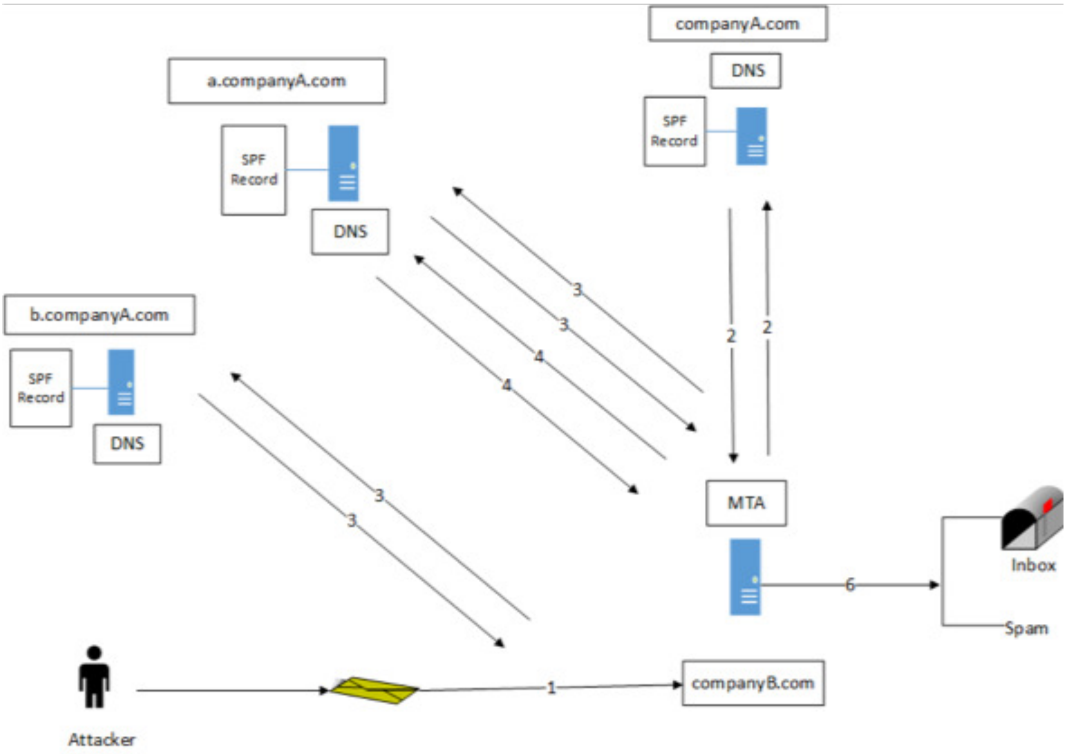


Figure 2.3: Example of `include` Mechanism

Company A uses the `include` mechanism in their SPF policy, which introduces the possibility of up to three DNS lookups beyond the original lookup to find its primary policy: 1) to obtain the SPF policy of `a.companyA.com` ; 2) to obtain the SPF policy of `b.companyA.com`; and 3) to get the IP address from the `A` record for `a.companyA.com`. As observed from the example, the `include` mechanism has the potential to introduce multiple DNS lookups and trigger a recursive evaluation.

Chapter 3

Previous Work

Even though there are benefits in implementing anti-spoofing mechanisms, there is generally a lack of incentives by the organizations in implementing the necessary protocols to secure their emails and reduce phishing. Anh et al. [2] attribute the low SPF adoption rates to philosophical issues. Network administrators exhibit little desire to work on something that only benefits others. They propose overcoming the philosophical issues by auto generating SPF records using email domain and IP information from email service providers. Anh et al. presented a rather interesting idea to build a comprehensive list of SPF policies for all the domains in the world, for which historical emails from every single email provider in the world would be needed. Dalkılıç and Sipahi [5], propose to automatically create SPF policies with a strict rejection model to circumvent the low adoption rates and increase the usefulness of SPF.

The non-existence of any inherent security for SMTP has opened the door for phishing abuse. This abuse has led to undertakings by different communities to propose, build and improve various SMTP security extensions such as STARTTLS, SPF, DKIM and DMARC. Durumeric et al. [6] has analyzed the global adoption rates of STARTTLS, SPF, DKIM and DMARC, while Görling [8] has studied the global adoption rates of SPF only. Görling, offers some explanations for the mild SPF adoption and encourage further adoption by discussing SPF merits.

Foster et al. [7] conduct a measurement study on the email security protocols supported by the major email providers such as hotmail.com, gmail.com and others. The security protocols being studied are DNSSEC, TLS, SPF, DKIM and DMARC. The study reveals that the support of SPF is common among the major email providers, however only a relative handful adhere to the RFC7208 recommendation of rejecting any mail that fails the strict rejection policy of "-all."

Our work further studies SPF by developing a model and metrics for assessing the complexity of an SPF policy. We apply our model and metrics to published SPF policies to learn whether policy complexity might be a hindrance in SPF deployment. We aim to improve the SPF adoption rates by equipping the domains with the methodology and tools presented in this thesis.

Chapter 4

Methodology

An effective and efficient SPF deployment lies in the appropriate use of SPF terms to create policies that accurately capture all the legitimate email senders for a domain and the conscientious effort to conserve DNS queries through the use of SPF terms. Thus, the complexity of an SPF policy is based on the number and nature of DNS queries required by an SPF validator to find a matching IP address for the email sender.

However, we have no idea how simple or how complex an SPF can be by simply looking at the policy; An SPF policy can recursively include other SPF policies via the use of the `include` mechanism. The `include` mechanism introduces administrative flexibility by allowing the administrator to designate an independent third-party email provider such as Gmail, Hotmail or Yahoo!, within an SPF policy. Paradoxically, this flexibility increases the policy complexity beyond the initial one-liner sentence.

In view of that, we developed a model to capture the complexity of an SPF policy by detailing the steps involved for the evaluation of each SPF term in the policy for the derivation of the IP address. The qualifier, an optional value, is only applicable after the derivation of IP addresses. Thus, it is irrelevant to our study and will not be part of our model. In addition, metrics were derived from the developed model to quantify the policy complexity. We use these metrics to measure the efficiency of the SPF policy. Refer to section 4.1 for model development and section 4.2 for the metrics and their uses.

4.1 Model Development

We use SPF_d to denote the SPF policy of a given domain, and we model SPF_d as a tree data structure because a tree intuitively depicts the recursive nature of the process associated with the evaluation of SPF_d. We define the tree associated with SPF_d as:

$$G_d = (V, E) \tag{4.1}$$

G_d stands for the directed, acyclic graph model of a given domain. V stands for the set of nodes that represent the evaluated SPF terms. E is the set of edges that connect the nodes in the tree. The edges represent the actions taken by an SPF validator during the evaluation of SPF terms. The positioning of nodes in the tree describes the ancestor and descendant relationships in the hierarchy of G_d.

4.1.1 Adding Nodes and Edges

SPF terms are the building block of G_d. We will describe the technicality of each term and the way they are added as nodes and edges to G_d.

Mechanism `all`

This is the catch-all mechanism which appears at the end of a policy to instruct the SPF validator to match any sender's IP address that fails to match any previous IP address in the policy. This mechanism does not require a validator to issue any DNS lookups nor does it specify any authorized IP addresses. Thus, it is not reflected in G_d.

Mechanism `ip4` and `ip6`

The `ip4` and `ip6` mechanisms specify actual IPv4 and IPv6 addresses (or ranges), respectively. These two mechanisms can be used by an SPF validator for direct matching without the need for further processing (i.e. ,with DNS lookups). For policy $u \in V$, with `ip4` or `ip6` mechanism having value v, v is added to G_d with the creation of edge *(u,v)*.

Mechanism `a`

This mechanism requires the SPF validator to issue a DNS lookup of type `A` or `AAAA` to find the IP address of the current or reference domain. For policy $u \in V$, with an `a` mechanism having value v:

1. v is added to the graph with the creation of edge *(u, v)*; and
2. nodes are created for each of the n IPv4 and IPv6 addresses to which v resolves, $w_1, w_2, ..., w_n$, and edges are added to connect each to $v : (v, w_i) \forall i, 1 \leq i \leq n$.

Mechanism `mx`

This mechanism requires the SPF validator to issue a DNS lookup of type `MX` to look for the `MX` records of the current or reference domain, as well as the IPv4 and IPv6 addresses corresponding to each `MX` record returned. For policy $u \in V$, with an `mx` mechanism having value v:

1. v is added to the graph with the creation of edge *(u, v)*;
2. nodes are created for each of the n domain names resulting from the DNS lookup of type `MX` for $v, w_1, w_2, ..., w_n$, and edges are added to connect each to $v : (v, w_i) \forall i, 1 \leq i \leq n$; and
3. For $1 \leq i \leq n, w_i$ is resolved to its `m` corresponding IPv4 and IPv6 addresses, and edges are drawn to connect each $w_i : (w_i, x_j) \forall j, 1 \leq j \leq m$.

Mechanism `ptr`

This mechanism performs a reverse DNS lookup for the corresponding domain of the sender's IP address, followed by a DNS lookup of type `A` or `AAAA` of the corresponding returned domain name to match against the sender's IP address. Since this mechanism requires the sender's IP address, which is not known until the time of validation, nodes and edges are created as placeholders only. For policy $u \in V$, with a `ptr` mechanism, assuming a sender's IP address v:

1. v is added to the graph with the creation of edge *(u, v)*;
2. a single node, w, is created to represent the `A` and `AAAA` record lookup of v, and it is connected to the graph using edge *(u, w)*; and
3. two nodes, x_1 and x_2, are added to the graph, labeled with arbitrary IPv4 and IPv6 address, connected by edges (w, x_1) and (w, x_2) respectively.

*Mechanism **`include`** and Modifier **`redirect`***

These two mechanisms instruct the SPF validator to obtain the SPF policy of the reference domain. The `redirect` modifier, similar to `include` mechanism, was created to allow an organization to apply the same SPF policy across multiple domains. Thus, it will be handled like the `include` mechanism for the purpose of building G_d. For policy $u \in V$, with an `include` mechanism or `redirect` modifier having value v, v is added to the graph with the creation of edge *(u, v)*.

4.1.2 Generating G_d

We developed software to take an SPF policy from the DNS and construct a tree following the model described in this section Our software was developed in Python and used the pyspf and pygraphviz libraries. In the near future, we plan to make the code available under an open source license.

We present two examples. One is a contrived SPF policy, `example.com`, and the other is the live policy for the domain name `idea.rs`. We refer to these examples when describing the metrics presented hereafter. The policies and graphs, $SPF_{example.com}$, $SPF_{idea.rs}$, $G_{example.com}$ and $G_{idea.rs}$ are shown in Table 4.1, Table 4.2, Figure 4.1, Figure 4.2 respectively.

As seen in Figure 4.1, there are four subtrees in $G_{example.com}$ stemming from its root. Each corresponds to a term of $SPF_{example.com}$: `a`, `mx`, `include`, and `ptr`. Theoretically, each subtree constitutes a potential path that an SPF validator might follow for its validation. Each edge represents the action triggered by an SPF term, and each node — except the root and the leaf nodes represents a DNS query being issued. These DNS lookups are: 1) a DNS query for the `A` record of domain name `example.com`; 2) a DNS query for the `MX` record of `example.com`; 3) a DNS query for the `A/AAAA` record of domain name `mail20.example.com`; 4) a DNS query for the SPF policy of domain name `example2.com`; 5) a DNS query for the SPF policy of domain name `example3.com`; 6) a placeholder representing the reverse DNS query for the sender's IP address; and 7) a placeholder representing the DNS query for the `A/AAAA` record corresponding to the returned reverse DNS response. Finally, each leaf node of $G_{example.com}$ contains the authorized hosts of $SPF_{example.com}$.

There are three subtrees in $G_{idea.rs}$ (Figure 4.2) stemming from the root node. Each corresponds to one of the three terms in $SPF_{example.com}$, namely the `include` and two `ip4` mechanism of $SPF_{example.com}$. The DNS lookups for $SPF_{idea.rs}$ represented in the nodes of $G_{idea.rs}$ are: 1) a DNS query for the SPF policy of domain name `agrokor.hr`; 2) a DNS query for the `MX` record of the domain name `agrokor.hr`; 3) a DNS query for the `A/AAAA` records of domain name `muhlo.agrokor.hr`; and 4) a DNS query for the `A`/`AAAA` record of domain name `lobel.agrokor.hr`.

Domains	**Type**	**Value**
example.com	TXT	“v=spf1 a mx include: example2.com ptr -all”
example.com	A	192.0.2.10
example.com	AAAA	2001:db8::10
example.com	MX	mail20.example.com
mail20.example.com	A	192.0.2.20
mail20.example.com	AAAA	2001:db8::20
example2.com	TXT	“v=spf1 redirect= example3.com all”
example3.com	TXT	“v=spf1 ip4:192.0.2.30 all”

Table 4.1: SPF data of `example.com`

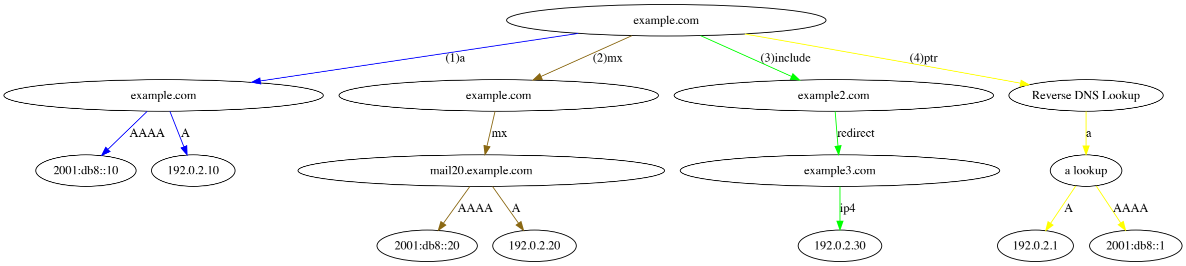


Figure 4.1: $G_{example.com}$

Domains	**Type**	**Value**
idea.rs	TXT	“v=spf1 include:agrokor.hr ip4:194.126.214.242 ip4:213.186.0.5 -all”
agrokor.hr	TXT	“v=spf1 +mx ip4:194.126.214.229 ip4:194.126.214.197 ip4:194.126.214.232 ip4:213.186.0.5 -all”
agrokor.hr	MX	muhlo.agrokor.hr, lobel.agrokor.hr
muhlo.agrokor.hr	A	194.126.214.160
muhlo.agrokor.hr	AAAA	Not available
lobel.agrokor.hr	A	194.126.214.161
lobel.agrokor.hr	AAAA	Not available

Table 4.2: SPF data of `idea.rs`

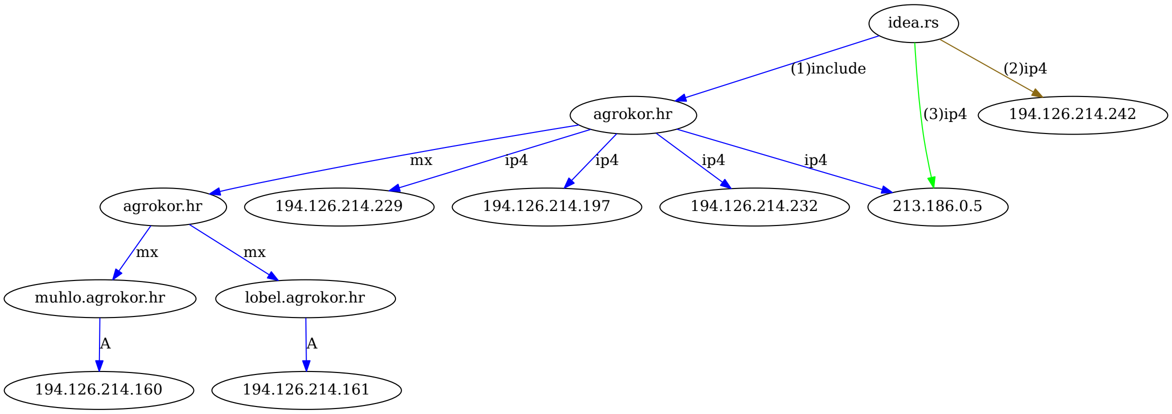


Figure 4.2: $G_{idea.rs}$

4.2 Metrics

As observed in Figure 4.1, the model not only provides the user with a visual representation but captures the essence of its SPF policy. Thus, we use G_d to derive meaningful metrics to help us quantify SPF policy complexity by answering questions such as: 1) what is the total number of DNS lookups required by the policy; 2) how many IP addresses are listed; and 3) what is the prevalence of SPF policy outsourcing. Answering these questions will aid in quantifying the level of control an organization has in its SPF policy, quantifying the complexity of an organization's email infrastructure, and assessing the effectiveness of SPF deployment as a function of its complexity.

4.2.1 IP Nodes

The leaf nodes of G_d constitute all the authorized IP addresses of SPF_d against which an SPF validator can match sender IP addresses. A diverse IP address range or a large pool of the IP address space might imply higher administrative costs due to the time and effort put in by the network administrator to enumerate and maintain the IP addresses.

4.2.2 Minimum DNS Lookups (min DNS)

The SPF specification indicates that an SPF validator should evaluate the terms of a given SPF policy in order from left to right [RFC 7208]. For many validators, this involves performing DNS lookups on demand, as the policy is evaluated. Consider the DNS lookups required for $\mathit{SPF}_{idea.rs}$ (refer to Figure 4.3 with nodes highlighted in red). An RFC 7208-compliant SPF validator will: 1) evaluate the first term of $\mathit{SPF}_{idea.rs}$, the `include mechanism`, and issue a DNS query for the SPF policy of `agrokor.hr`; 2) evaluate the first term of $\mathit{SPF}_{agrokor.hr}$, the `mx` mechanism, and issue a DNS query for the `MX` record for `agrokor.hr`; and 3) evaluate the first returned `MX` record of `agrokor.hr` and issue a DNS query for the `A` record corresponding to `muhlo.agrokor.hr`, one of the `MX` records for `agrokor.hr`. Thus, the total number of DNS lookups just to evaluate the sender's IP address against the first authorized host in $\mathit{SPF}_{idea.rs}$ is 3. We refer to this as the minimum DNS lookups.

The minimum DNS lookups is derived by traversing the left-most nodes of G_d, subtracting the root and left-most leaf nodes, as shown in Eq 4.2

$$G_d\ min\ DNS = |G_d \text{ left-most } nodes| - 1(root\ node) - |G_d \text{ left-most } leaf\ nodes| \tag{4.2}$$

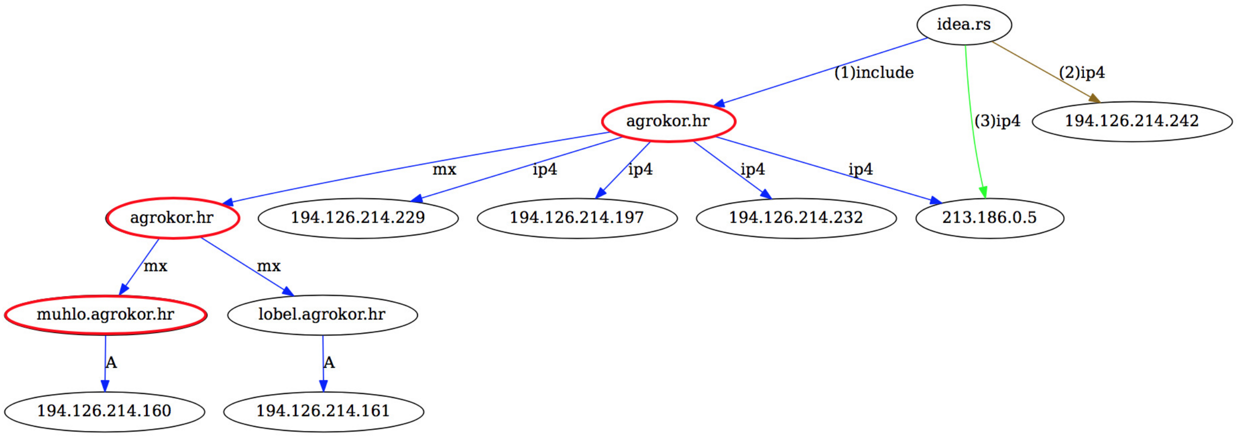


Figure 4.3: $G_{idea.rs}$

4.2.3 Maximum DNS Lookups (max DNS)

This metric captures all the possible DNS lookups that an SPF validator might need to issue for a given SPF_d. This is the upper bound of DNS lookups, supposing the sender's IP address never matches any of the authorized IP addresses. For example, we consider all the possible DNS lookups by an SPF validator for $SPF_{idea.rs}$ (refer to Figure 4.4 with nodes highlighted in red): 1) a DNS query for the SPF policy of `agrokor.hr`, 2) a DNS query for the `MX` record of `agrokor.hr`, 3) a DNS query for the `A` record of `muhlo.agrokor.hr`, 4) a DNS query for the `A` record of `lobel.agrokor.hr`. Thus, the total number of DNS lookups of $SPF_{idea.rs}$ is 4.

To obtain the maximum DNS lookups, we count the total nodes in a G_d, excluding the root node and the leaf nodes, as shown in Eq 4.3

$$G_d\ max\ DNS = |G_d\ nodes| - 1(rootnode) - |G_d\ leaf\ nodes| \tag{4.3}$$

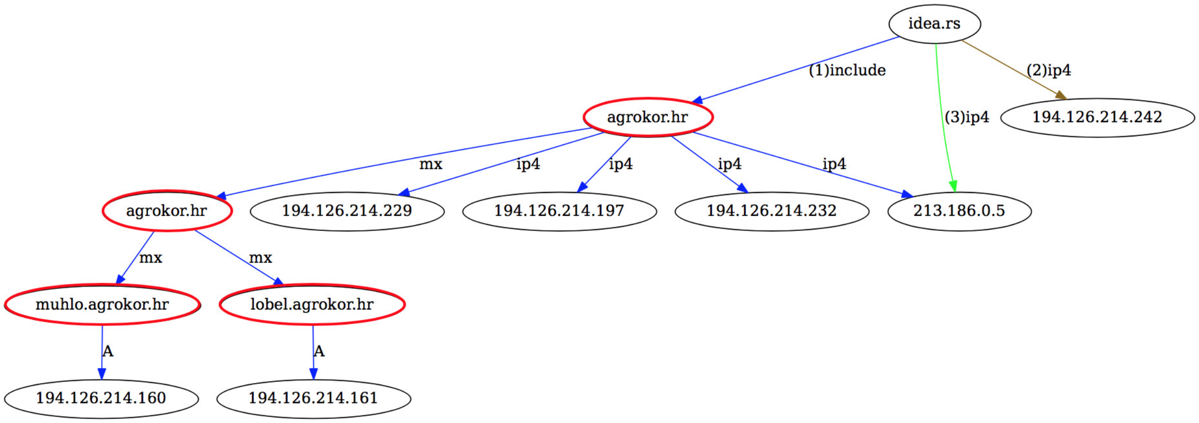


Figure 4.4: $G_{idea.rs}$

4.2.4 Normalized DNS Lookup Range

We can quantify the efficiency of SPF_d by analyzing the difference between the min DNS and max DNS of G_d, as shown in Eq 4.4.

$$G_d\ DNS\ Lookup\ Range = \begin{cases} \frac{G_d\ Max\ DNS - G_d\ Min\ DNS}{G_d\ Max\ DNS} & \text{if } G_d\ Max\ DNS > 0 \\ 0 & \text{otherwise} \end{cases} \tag{4.4}$$

Any range other than zero simply implies there is a difference between the min DNS and the max DNS. However, a normalized DNS lookup range of zero could mean that either the number of DNS lookups incurred by the SPF validator is the same for both min DNS and max DNS, or that no DNS lookups are required for validation. Hence, any SPF policy that can be evaluated without a single DNS lookup can be viewed as a more efficient SPF policy.

4.2.5 Duplicate IP Nodes (DIP)

It is possible that a single IP address might appear more than once in a policy, after expansion. The number of duplicate IP nodes considers the possibility of multiple instances of the same IP addresses in SPF_d. Duplicate authorized hosts for a given SPF_d are found by tracking any two or more leaf nodes with the same IP address (refer to Figure 4.5 with nodes highlighted in red). Finding any duplicate IP nodes found within SPF_d provides an opportunity for an organization to simplify its SPF policy. The simplifying process can include removing any unnecessary term(s) or deleting a reference domain that evaluated to the same IP address, thus reducing policy complexity and improving policy efficiency.

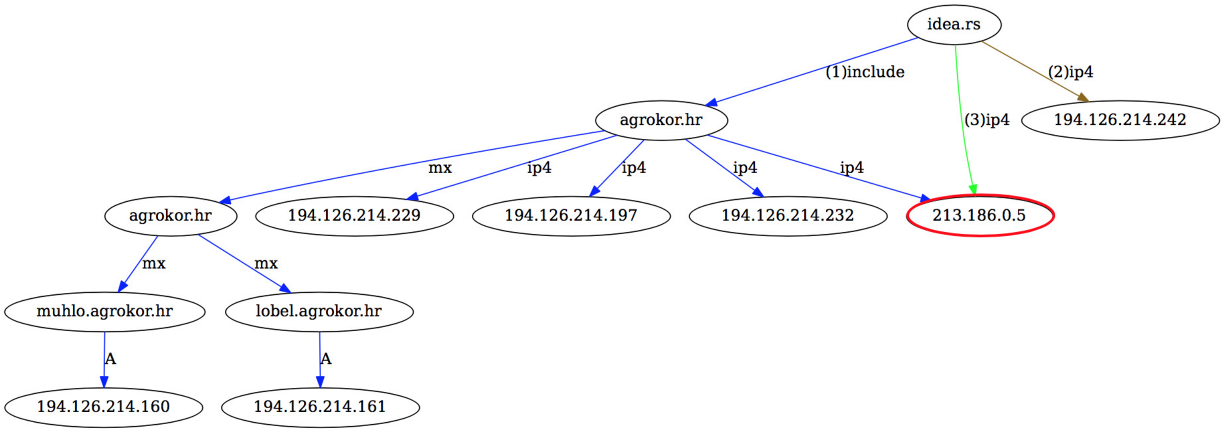


Figure 4.5: $G_{idea.rs}$

4.2.6 Third-Party Organizations

The owner of the domain, usually an organization, can either choose to setup its own email servers to send and receive email, outsource to a third-party email provider (third-party organization) to send and receive email on their behalf, or employ a mixture of both. In an SPF policy, the notion of outsourcing is accomplished by referencing a third-party organization as the reference domain of the `include` mechanism or `redirect` modifier.

To identify the organizations that contribute to the policy of a given domain, we employ a heuristic approach. This approach is based on the assumption that an organization that administers multiple domains will consistently use the same suffix in the email address for their administrative contact. The email address is found in the RNAME (responsible name) field of the start of authority (SOA) record for the domain. The SOA record contains administrative details associated with a given domain. To find the total number of unique third-party organizations in G_d, we: 1) retrieve the SOA DNS record and extract the email suffix of the RNAME for the base domain as well as each reference domain of the `include` mechanism or `redirect` modifier; and 2) count the number of unique email suffixes extracted. We then subtract one from the total, to account for the base organization domain in G_d.

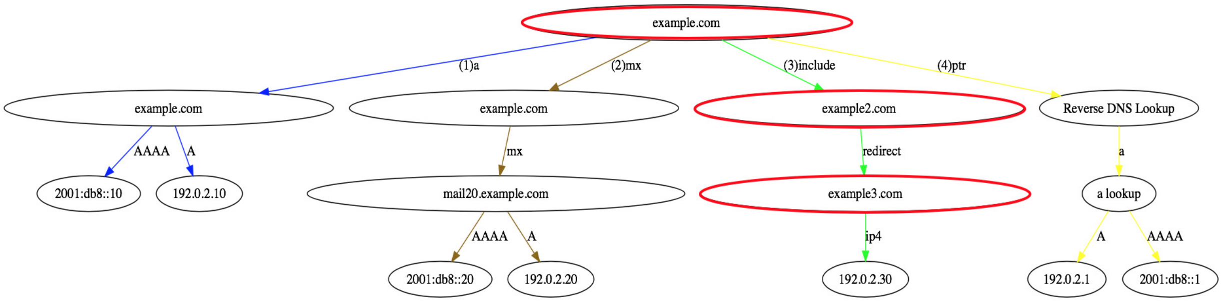


Figure 4.6: $G_{example.com}$

We illustrate this using $G_{example.com}$ (Figure 4.6) as an example. The nodes corresponding to the `include` mechanism or `redirect` modifier are highlighted in red, as is the root node. The number of third-party organizations is calculated as follows:

- The email suffix of the RNAME for the base domain `example.com` is `example.com`.
- The email suffix of the RNAME for the reference domain of the include mechanism `example2.com` is `example2.com`.
- The email suffix of the RNAME for the reference domain of the `include` mechanism `example3.com` is `example3.com`.
- The third-party organization count is 2 because there are 3 unique email suffixes extracted less one to account for the base organization domain.

4.2.7 Percentage of Outsourcing

While understanding the number of third-party organizations contributing to a policy is important, so is understanding the extent to which they contribute. The percentage of outsourcing works in conjunction with the third-party organization count and is used to reveal the total percentage of third-party organizations authorized hosts contributing to SPF_d.

To calculate the percentage of outsourcing in G_d for a given leaf node we: 1) identify the nearest ancestor node in the hierarchy representing an `include` mechanism or `redirect` modifier, if any; 2) determine whether the reference domain for the nearest ancestor node is a third-party organization; and 3) increment the outsourcing count if the returned status is true. The last step for this process is to tally the percentage of outsourcing after all G_d's leaf nodes have been processed, i.e., by dividing the total number of leaf nodes under any third-party organization by the total number of leaf nodes.

Using $G_{example.com}$ (Figure 4.7) as an example, we calculate its percentage of outsourcing as follows:

- Given the leaf node of IP 2001:db8::10, there is no nearest ancestor node in its hierarchy representing an `include` mechanism or `redirect` modifier. Thus, the next leaf node is considered.
- Given the leaf node of IP 192.0.2.30, the nearest ancestor node in its hierarchy representing an include mechanism or redirect modifier is the reference domain, `example3.com` of the `include` mechanism.

  1. The third-party organization status of `example3.com` returns true, i.e., that the organization associated with `example3.com` is different than that of `example.com`.
  2. Outsourcing count is increase by one.

- The percentage of outsourcing after all the G_d's leaf nodes have been processed is one out of seven or 14.29%.

As it pertains to policy maintenance, we expect that complete outsourcing of email services to a third-party organization might result in a lower complexity. This is because the setup and maintenance of any software and/or hardware related to the email infrastructure has been delegated to a professional third-party email provider. On the other hand, a mixture of in-house and outsourced email services employed by an organization might increase policy complexity due to managing different mailing systems with a single SPF policy.

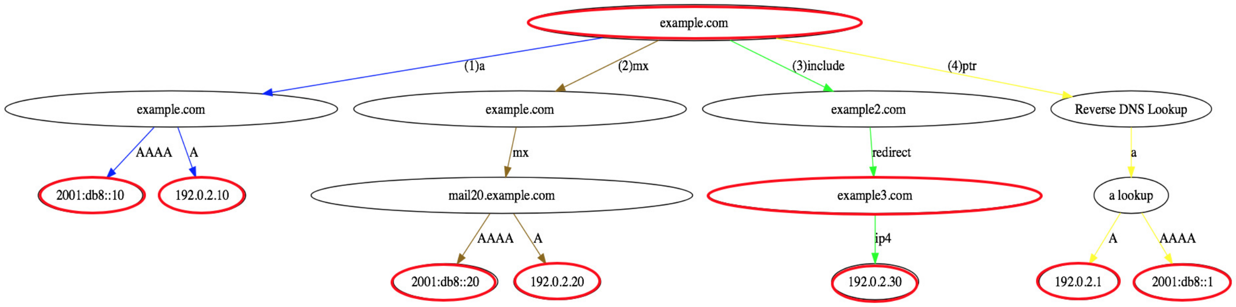


Figure 4.7: $G_{example.com}$

Chapter 5

Measurement Study

In order to assess the complexity of SPF policies currently deployed, we applied our metrics to the top 525,166 domains of Alexa top sites as of March 2018. Refer to table 5.1, out of the domains we analyzed, only 46.35% contained valid SPF records that allow email. The rest of the domains either had no SPF policy, had published their SPF policy incorrectly (e.g., had more than one SPF record per domain, published with a type SPF instead of a type `TXT`, or contained syntax errors that prevented the record from being parsed), or had a valid SPF configuration of `"v=spf1 -all"` (no email should be sent from the domains). In addition, there were 18 valid SPF records that went into a recursive loop due to careless usage of the `include` mechanism. Refer to Table 5.1 for the percentage breakdown.

Short Description	**Breakdown (%)**
Valid SPF Policy	46.345%
No SPF policy	46.355%
SPF policy with errors	6.892%
Valid SPF policy ”v=spf1 -all”	0.404%
Valid SPF policy with recursive loop	0.004%

Table 5.1: Overall SPF Policy Status

5.1 IP Nodes

Percentile	IPv4	IPv6	Ratio (IPv4 / IPv6)
25	4	0	Undefined
50	14	1	14
75	30	5	6
95	60	11	5.5
100	1830	1819	1.006

Table 5.2: Percentile of IPv4 and IPv6 nodes of SPF policies for Alexa Top Sites

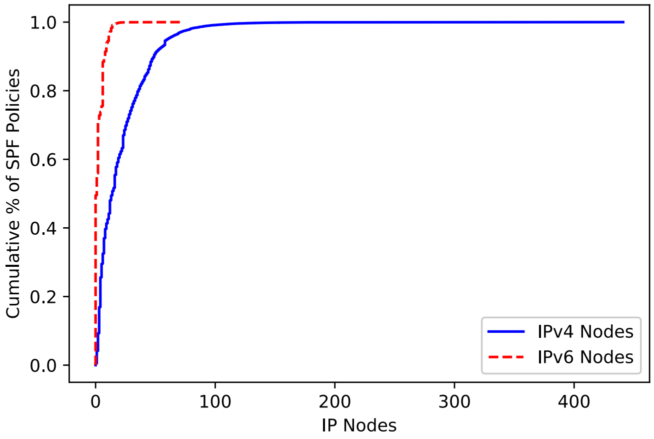
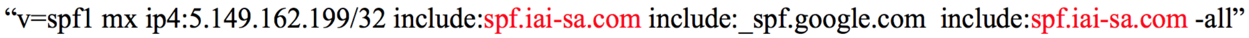


Figure 5.1: Cumulative distribution function(CDF) of IPv4 and IPv6 nodes for SPF policies of Alexa Top Sites

Half of the policies we analyzed included more than 14 IPv4 nodes, and 25% included more than 30 IPv4 nodes. In contrast to the number of IPv4 nodes, half of the policies included more than one IPv6 node, while 25% included more than 5 IPv6 nodes. Thus, there are, on average, more IPv4 than IPv6 nodes. Therefore, we might infer that the IPv6 adoption rate among the evaluated policies is low when compared to IPv4 adoption.

The amount of IPv4 and IPv6 nodes at the 100th percentile, 1830 and 1819 respectively, is more than 30 times greater than the amount of IPv4 and IPv6 nodes in the 95th percentile. The high number of IPv4 and IPv6 nodes at the 100th percentile is caused by a single policy, $SPF_{modbis.pl}$ (see Figure 5.2). $SPF_{modbis.pl}$ uses the reference domain `spf.iai-sa.com` twice for its two include mechanisms. A single use of the reference domain spf.iai-sa.com resolved to over nine hundred unique IPv4 and IPv6 addresses, and calling it two times doubles the IPv4 and IPv6 addresses to over a thousand. This resulted in an exceptionally high number of IPv4 and IPv6 nodes, as reflected in Table 5.2.



“v=spf1 mx ip4:5.149.162.199/32 include:spf.iai-sa.com include:_spf.google.com include:spf.iai-sa.com -all”

Figure 5.2: $SPF_{modbis.pl}$

5.2 Third-Party Organization and Percentage of Outsourcing

Third-Party Organizations	Percentage of SPF Policies	Third-Party Organizations	Percentage of SPF Policies
0	38.4445%	8	0.0308%
1	35.596%	9	0.0185%
2	12.9266%	10	0.0037%
3	8.1334%	11	0.0033%
4	3.3428%	12	0.00016%
5	1.0695%	13	0.0004%
6	0.332%	14	0.0004%
7	0.0966%		

Table 5.3: Breakdown of SPF policies for Alexa Top Sites by number of third-party organizations

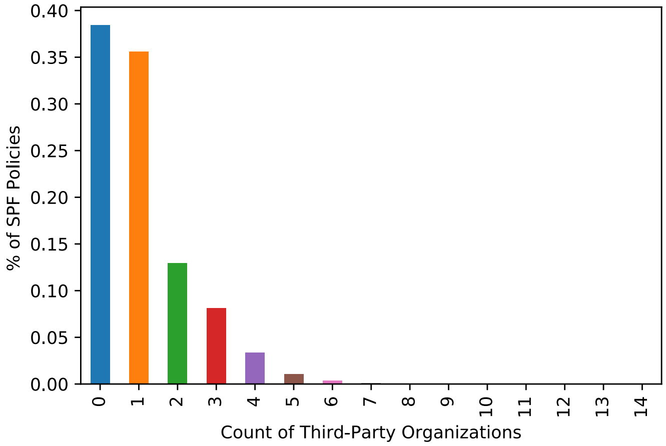


Figure 5.3: Histogram showing the percentage of SPF policies for Alexa Top Sites for each count of third-party organizations

Table 5.3 shows the percentage breakdown of the number of SPF policies based on the number of third-party organizations. Figure 5.4 shows a plot of the cumulative distribution function (CDF) of the percentage of outsourcing.

We observe that nearly 62% of the analyzed policies employ some degree of outsourcing. We also observe that approximately 23% of domains exhibit complete outsourcing of their policies (i.e., percentage of outsourcing = 1.0), 38.45% of domains exhibit no outsourcing of their policies (i.e., percentage of outsourcing = 0.0), and the rest of the domains outsource their policies partially (i.e., percentage of outsourcing is greater than 0 and less than 1). A

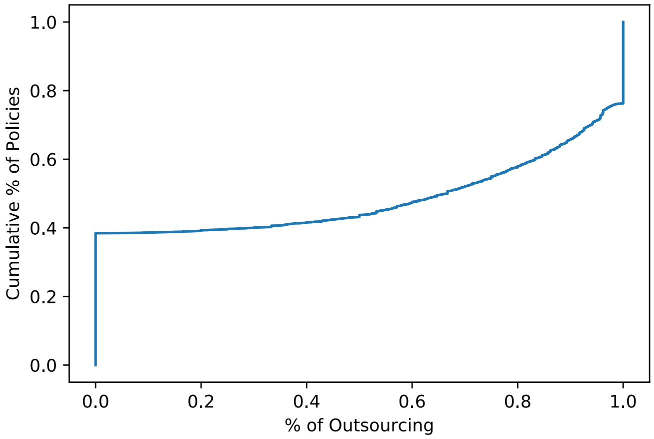


Figure 5.4: CDF of percentage of outsourcing found in SPF policies for Alexa Top Sites

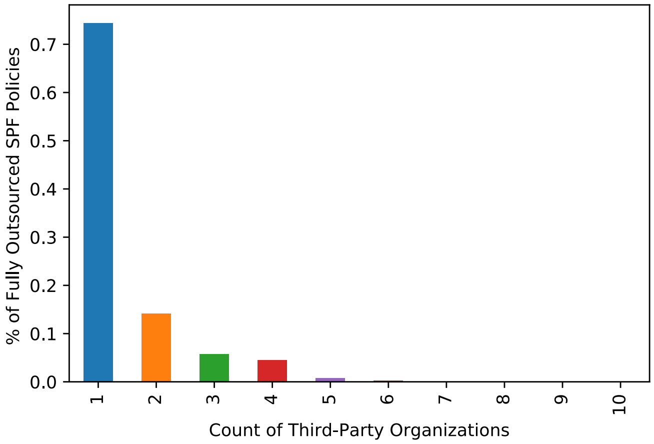


Figure 5.5: Histogram showing the percentage of SPF policies for Alexa Top Sites for each count of third-party organizations, for fully outsourced policies

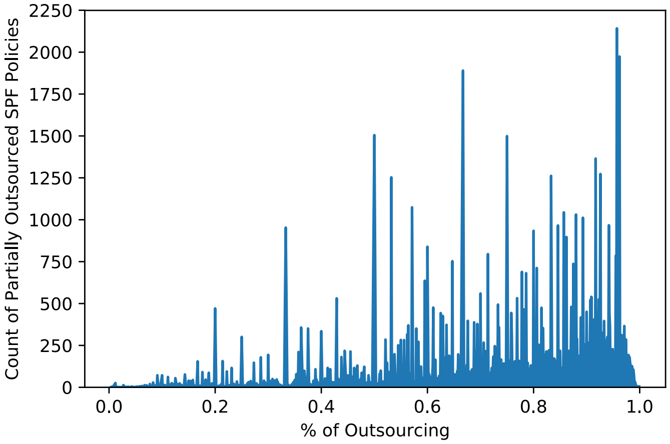


Figure 5.6: Area chart showing the count of SPF policies for Alexa Top Sites over the percentage of sourcing, for partially outsourced policies

high third-party organization count in a given SPF policy does not necessary imply that its policy is fully outsourced. For example, one of the SPF policies we analyzed has a third-party organization count of 14, but its outsourcing percentage is 97.1%. As a matter of fact, we observe that roughly 75% of domains with fully outsourced policies have only a single third-party organization (see Figure 5.5).

Figure 5.6 shows an area chart of the outsourcing percentage of domains that exhibit partial outsourcing. The mean and median for the percentages of outsourcing are 0.739 and 0.79, respectively. Additionally, for about 85% of partially outsourced policies, the majority of derived IP addresses (at least 52%) are associated with third-party organizations.

We observe that most of the domains (about 62%) engage in some sort of outsourcing as reflected in their SPF policies. By correlating the percentage of outsourcing with other metrics such as DIP, min DNS, max DNS, and normalized DNS lookup range, we can study the subtleties of how outsourcing might impact DIP, min DNS, max DNS, and normalized DNS lookup range.

5.3 DIP

Percentile	Overall DIP (Category 1)	DIP of SPF Policies with No Outsourcing (Category 2)	DIP of the Partially Outsourced SPF Policies (Category 3)	DIP of the Fully Outsourced SPF Policies (Category 4)
25	0	0	0	0
50	0	1	0	0
75	1	1	1	0
95	3	2	8	0
100	1808	57	1808	83

Table 5.4: Percentile of DIP of SPF policies for Alexa Top Sites

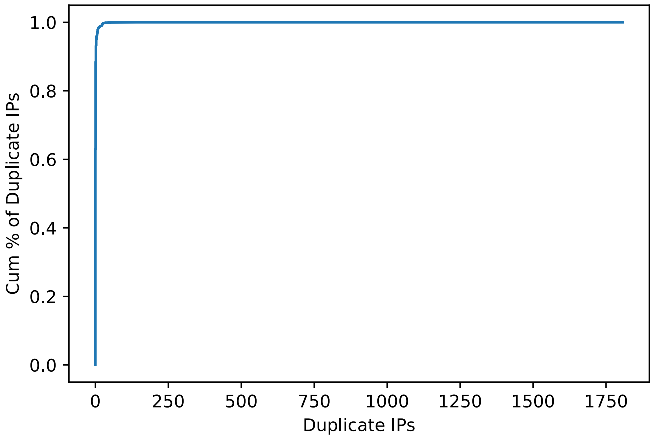


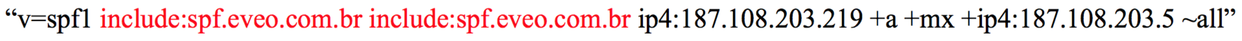
Figure 5.7: CDF of DIP for SPF policies of Alexa Top Sites

There are 4 categories presented in Table 5.4 Category 1 contains the statistical measure of DIP, as percentiles, of all evaluated SPF policies. Category 2 includes only the DIP of SPF policies with no outsourcing. Category 3 represents the DIP of SPF policies that employed partial outsourcing. Category 4 consists of DIP with fully outsourced SPF policies.

Figure 5.7 shows the CDF of the overall number of DIP found in each policy. There is a huge disparity in the number of DIP between the 95th and the 100th percentile policies. However, by correlating the number of DIPs with the percentage of outsourcing, we realized that 1) the DIP count of 1808 belongs to an SPF policy with partially outsourced configuration and 2) the max DIP count for categories 2 and 4 are respectively, 28 and 83 times higher than the rest of the 95th percentile policies in their corresponding categories.

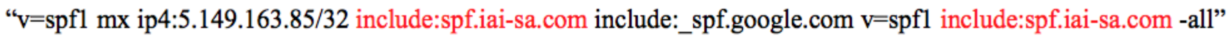
$SPF_{redetiradentes.com.br}$, $SPF_{epinokio.pl}$ and $SPF_{reliancemoney.com}$ are the originator policies for the 100th percentile DIP count in categories 2, 3 and 4 respectively (see Figure 5.8, 5.9 and 5.10). The reason for such a high count in each case is the use of two `include` mechanisms pointing to the same reference domain in both $SPF_{redetiradentes.com.br}$ and $SPF_{epinokio.pl}$. $SPF_{reliancemoney.com}$, on the other hand, uses two `include` mechanisms, but each `include` mechanism points to a unique reference domain, and both resolve to the same set of IP addresses.

To resolve DIPs, an SPF administrator might choose to remove one of the redundant `include` mechanisms or take other appropriate actions, as listed in section 4.2.5.



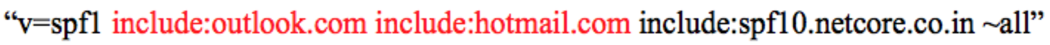
“v=spf1 include:spf.eveo.com.br include:spf.eveo.com.br ip4:187.108.203.219 +a +mx +ip4:187.108.203.5 ~all”

Figure 5.8: $SPF_{redetiradentes.com.br}$



“v=spf1 mx ip4:5.149.163.85/32 include:spf.iai-sa.com include:_spf.google.com v=spf1 include:spf.iai-sa.com -all”

Figure 5.9: $SPF_{epinokio.pl}$



“v=spf1 include:outlook.com include:hotmail.com include:spf10.netcore.co.in ~all”

Figure 5.10: $SPF_{reliancemoney.com}$

5.4 Min DNS

Percentile	Min DNS of SPF Policies with No Outsourcing (Category 1)	Min DNS of the Partially Out-sourced SPF Policies (Category 2)	Min DNS of the Fully Out-sourced SPF Policies (Category 3)
25	0	0	1
50	1	1	2
75	1	2	2
95	2	3	3
100	9	5	5

Table 5.5: Percentile of Min DNS of SPF policies for Alexa Top Sites

Typically, a min DNS of zero denotes the use of either `ip4` or `ip6` as the left most term in the SPF policy. This is usually an indication of a more efficient SPF policy since it does not require any DNS lookups for the derivation of IP addresses. On the other hand, a fully outsourced policy should have at least a min DNS of 1. To qualify as a fully outsourced policy: 1) all the terms in the SPF policy should either be an `include` mechanism or a `redirect` modifier; 2) its reference domain is associated with a third-party organization; and 3) all its authorized hosts stem from the third-party organizations.

In categories 1 and 2 of Table 5.5, we observe that 25% of the policies do not require any DNS lookups for IP address derivation, while 95% of the policies are guaranteed to obtain an IP address to match against within two DNS lookups. However, there is one policy, $SPF_{hubspot.com}$, in category 1 with a min DNS that is four times higher than the 95th percentile policies in category 1. This was caused by the use of `include` mechanism, as the left-most term in the base domain SPF policy, as well as in all of its subsequent reference domains' SPF policies.

There were nine SPF policies under category 3 with a min DNS of zero, deviating from the established norm of having at least a min DNS of 1 for a full outsource policy. These policies exhibited this behavior because they used the `include` mechanism as their left-most term in the policy (see Figure 5.11). However, the SPF policy of the reference domain of the `include` mechanism had only one term in its entire configuration: `"v=spf1 -all"`, meaning no email should be sent from the domain. Such unexpected configuration caused the program to return from that particular subtree, and a leaf node was generated to represent the `include` mechanism.

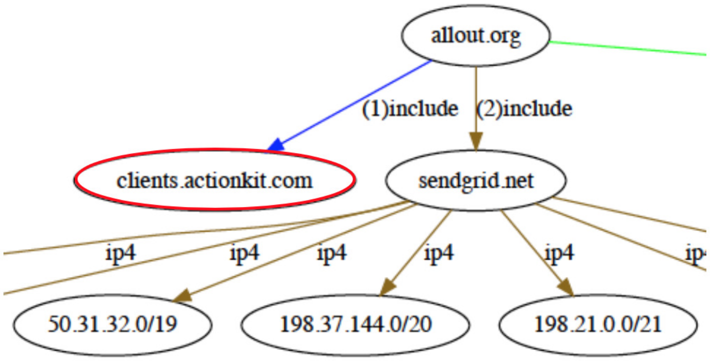


Figure 5.11: $G_{allout.org}$

5.5 Max DNS

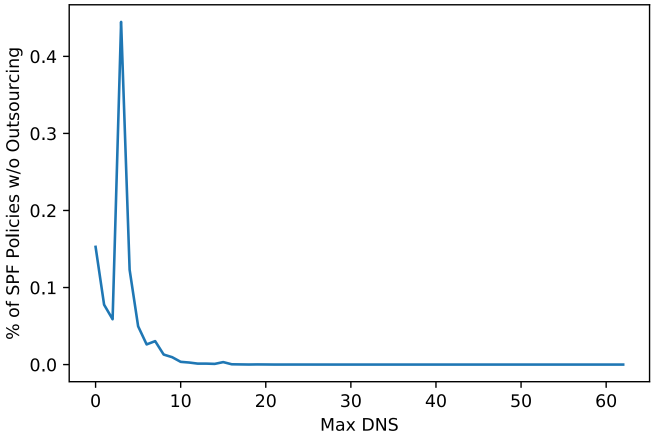


Figure 5.12: Line chart showing the percentage of SPF policies for Alexa Top Sites for different value of max DNS, considering only policies with no outsourcing

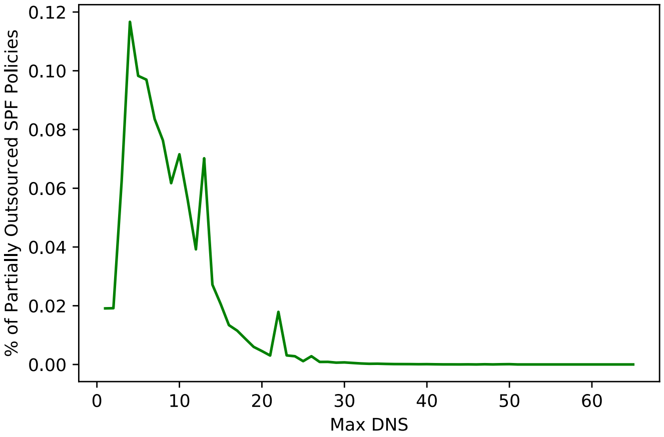


Figure 5.13: Line chart showing the percentage of SPF policies for Alexa Top Sites for different value of max DNS, considering only partially outsourced policies

Max DNS tracks the maximum possible DNS lookups that a SPF validator might need to issue for any given policy. About 95% of the SPF policies exhibiting no outsourcing or complete outsourcing have a max DNS that is less than or equal to 7 and 9, respectively (see Figure 5.12, 5.14). This means a majority of the SPF policies with no outsourcing or complete outsourcing are being resource conscious by minimizing their policies' impact on

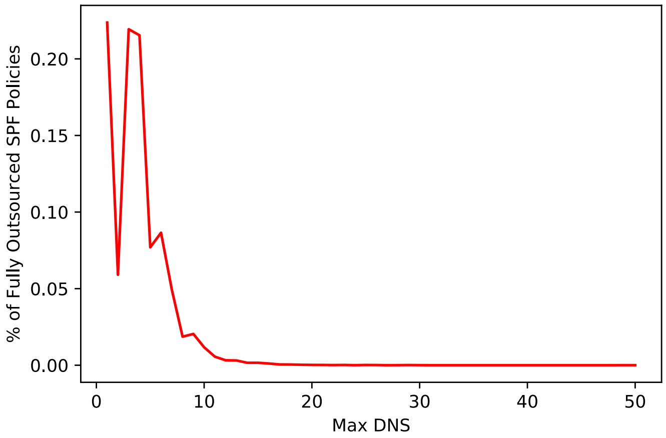


Figure 5.14: Line chart showing the percentage of SPF policies for Alexa Top Sites for different value of max DNS, considering only fully outsourced policies

the DNS infrastructure. This can be done by limiting the maximum number of DNS queries that an SPF validator might need to issue during evaluation.

On the other hand, max DNS is more sparse for the partially outsourced SPF policies (see Figure 5.13), with only 50% of those policies having a max DNS that is less than or equal to 8. To minimize the impact of SPF policies on the DNS resources, any organization that partially outsourced its email infrastructure must consider the potential DNS lookups, both those associated with its own organization and those associated with third-party organizations, when designing its SPF policy.

5.6 Normalized DNS Lookup Range

Normalized DNS Lookup Range	SPF Policies with No Outsourcing(Category 1)	Partially Outsourced SPF Policies (Category 2)	Fully Outsourced SPF Policies(Category 3)
0	28.47%	0.49%	25.99%

Table 5.6: Normalized DNS lookup range of zero of SPF policies for Alexa Top Sites

As mentioned in section 4.2.4, there are two scenarios where a normalized DNS lookup range of zero can occur. In the first scenario, the min DNS is equal to the max DNS, and the min DNS and max DNS are both greater than zero. In the second, the min DNS is equal to the max DNS, but the min DNS and max DNS are both zero. This can only occur if all the terms in the policy consist only of `ip4` and/or `ip6`. Refer to figure 5.6.2 for a second scenario example. The second scenario is considered to be more efficient since the SPF policies can be evaluated by an SPF validator without any DNS queries. Thus, any organization that wishes to maximize its SPF policy efficiency should only use `ip4` and/or `ip6` term to construct their SPF policy.

From Table 5.6, we observe that fewer than 0.5% of the SPF policies from the partially outsourced category have a normalized DNS lookup range of zero. Whereas 28.47% of policies with on outsourcing and 25.99% of fully outsource policies have a normalized DNS lookup range of zero.

Of the 28.47% of SPF policies that don't outsource and have a normalized DNS lookup range of zero, 46.33% of them fall in the first scenario — that is, the min DNS and max DNS are both greater than zero. The remaining 53.67% fall under the more efficient second scenario. This is very different compared to fully outsourced and partially outsourced SPF policies, where 100% of policies with a normalized DNS lookup range of zero fall under the first scenario.

5.7 Historical Trends

As mentioned in Section 4, SPF policies must be maintained continuously to keep up with organizational changes. In this section we seek to understand the general SPF deployment trend through the use of historical SPF policies. To facilitate this study, the SPF data of top 100,000 of Alexa Top Sites was obtained through the use of Farsight Securitys DNSDB for the years 2011 to 2017. We calculated the yearly median and mean based on the metric of IP nodes, min DNS, max DNS, third-party organizations, and percentage of outsourcing into a time series graph to capture any changes in the policy over time.

5.7.1 IP Nodes

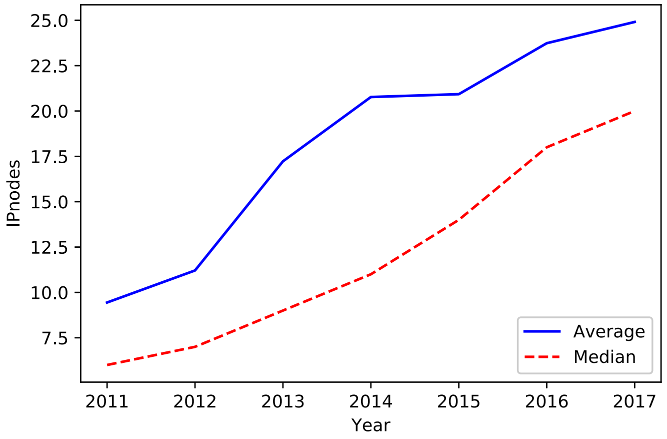


Figure 5.15: Time Plot of average and median number of IP Nodes in SPF policies for Alexa Top Sites from 2011-2017

Figure 5.15 shows the time plot of average and median IP nodes (both IPv4 and IPv6 nodes) of the yearly aggregated SPF policies. There has been an upward trend for both mean and median over the period of six years. This upward trend demonstrates: 1) an overall increase in the number of IP addresses; and 2) a growth in both the upper and lower 50th percentiles of the SPF policies. Thus, this implies that policies have become more complex over the analyzed time periods.

5.7.2 Min DNS and Max DNS

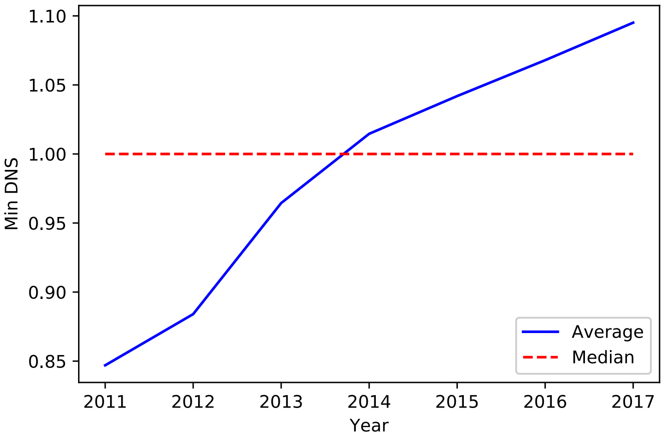


Figure 5.16: Time plot of average and median number of Min DNS in SPF policies for Alexa Top Sites from 2011-2017

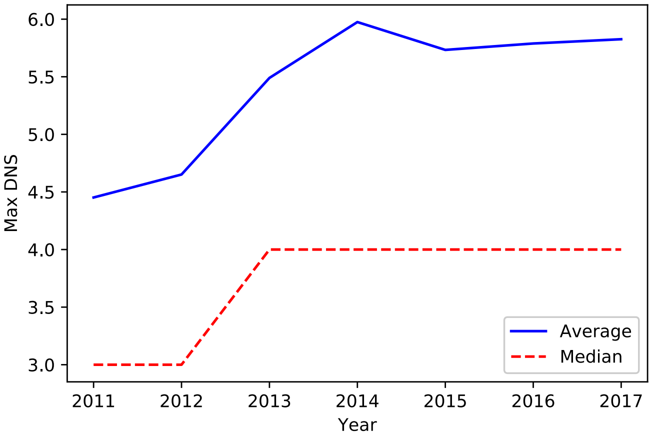


Figure 5.17: Time plot of average and median number of Max DNS in SPF policies for Alexa Top Sites from 2011-2017

Figure 5.16 shows the time plot of min DNS. While we observed an upward trend in the mean, the median remained constant over the six-year period. This indicates the increase is associated with the upper 50th percentiles of the policies.

Figure 5.17 shows the time plot of max DNS. There is some fluctuation in the mean, but the overall trend is upward. The mean is consistently higher than the median. Thus, the

SPF policies of the lower 50th percentile are consistently skewed toward the lower end of the max DNS.

5.7.3 Outsourcing

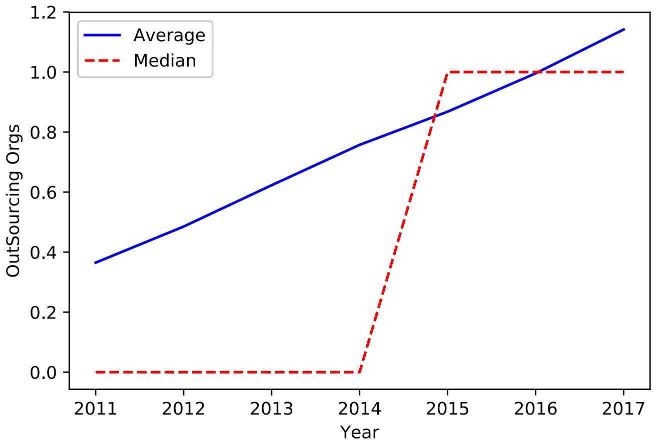


Figure 5.18: Time Plot of average and median number of count of third-party organizations in SPF policies for Alexa Top Sites from 2011-2017

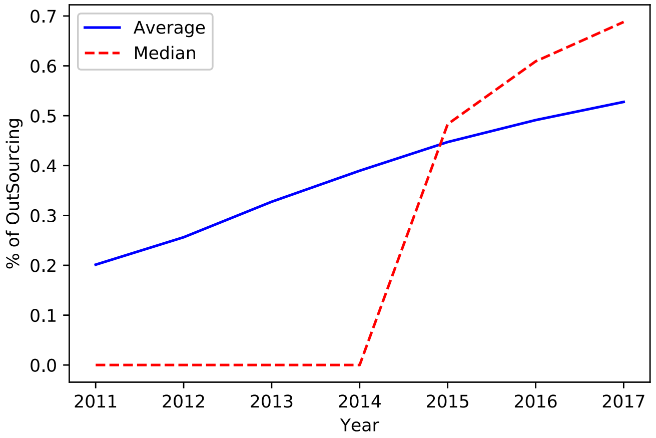


Figure 5.19: Time Plot of average and median number of percentage of outsourcing in SPF policies for Alexa Top Sites from 2011-2017

Overall, we saw an increasing trend for outsourcing (see figure 5.18 and 5.19). In 2011, only 30% of the SPF policies engage in outsourcing. However, that number increased to 64% in 2017 as more policies have embraced outsourcing.

Chapter 6

Conclusion and Future Work

In this thesis, we have developed a graph based model to capture the essence of an SPF policy and have provided organizations with tools and metrics to quantify the efficiency of their SPF deployment.

Through the use of metrics, we noticed the IPv6 adoption rate was significantly lower than IPv4. We also observed that about 62% of the policies we analyzed employed some degree of outsourcing on their email infrastructure. Also, 15.28% of SPF policies were considered to be efficient since the evaluation of a sender IP address can be done without issuing a single DNS lookup. We also investigated the organizational network complexity by studying historical trends using metrics to measure the growth of IP address space and the adoption of SPF policy outsourcing.

To improve the efficiency of an SPF policy, an organization can use metrics such as DIP to eliminate any duplication of IP address found within a SPF policy. With the help of min DNS and max DNS, a resource conscious organization can choose to reduce its load on the SPF validators and DNS servers by cutting back on the number of DNS lookups through a redesign of its SPF policy.

Future work includes expanding our model to more accurately handle cases such as calculating min DNS when there is an include directive that yields no IP addresses, as described in section 5.4. To expand the scope of our analysis, future work could include performing a thorough investigation of all the errors documented in RFC 7208. We could analyze the impact an error has on min DNS and perhaps shed some light on the most frequently made errors. Error analysis should explore the possibility of equipping an organization with the means to fix its SPF policy error(s), thus improving the quality of SPF deployment rate.

Additionally, a better understanding of the SPF validators′ behaviors can be done by conducting a measurement study against the proposed RFC 7208 standard. Such a measurement study might uncover differences between the proposed standard and the behaviors of an actual SPF validator (i.e. operational processing, evaluation, and error-handling of mechanisms and modifiers). This essentially equips an organization with knowledge of the common validation practices adopted by the SPF validators and aids the organization in designing an efficient, all-inclusive, anti-spoofing email policy to protect its domain from being spoofed.

We believe that the model and metrics herein presented can help organizations not only with maintaining the correctness of SPF policies, but also with identifying and minimizing complexity. We hope that these efforts will aid the network administrator to not only design an effective, but also an efficient SPF policy that uses resources carefully and achieves its purpose of protecting the domain from compromise.

References

[1] APWG phishing attack trends reports. https://apwg.org/resources/apwg-reports/. Accessed: 2017-03-01.

[2] Nguyen Tuan Anh, Tran Quang Anh, and Nguyen Xuen Thang. Spam filter based on dynamic sender policy framework. In *Knowledge and Systems Engineering (KSE), 2010 Second International Conference on*, pages 224–228. IEEE, 2010.

[3] Uchenna P Daniel Ani, Hongmei Mary He, and Ashutosh Tiwari. Human capability evaluation approach for cyber security in critical industrial infrastructure. In *Advances in Human Factors in Cybersecurity*, pages 169–182. Springer, 2016.

[4] J Brumfield. Verizons 2016 data breach investigations report finds cybercriminals are exploiting human nature. https://www.verizon.com/about/news/verizons-2016-data-breach-investigations-report-finds-cybercriminals-are-exploiting-human. Accessed: 2017-03-01.

[5] Gökhan Dalkılıç and Devrim Sipahi. Spam filtering with sender authentication network. *Computer Communications*, 98:72–79, 2017.

[6] Zakir Durumeric, David Adrian, Ariana Mirian, James Kasten, Elie Bursztein, Nicolas Lidzborski, Kurt Thomas, Vijay Eranti, Michael Bailey, and J Alex Halderman. Neither snow nor rain nor MITM...: An empirical analysis of email delivery security. In *Proceedings of the 2015 Internet Measurement Conference*, pages 27–39. ACM, 2015.

[7] Ian D Foster, Jon Larson, Max Masich, Alex C Snoeren, Stefan Savage, and Kirill Levchenko. Security by any other name: On the effectiveness of provider based email security. In *Proceedings of the 22nd ACM SIGSAC Conference on Computer and Communications Security*, pages 450–464. ACM, 2015.

[8] Stefan Görling. An overview of the sender policy framework (SPF) as an anti-phishing mechanism. *Internet Research*, 17(2):169–179, 2007.

[9] Fanglu Guo, Jiawu Chen, and Tzi-cker Chiueh. Spoof detection for preventing DoS attacks against DNS servers. In *Distributed Computing Systems, 2006. ICDCS 2006. 26th IEEE International Conference on*, pages 37–37. IEEE, 2006.

[10] Scott Kitterman. RFC 7208: Sender policy framework (SPF) for authorizing use of domains in email, version 1, April 2014.

[11] Sara Radicati and Q Hoang. Email statistics report 2011-2015. the Radicati Group, Inc. a technology market research firm. https://www.radicati.com/wp/wp-content/uploads/2011/05/Email-Statistics-Report-2011-2015-Executive-Summary.pdf, 2011. Accessed: 2017-03-01.

[12] Hossein Siadati, Sima Jafarikhah, and Markus Jakobsson. Traditional countermeasures to unwanted email. In *Understanding social engineering based scams*, pages 51–62. Springer, 2016.

[13] Mario Silic and Andrea Back. The dark side of social networking sites: Understanding phishing risks. *Computers in Human Behavior*, 60:35–43, 2016.

[14] Verizon RISK Team and R Team. 2015 data breach investigations report. https://iapp.org/media/pdf/resource_center/Verizon_data-breach-investigation-report-2015.pdf. Accessed: 2017-03-01.

[15] Iryna Yevseyeva, Vitor Basto-Fernandes, and José R Méndez. Survey on anti-spam single and multi-objective optimization. In *International Conference on ENTERprise Information Systems*, pages 120–129. Springer, 2011.