From Defensive Registration to Subdomain Protection: Evaluation of Email Anti-Spoofing Schemes for High-Profile Domains

Sourena Maroofi, Maciej Korczyński, Andrzej Duda
Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG, F-38000 Grenoble, France
Email: firstname.lastname@univ-grenoble-alpes.fr

***Abstract*—Sending forged emails by taking advantage of domain spoofing is a common technique used by attackers. The lack of appropriate email anti-spoofing schemes or their misconfiguration lead to successful phishing attacks or spam dissemination. In this paper, we evaluate the adoption of the SPF and DMARC security extensions by high-profile domains and analyze spoofing possibilities enabled by the absence or misconfigurations of their rules. The results show that for top 500 domains of 139 countries, the adoption rate of SPF and DMARC rules are 65.9% and 34.3%, respectively. For banking websites, we obtain almost the same results (64.9% and 35.9%) as for the TOP500 list. However, for defensively registered domains, the results are significantly higher especially in terms of published SPF records with 95.37% adoption and 40.1% for DMARC. We also, for the first time, investigate the problem of subdomains in the anti-spoofing techniques and their possible abuse to send forged emails. We show that even major companies such as Microsoft or ESET Security do not correctly configure the SPF rules, which leads to the possibility of mail spoofing. Based on the emulation of the SPF check function, we show that syntactically wrong SPF rules may break the trust-based authentication system of email service providers by allowing forged emails to land in the user inbox. Finally, to help in remediation, we have issued notifications to CSIRTs responsible for domains with misconfigured SPF records to engage them in the mitigation action.**

I. Introduction

Email spoofing consists of sending a message with a forged sender address and other parts of the email header so that it appears as sent from a legitimate source. Attackers commonly use this method to mislead the receivers, gain their trust, and eventually, achieve some malicious goals. Phishing and spam campaigns are examples of attacks that rely on email spoofing. Despite tremendous efforts deployed to mitigate this technique, it is still one of the most successful attacks responsible for significant damage. According to the Internet crime report [1], email spoofing costed US victims more than 1.2 billion dollars in 2018.

Email spoofing comes in two types. The first one consists of *compromising legitimate servers* and using their mail transfer agent to send spoofed emails to victims either by specifying a different 'Reply-to:' address or providing a phishing URL in the body of the message. The second type is *domain spoofing* in which attackers send emails on behalf of legitimate domains, e.g., a forged email from *account-security-noreply@accountprotection.microsoft.com* impersonating the Microsoft support team with a fake landing page looking alike a real Microsoft login page to steal user credentials [2]. In this paper, we investigate this last type of email spoofing.

The Simple Mail Transfer Protocol (SMTP) for email distribution does not provide support for preventing spoofing [3]. The system needs to rely on *security extensions* such as the Sender Policy Framework (SPF) [4], the DomainKeys Identified Mail (DKIM) [5], and Domain-based Message Authentication, Reporting, and Conformance (DMARC) [6] to authenticate the sender and decide what to do with suspicious mails. The extensions define a set of rules that specify who is allowed to send emails on behalf of a domain name and provide strategies for dealing with spoofed messages. Carefully implementing the extensions can completely mitigate the problem of domain spoofing. However, to be effective, both the domain owner and the mail transfer agent of the recipient should implement the extensions: the domain owner needs to correctly set SPF, DKIM, and DMARC rules, and the recipient has to authenticate incoming messages as well correctly implement verification of the SPF and DMARC rules.

In this paper, we evaluate the adoption of the SPF and DMARC security extensions[1] and analyze spoofing possibilities enabled by the absence or misconfigurations of their rules. While previous work already investigated the adoption of SPF and DMARC by the Alexa top-ranked one million domains [7], [8], we consider a different threat model in which attackers use subdomains (both existent and non-existent) for email spoofing. We also identify *defensively registered domains* and evaluate their adoption of email anti-spoofing schemes. We show that even if defensive registration can mitigate some types of attacks like *cybersquatting* and *brand abuse*, these domains need to be protected against domain spoofing as well.

More specifically, our contributions are as follows:

1) in a measurement campaign, we evaluate the adoption of SPF and DMARC by top 500 most popular domains of 139 countries including local businesses, national websites, local governments, and financial sectors,
2) we propose a method to find defensively registered domains for top-ranked websites and assess the extent of

[1]We do not analyze DKIM as it requires access to the selector tag in the email header (see RFC 6376 for more details), not publicly available.

978-3-903176-27-0

TABLE I
POSSIBLE RESULTS OF THE SPF **CHECK_HOST** FUNCTION AND THEIR DEFINITIONS.

Result	Definition	Recommended action
None	1) No valid domain name was extracted from the SMTP session. 2) No SPF record was retrieved from the domain name.	1) The action must be the same as the *Neutral* output.
Neutral	1) There is no definite assertion (authorized or not) about the sender.	1) Depends on the receiver's system.
Pass	1) Client is authorized to send emails with the given identity.	1) Whitelist the domain in terms of SPF.
Fail	1) Client is not authorized to send emails with the given identity.	1) Depends on the receiver's system. 2) Make decision based on the DMARC policy.
Softfail	1) Client is not authorized to send emails with the given identity. 2) No strong policy specified by the domain owner.	1) Receiver should not reject the message. 2) May mark the message as suspicious.
Temperror	1) A temporary error occurred during retrieving the SPF policy.	1) May defer the message. 2) May deliver the message and mark it.
Permerror	1) Parsing problem in published SPF.	1) May deliver the message and mark it.

their adoption of email security extensions,

3) we are the first to measure the extent of SPF and DMARC deployment by the subdomains of the top-ranked websites to gain better insight into how attackers can abuse subdomains to send spoofed emails,
4) we show that it is possible to send forged emails from non-existent subdomains when a DMARC rule is not strict enough regarding subdomains,
5) we also demonstrate how syntactically wrong SPF rules may break the trust-based authentication system of selected email service providers by allowing forged emails to land in the user inbox.

To remediate vulnerable SPF rules, we contact relevant Computer Security Incident Response Teams (CSIRTs) responsible for misconfigured domains and we measure the effectiveness of our notifications. To encourage reproducibility, we make our measurement data available upon request.

The rest of the paper is organized as follows. Section II provides background on SPF and DMARC. Section III specifies possible threat models and introduces our approach to generate the datasets and find defensively registered domains. Sections IV presents the analysis of the results for scanned domains and subdomains as well as for emulation of SPF rules. In Section V, we study the trust-based authentication issue and Section VI describes remediation. Finally, Section VII reviews related work and Section VIII concludes the paper.

II. BACKGROUND ON ANTI-SPOOFING TECHNIQUES

To understand the issue of email authentication better, we briefly explain the process of mail delivery. Figure 1 shows Bob (sender) who sends legitimate mails to Alice (receiver). Mallory (attacker) wants to send an email that impersonates Bob to Alice. Mallory and Bob use their respective servers (**mallory.com** and **bob.com**) to send mails. The Mail Delivery Agent (MDA) on the Alice server delivers two emails with the same sender address (**me@bob.com**) but coming from different IP addresses (assuming there is no spam filtering involved). One mail is from Bob (originated from the **1.2.3.4** IP address) and the other from Mallory (originated from **5.6.7.8**).

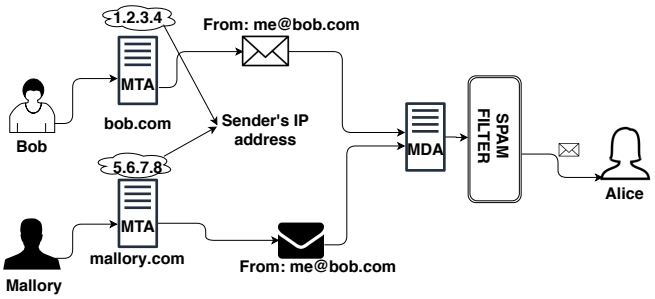

Fig. 1. Email sending and receiving procedure.

An effective anti-spoofing mechanism needs to differentiate the Mallory message from the legitimate Bob's mail. The current first lines of defence to protect end-users from spoofed emails include SPF [4], DKIM [5], and DMARC [6].

A. SPF – Sender Policy Framework

SPF is a set of text-form rules in **TXT** resource records of the Domain Name System (DNS). SPF specifies a list of servers allowed to send emails on behalf of a specific domain. During mail delivery over the SMTP protocol, the recipient server authenticates the sender Mail Transfer Agent (MTA) using a given **HELO** or **MAIL FROM** identity based on the published SPF record and the IP address of the sender—SPF needs to contain the domain portion of the **MAIL FROM** identity. In our example, the Alice server gets the **TXT** records of the **bob.com** domain from DNS. Then, it checks whether the sender IP address is on the list of IP addresses allowed to send emails from the **bob.com** domain and decides whether the message should be rejected or delivered to Alice.

The decision is made by the **check_host** function described in RFC 7208 [4] that takes three arguments on input (IP address of the sender, the domain, the **MAIL FROM** or **HELO** identity) and returns one of the seven possible results shown in Table I. The third column of the table presents the actions recommended by RFC 7208.

Below, we review the most common SPF rules useful for understanding the threat models presented in the next section (cf. RFC 7208 for more details). A valid SPF version 1

record must begin with string **v=spf1** followed by other SPF *mechanisms*, *qualifiers*, and *modifiers*. Mechanisms describe the set of mail servers for a domain and can be prefixed with one of four qualifiers: **+** (*Pass*), **-** (*Fail*), ∼ (*SoftFail*), **?** (*Neutral*). If a mechanism results in a match, its qualifier value is used. *Pass* (i.e., **+**) is the default qualifier.

The most common SFP mechanisms are the following:

- **ip4** and **ip6** – they specify an address or a set of IPv4 (or IPv6) addresses to match by the **check_host** function with respect to the sender IP address.
- **a** and **mx** – they tell the **check_host** function to perform first a DNS lookup for **A** (or **MX**) records of a given domain and then compare the returned IP addresses with the IP address of the sender.
- **exists** – it indicates a DNS domain name used for a DNS **A** query. If the query returns any **A** record, this mechanism matches.
- **include** – it tells the **check_host** function to include the SPF rule of another domain in the evaluation, which may result in calling the **check_host** function recursively to fetch and analyze the SPF records of the included domains.
- **all** – it always matches, so its corresponding qualifier results in the final decision. For example, **v=spf1 mx -all** means: allow **MX** servers of the domain to send mail and prohibit all others.

The final result of the mechanisms could be *Match*, *No match*, or *Exception*. Qualifiers combined with mechanisms, generate the final input for the **check_host** function that evaluates the SPF rule.

Modifiers provide additional information about the SPF records, for instance:

- **redirect: another-domain** – the SPF record for **another-domain** replaces the current record. The redirected domain becomes the target of all DNS queries and evaluations instead of the original domain.

Let us consider the following example:

v=spf1 a ip4:1.2.3.0/24 -all

where the **A** record **example.com A 6.7.8.9** is stored in DNS. The SPF rule states that only machines with the IP address of **6.7.8.9** (the **a** mechanism) or with the IP address in the range of **1.2.3.0**...**255** (the **ip4** mechanism) are permitted senders (all others are forbidden). However, by only changing **-all** to **+all**, any machine is permitted to send emails on behalf of the domain **example.com** with the successful SPF *Pass* result.

B. DMARC

DMARC [6] builds on top of SPF and DKIM by explicitly stating the policies to apply to the results of SPF and DKIM. In particular, DMARC binds names checked by SPF with what is listed in the '*From:*' field of the mail header by means of *alignment*, which expresses the fact that these domain names should match (or partially match when using a relaxed setup). For instance, DMARC checks whether the name in the '*Mail From:*' SMTP command and the '*From:*' field of the mail header match or not. In the case of the alignment test failure, a DMARC policy can specify what to do with the message (accept, reject, or quarantine) and where to send reports in case of a mismatch. The DMARC policy is stored in the **TXT** record of **_dmarc.domain.tld**. Below, we present selected tags of DMARC that, when misconfigured, can be exploited by an adversary.

- **aspf** (Alignment mode for SPF) – it specifies whether the strict (**s** value) or relaxed (**r** value) alignment mode is required by the domain owner. The default value is the relaxed mode. In the strict mode, the domain name used in SPF must be the same as the domain used in the '*From:*' field of the header. In the relaxed mode, any subdomain of the domain can be used in the '*From:*' field of the header and will result in *Pass*.
- **p** (Policy) – it specifies the action to be taken by the receiver if the alignment test results in *Fail*. Possible values for this tag are: 1) **none** – no specific action, 2) **quarantine** – the message is suspicious and depending on the mail system of the recipient, it could be delivered as spam, 3) **reject** – the domain owner wishes to reject emails during the SMTP transaction that fail the alignment test.
- **sp** (Subdomain policy) – it has the same syntax as **p** but applies to subdomains of the domain name. In the absence of this tag, the policy of the **p** tag must be applied to all subdomains [6]. If subdomains are not used to send emails, the owner can set this tag to the **reject** value to prevent subdomain email spoofing.

Let us assume that the DMARC rule of the domain **example.com** is **v=DMARC1; p=none; aspf=r;**. If we have the previously mentioned SPF rule for this domain, an illegal sender with the IP address of **9.10.11.12** can forge emails on behalf of **example.com** or any (existent or non-existent) subdomain of **example.com**, and the delivery decision is up to the receiver since no strict rule has been specified in DMARC. However, changing the DMARC rule to **v=DMARC1; p=quarantine; sp=reject; aspf=s;** tells the receiver to label all the emails that did not pass the SPF evaluation as spam and reject all the emails from the subdomains of **example.com** at the SMTP level.

C. Threat Models

We now consider threats regarding SPF and DMARC in detail. To mitigate mail spoofing, domain owners set up SPF and DMARC rules then used by inbound mail servers. Therefore, if the recipient MTA does not support the SPF or DMARC check, no matter how strict the rules are, they will not be effective. A misconfigured SPF or DMARC (either syntactically or semantically) rule is as dangerous as the absence of the rules since the output of the evaluation does not lead to a correct decision.

We consider three possible types of threats:

- **Related to domain names**. If a domain uses a misconfigured SPF rule, then it is possible to send forged

emails from any IP address with the SPF *Pass* result. For example, we have discovered that **microsoft.com.tr** used the **+all** mechanism in its SPF rule, which made it easy for attackers to send forged emails on behalf of Microsoft from any IP address[2].

- **Related to subdomains**. Each subdomain should have its own SPF and DMARC rules. Another possibility is to use the **sp** tag in DMARC of the domain name (lower-level domain) to explicitly specify the action to take when receiving messages from subdomains. A possible abuse of subdomains is the following:
  - If a subdomain has no SPF rule (and there is no specified wildcard rule) and no explicit DMARC action, then it is possible to misuse the subdomain for sending forged emails. For example, while **icann.org** has a strict SPF rule, there is no rule specified in **account.icann.org** and no DMARC policy regarding subdomains (also the default action for domains is **none**, which in this case applies to subdomains). Hence, it is possible to send emails with forged sender addresses (e.g., **support@account.icann.org**) with the SPF *Neutral* result.
  - If a subdomain does not exist, the result of the DNS query for the **TXT** record returns a name error (NXDOMAIN). Thus, the **check_host** function returns the *None* result (see Table I). If there is no wildcard **TXT** record that covers non-existing subdomains and there is no DMARC policy specified for subdomains and the domain itself, then again, it is possible to send spoofed emails.
- **Wrong SPF rules**. If the **check_host** function cannot evaluate the existing SPF record of a domain name because of a syntax error, then the result is either *Temperror* or *Permerror*, and a legitimate email will likely arrive in the spam box. However, when the user marks this email as safe, the mail service may also accept spoofed emails from other IP addresses. We show in Section V how syntactically wrong SPF rules may break the trust-based authentication system of email service providers by allowing forged emails to land in the user inbox.

III. Methodology

In this section, we describe the methodology for analyzing the deployment of SPF and DMARC at high-profile and defensively registered domains. Our focus is on well-known companies, governmental websites, financial institutions as well as defensive domain registrations. We start with two datasets: top 500 domains of 139 countries from the Alexa list [9] and online banking systems for all countries provided by FONDY[3].

A. Top 500 Websites of All Countries

While the Alexa list can easily be manipulated [10], it provides top 500 lists of most visited websites for 139 countries, which we collect for the purpose of this study. Previous work [8], [11] used the Alexa top 1 million domains. However, we are interested in specific domains that may not be in the top 1M list but in the top list of each country e.g., government websites or national businesses. In total, we collect 69,500 fully qualified domain names (FQDNs), which lead to 32,042 unique domains. Domain names are defined as 2nd–level, or lower-level if a given TLD operator provides such registrations, e.g., **example.br** or **example.com.br** [12]. We use a modified version of the public suffix list maintained by Mozilla[4] to get domains from FQDNs. For the purpose of this study, we exclude all private TLDs such as **s3.amazonaws.com** or **blogspot.com**. The dataset consist of 14,084 domains with legacy generic top-level domains (gTLDs), 1,070 domains with new gTLDs, and 14,084 domains with country-code TLDs (ccTLDs). We refer to this list as the TOP500 list.

B. Defensive Registrations

They refer to the process of registering domain names (often across multiple TLDs) with different grammatical formats to protect brands from attacks like *typo-squatting* [13]. For example, the **brand.com** company may register **brand.net** and **brand.org**, then redirect them to the original website. We use the following steps to generate defensively registered names using the names in the TOP500 list:

- For each domain name in the TOP500 list, we generate the domain names over all the possible TLDs including new gTLDs, legacy gTLDs, and ccTLDs. For example, for **paypal.com**, we generate **paypal.tld** where **tld** is all the ccTLDs (e.g, **paypal.in**), legacy gTLDs (e.g., **paypal.net**), and new gTLDs (e.g., **paypal.support**).
- For each domain in the TOP500 list that uses country code TLD or legacy gTLD, we generate *-squatting domains (for *-squatting, we use insertion, deletion, substitution, and internationalized domain names [14]).

We generate 145,250,849 unique domain names. Then, we scan all domains for **TXT** records with the ZDNS scanner from the ZMap project [15]. By excluding all DNS error results (e.g., NXDOMAIN, TIMEOUT, and SERVFAIL), we end up with 1,185,167 unique domains. Then, we extract the defensively registered domains based on the following three conditions:

1) IP address in the requested **A** record of the domain is the same as for the **A** record of at least one corresponding domain in the TOP500 list,
2) authoritative name server in the **NS** record of the domain is the same as in the **NS** record of at least one corresponding domain in the TOP500 list,
3) domain part of the automatically visited domain homepage URL is the same as one domain in the TOP500 list,

and the list reduces to 235,508 domains. Some of the domains in the list are related to web trackers [16] and parked

[2]After notifying Microsoft, the issue was fixed.

[3]https://fondy.eu

[4]https://publicsuffix.org

TABLE II
SCAN RESULTS FOR SPF RULES.

dataset	total	norecord (%)	noerror (%)	servfail (%)	nxdomain (%)	timeout (%)
TOP500 domains	32,017	29.88	65.92	0.23	0.18	3.78
TOP500 subdomains	212,361	76.15	5.77	0.1	16.31	1.68
Bank domains	7,022	22.39	64.95	1.28	2.75	8.63
Bank subdomains	39,310	70.34	3.53	0.09	22.96	3.09
Defensive domains	55,095	1.2	95.37	0.43	1.03	1.97

TABLE III
SCAN RESULTS FOR DMARC RULES.

dataset	total	noerror (%)	servfail (%)	nxdomain (%)	timeout (%)
TOP500 domains	32,017	34.32	0.24	63.44	2.0
TOP500 subdomains	21,2361	12.61	0.36	82.95	4.09
Bank domains	7,022	35.86	1.21	52.32	10.61
Bank subdomains	39,310	7.95	0.55	87.92	3.58
Defensive domains	55,095	40.08	0.36	57.86	1.7

domains. For parked domains, we exclude them using the method proposed by Vissers et al. [17], whereas for web trackers and advertising domains, we exclude them by using the Mozilla blacklist for trackers [18]. Finally, the list contains 55,059 defensively registered domains. For example, our list contains 226 domain names either registered by Google Inc. for **`google.com`** or by MarkMonitor[5] on behalf of Google.

C. Subdomain Enumeration

We have generated the list of known subdomains for each entry of the TOP500 list using the Spyse[6] API. We only consider first-level subdomains and exclude **`www`** and name servers. In total, we generate 212,361 subdomains for domains in the TOP500 list.

D. Banks and Financial Websites

For banking and financial websites, we leverage a list of 7,022 domains from the FONDY github repository[7] and generate 39,310 subdomains using the same method as described in the previous section.

IV. RESULTS

After collecting all the datasets, we perform three types of scans for all domains and subdomains: 1) find **`TXT`** records to extract SPF rules, 2) find **`TXT`** records by prepending **`_dmarc`** to the domains and subdomains (i.e., **`_dmarc.domain.tld`**) to retrieve DMARC rules, and 3) analyze SPF and DMARC rules by emulating the **`check_host`** function [19] using our server IP address as the IP address of the sender (without actually sending emails).

In this section, we present the results of the first two scans for SPF and DMARC rules at each domain and its subdomains.

A. High-Profile Domains and Defensive Registrations

Tables II and III present the results of the scans using ZDNS[8] to retrieve SPF and DMARC rules. Columns contain the following information: 'norecord' – domains exist but there is no SPF rule in the **`TXT`** record of the domains, 'noerror' – the record exists and can be retrieved successfully, 'servfail' – DNS lookup failure, 'nxdomain' – the domain name does not exist in the zone file, 'timeout' – DNS timeout error. For DMARC, the 'nxdomain' column is the same as 'norecord' column for SPF (if we get 'NXDOMAIN' answer to the DNS query for **`_dmarc.domain.tld`**, it means that **`_dmarc`** subdomain does not exist so there is no DMARC rule).

We can notice in Table II that **29.9%** of the domains in the TOP500 list and **22.4%** of the online banking domains do not have SPF rules at all. As the **`check_host`** function for the domains without SPF rules returns *None* (see Table I), it is up to the receiver of the email to decide on whether to deliver a message and/or mark it as suspicious or not. While this behavior can be acceptable for regular domains, it is insecure for transactional domains (e.g., banking domains) as well as for high-profile domains (e.g., domains in the TOP500 list).

For defensively registered domains, Table II shows that only **1.2%** of them have no SPF rules, which is significantly lower than the results for TOP500 and banking domains. However, evaluating SPF alone is not sufficient since the final decisions about the delivery of messages are made by DMARC policies.

As shown in Table III, as many as **63.4%** and **52.3%** of TOP500 and banking domains have no DMARC rule, which means that even with correctly configured SPF rules it is still possible to spoof emails. Furthermore, for the domains with a DMARC rule in place (34.3% and 35.9% for TOP500 and banking domains, respectively), we have observed that a large part of them have the tag **`p`** equal to **`none`** (**60%** and **53.8%**, respectively, not shown in the table), which make them prone to email spoofing as well.

[5]https://markmonitor.com
[6]https://spyse.com
[7]https://github.com/cloudipsp/all_banks_ips
[8]https://github.com/zmap/zdns

TABLE IV
SPECIFIED DMARC ACTION FOR SUBDOMAINS WITH NO SPF RULE IN THE **TXT** RESOURCE RECORD.

data	total	no-DMARC	**none**	**reject**	**quarantine**	invalid rule
TOP500-sub-no-SPF	161,720	108,535 (67.1%)	32,008 (19.7%)	13,286 (8.21%)	7,803 (4.82%)	88 (0.05%)
Bank-sub-no-SPF	27,650	19,070 (68.9%)	4,849 (17.5%)	2,682 (9.6%)	1,023 (3.69%)	26 (0.09%)

TABLE V
RESULT OF THE SPF **CHECK_HOST** EMULATION.

Result	TOP500	bank	defensive	bank subdomains	TOP500 subdomains
None	10,106	1,956	1,441	37,149	198,615
Neutral	1,497	236	6,220	56	683
Pass	50	10	114	2	37
Fail	7,083	2,268	22,255	860	4,511
Softfail	10,617	1,591	21,804	354	6,019
Temperror	135	155	523	778	1,485
Permerror	2,529	806	2,738	111	1,011
Total	32,017	7,022	55,095	39,310	212,361

For defensively registered domains (see Table III), **57.9%** of them do not have a DMARC rule meaning that it is possible to send spoofed emails. Among 40.1% of the domains with a DMARC rule, **26.7%** have the **p** tag equal to **none** and 65% have the **p** tag set to **reject**, which makes them bulletproof from domain spoofing at the SMTP transaction level.

Overall, we expect much larger deployment of SPF and stricter DMARC rules for defensively registered domains in comparison to high-profile domains—if organizations decide to register domains defensively to avoid domain name abuse, they are also likely to configure appropriate SPF and DMARC rules.

B. Analysis of Spoofing Possibilities for Subdomains

Regarding subdomains, the results are worse since **76.1%** of the subdomains related to the domains in the TOP500 list and **70%** of the subdomains related to banking websites do not have SPF records at all (see Table II). While it is not dangerous in itself, the absence of strict DMARC rules for subdomains makes them prone to subdomain spoofing. To mitigate this vulnerability, domains need to provide appropriate DMARC rules. The **sp** tag (or **p** tag in the absence of **sp**) in a DMARC rule specifies the default action to be taken upon receiving messages from subdomains with no SPF rule [6].

Table IV shows the DMARC results for subdomains without SPF rules in both TOP500 and banking website lists. To obtain this result, we first scan **_dmarc.sub.domain.tld** to extract a **p** tag from each subdomain and in case of no DMARC rule in the subdomain, we scan **_dmarc.domain.tld** for **sp** or (in the case of its absence) **p** tags and apply the rule to subdomains (cf. RFC 7489 for more details [6]). In Table IV, **none**, **reject**, and **quarantine** columns correspond to the extracted rules as explained in Section II-B. The 'invalid rule' column refers to the rules that do not follow the syntax specified in RFC 7489 and 'no-DMARC' column corresponds to the domains without DMARC rules in subdomains nor in the domain name. Note that sending emails from a subdomain of any domain with 'no-DMARC' (**67.1%** for TOP500 and **68.9%** for banking websites), with 'none' rule (**19.7%** for TOP500 and **17.5%** for banking websites), and 'invalid-rule' (less than 0.1% in both cases), regardless of the fact if the subdomain exists or not (non-existing subdomains), does not result in a strict reject decision. This behavior is potentially dangerous for transactional domains as it is possible to send emails with forged sender address using subdomains with no SPF record for as many as approximately **87%** of TOP500 and banking domains.

C. SPF Emulation Results

To analyze the validity of SPF rules using the **check_host** function further, we take advantage of **pyspf** [19] with our server IP address as the IP address of the mail sender. **pyspf** evaluates the SPF rule for a given domain and returns the SPF result. Table V shows the results of the SPF emulation (see Table I for the definition of each result and the corresponding recommended action). The reason for the SPF *Pass* result is either because of the **+all** mechanism in the SPF rule or the possible **redirect** modifier. Among the defensively registered domain names with the *Pass* result (114 domains), we have observed some well-known names like **microsoft.com.tr**[9] registered by MarkMonitor Inc.[5] on behalf of the Microsoft Corporation, as well as some major IT companies, local government, and TV channels websites for which we cannot provide the names for security considerations. However, the emulation results are available upon request.

We observe 12 different banking websites (1 in Spain and 11 in the United States) with the SPF *Pass* result. Although the number is fairly low, it is still enough for attackers to conduct a successful attack if they obtain the list of customer emails. In the TOP500 list for domains and subdomains, we observe 87 records with the SPF *Pass* result (50 for domains and 37 for subdomains) including several local government websites (mostly in the US), national financial websites, and national mobile operators with thousands of customers.

Table V shows 7,195 *Permerror* as the result of the **check_host** function. The majority of these domains and

[9]The issue was fixed after sending notifications.

subdomains have at least one of the following three problems: i) syntax problem in the published SPF rule (approximately 5,400 records), ii) exceeding the number of DNS lookups because of too many recursive `include` mechanisms [4] (1,131 records), and iii) published more than one valid SPF records (640 samples). The domains and subdomains with *Permerror* are important because they may cause serious problems. Since the domains have SPF records, it indicates that they are used by their owners to send legitimate messages to users. However, emails may never get delivered or delivered but labelled as spam (based on the action recommended for *Permerror* as described in Table I). Importantly, we find that any attempt by the end user to detach the spam label from the legitimate email may whitelist all the emails from that domain name with the SPF *Permerror* result including forged emails (see Section V).

Moreover, a wrong implementation of the `check_host` function on the receiver without strict limitation of the number of DNS queries, may allow the attacker to put extra burden on the local recursive DNS resolver, which may lead to a Denial of Service (DoS) attack against the DNS server, as explained by Scheffler et al. [20]. Among the domains with a syntactically wrong SPF rules, we observe some major IT companies e.g., `eset.lu`, the defensively registered domain for `eset.com` related to the ESET Internet Security[9].

The SPF emulation results show that for several major IT companies, government websites, one of the topmost banking website in the world, it is possible to send spoofed emails from both existent and non-existent subdomains as well as from some of their defensively registered domains due to weak or misconfigured SPF or DMARC rules.

V. Trust-based Authentication Issue

In this section, we show how a syntactically wrong SPF rule in a legitimate domain can push users to break the trust-based authentication system by labeling a legitimate email as safe and letting forged emails land in the user inbox. We examine five popular email providers: Outlook, Yahoo, Gmail, Laposte, and Yandex. We explain the issue using the Outlook service as an example, but the process is the same for other email service providers.

First, we register a domain (`dnsabuse.xyz`), set up a mail server, and the DNS `A` record of the domain. We use `v=spf1 a aaaa -all` as the SPF rule in the `TXT` record for our registered domain (i.e., syntactically wrong SPF rule because of nonexistent `aaaa` mechanism, to generate the *Permerror* result). Then, we send a legitimate email with our server to our `outlook.com` email address. Since the SPF record is syntactically wrong and the reputation of our domain is low, the legitimate email lands in the spam box (as we expect) with the SPF *Permerror* result. If the user marks the email as *'safe sender'* (in case of Yahoo mail, the button label is *'add sender to contacts'*), then the Outlook service considers this email as safe (correct assumption as it is a legitimate email). However, from now on, Outlook (as well as Yahoo) also accepts spoofed emails from other IP addresses that spoof the domain name.

We suspect that Yahoo and Outlook services, whitelist the sender domain name instead of their IP addresses. On the other hand, the Laposte service rejects the sender with SPF *Permerror* at the SMTP level and sends a bounce message informing the sender about the reason of rejecting mails (i.e., syntax error of SPF). For the Yandex mail service, we were not able to evaluate the trust-based authentication since both emails (from the legitimate and illegitimate servers) land in the user inbox. Finally, the Gmail service does not suffer from the issue. We assume that when users detach the spam label from a legitimate email, the Gmail service only whitelists the IP address instead of the domain name.

VI. Remediation

Notifying owners of the affected domains with misconfigured (or lack of) SPF and DMARC rules is highly problematic since there is no straight way to retrieve the contact information of the domain owners [21]. Public availability of domain WHOIS data is affected by the introduction of the General Data Protection Regulation (GDPR) and "Temporary Specification for gTLD Registration Data" [22]. It allows generic TLD registries to redact the Registrant and Administrative Contact from the public WHOIS. Therefore, we decided to perform notifications through the Computer Security Incident Response Teams (CSIRTs). We use the following bottom-up approach to send notifications—we send email notifications if there is a CSIRT responsible for: 1) the domain name, 2) the TLD of the domain, 3) the IP range to which the IP address of the domain belongs to, 4) the autonomous system of the IP address for that domain, or 5) the ccTLD (not the registry operator itself). In total, we have sent 128 emails to notify CSIRTs responsible for 7,653 domains about the problem. We were not able to find any abuse contact address for 573 domains. For some high-profile domains prone to phishing attacks, e.g., `microsoft.com.tr`, we manually visited their websites and contacted them directly. In the first 5 days after sending notifications, we repeated our scans and found that 160 domain owners remediated the problem by re-configuring their SPF rules. The quickest clean-up action was initiated by US government CERT (50 domains), national CERT of Austria (7 domains), Spain (7 domains) followed by CERT Polska, French CERT (ANSSI) and Danish CERT (CFCS-DK): 5 domains each. We plan to perform a large scale scanning of domains and notifying CSIRTs about misconfigured SPF and DMARC.

VII. Related Work

In this section, we briefly review previous work concerning measuring and analyzing email authentication systems.

Durumeric et al. [23] measured the global adoption of SMTP security extensions and the resulting impact on end users. They studied SMTP server configurations for the Alexa top million domains and over a year of SMTP connections to and from Gmail. They reported the existence of a long tail of over 700,000 SMTP servers, of which only 35%

successfully configure encryption, and only 1.1% specify a DMARC authentication policy.

In 2017, Durumeric [7] measured the extent of SPF and DMARC adoption for one million top domains in the Alexa list. His results showed that 40.1% of the domains have published SPF records while only 1.1% of them have valid DMARC records. Hu and Wang [8] reported similar statistics in 2018 with the results of 44.9% published SPF records and 5.1% published DMARC records showing approximately 5% of increase in one year. In their end-to-end experiment, they spoofed 30 high-profile domains and reported the ratio of emails that reached inboxes of selected email providers. We perform a similar analysis for both SPF and DMARC records but with the focus on more prominent domains (with transactional emails) including banking websites, government portals, national and international businesses as well as defensively registered domains, existent and non-existent domains. However, we did not perform domain and subdomain spoofing on high-profile domains.

Foster et al. [11] evaluated the security extensions using a combination of measurement techniques to determine whether major providers support the Transport Layer Security (TLS) protocol [24] at each point in their email message path, and whether they support SPF and DKIM on incoming and outgoing mail. They reported that while the use of SPF is common, enforcement was limited. Scheffler et al. [20] investigated the consequence of a wrong implementation of the **`check_host`** function at the receiver, which lets attackers perform DoS attacks on a closed local DNS resolver. While our goal is not to evaluate the SPF abuse, we show that 1,131 domains have published SPF records that require more than 10 DNS lookups and thus, may abuse local DNS resolvers.

Hu et al. [25] investigated the reasons behind the low adoption rates of anti-spoofing protocols. They conducted a user study involving email administrators and showed that they believe the current protocol adoption lacks the crucial mass due to the protocol defects, weak incentives, and practical deployment challenges.

VIII. Conclusion

In this paper, we evaluate the adoption of the SPF and DMARC security extensions by high-profile domains and analyze spoofing possibilities enabled by the absence or misconfigurations of their rules. The results show that a large part of the domains do not correctly configure the SPF and DMARC rules, which enables attackers to successfully deliver forged emails to user inboxes. In particular, we show that for top 500 domains of 139 countries, the adoption rate of SPF and DMARC records are 65.9% and 34.3%, respectively. For banking websites, we obtain almost the same results (64.9% and 35.9%) as for the TOP500 list. However, for defensively registered domains, the results are significantly higher especially in terms of published SPF records with 95.37% adoption and 40.1% for DMARC. We also, for the first time, investigate the problem of subdomains in the anti-spoofing techniques and their possible abuse to send forged emails.

We also emulate the SPF **`check_host`** function not only to evaluate *Pass* and *Fail* results but also all the possible results such as *Permerror*, *None*, and *Neutral* for both domains and subdomains. The investigation shows that syntactically wrong SPF rules may break the trust-based authentication system of email service providers (e.g., Outlook and Yahoo) by allowing forged emails to land in the user inbox.

To help in remediation, we have sent 128 emails to notify CSIRTs responsible for 7,653 domains. Within the first five days after the notification campaign, they managed to inform domain owners and to mitigate SPF configuration errors of 160 vulnerable domains.

Finally, while we do not publish the scan data because of ethical concerns, we make the data available upon request to encourage reproducibility.

Acknowledgments

We thank Arnold Hölzel (SMT B.V) for his valuable feedback. This work has been carried out in the framework of the COMAR project funded by SIDN, the .NL Registry and AFNIC, the .FR Registry. It was partially supported by the PrevDDoS project funded by IDEX UGA IRS and the ANR projects: the Grenoble Alpes Cybersecurity Institute CYBER@ALPS under contract ANR-15-IDEX-02, PERSYVAL-Lab under contract ANR-11-LABX-0025-01, and DiNS under contract ANR-19-CE25-0009-01.

References

[1] (2019) Internet Crime Report. [Online]. Available: https://www.fbi.gov/news/stories/2019-internet-crime-report-released-021120

[2] (2019) A Phishing Campaign Reports Unusual Activity on Your Microsoft Account. [Online]. Available: https://www.logitheque.com

[3] J. Klensin, "RFC 5321: Simple Mail Transfer Protocol," Internet Requests for Comments, 2015. [Online]. Available: http://tools.ietf.org/html/rfc5321

[4] S. Kitterman, "RFC 7208: Sender Policy Framework (SPF) for Authorizing Use of Domains in Email," Internet Requests for Comments, 2014. [Online]. Available: http://tools.ietf.org/html/rfc7208

[5] D. Crocker, T. Hansen, and M. Kucherawy, "RFC 6376: DomainKeys Identified Mail (DKIM) Signatures," Internet Requests for Comments, 2011. [Online]. Available: http://tools.ietf.org/html/rfc6376

[6] E. Kucherawy, M. Zwicky, and E. Zwicky, "RFC 7489: Domain-Based Message Authentication, Reporting, and Conformance (DMARC)," Internet Requests for Comments, 2015. [Online]. Available: http://tools.ietf.org/html/rfc7489

[7] Z. Durumeric, "Fast Internet-Wide Scanning: A New Security Perspective," Ph.D. dissertation, University of Michigan, 2017.

[8] H. Hu and G. Wang, "End-to-End Measurements of Email Spoofing Attacks," in *Proc. 27th USENIX Security Symposium*, 2018, pp. 1095–1112.

[9] (2019) The Top 500 Sites on the Web. [Online]. Available: https://www.alexa.com/topsites/countries

[10] V. L. Pochat, T. Van Goethem, S. Tajalizadehkhoob, M. Korczyński, and W. Joosen, "Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation," in *Proc. NDSS Symposium*, 2019.

[11] I. D. Foster *et al.*, "Security by Any Other Name: On the Effectiveness of Provider Based Email Security," in *Proc. 22nd ACM CCS Conference*. ACM, 2015, pp. 450–464.

[12] M. Korczyński, S. Tajalizadehkhoob, A. Noroozian, M. Wullink, C. Hesselman, and M. van Eeten, "Reputation Metrics Design to Improve Intermediary Incentives for Security of TLDs," in *Proc. Euro S&P*, 2017, pp. 579–594.

[13] J. Szurdi, B. Kocso, G. Cseh, J. Spring, M. Felegyhazi, and C. Kanich, "The Long "Taile" of Typosquatting Domain Names," in *Proc. 23rd USENIX Security Symposium*, 2014, pp. 191–206.

[14] V. Le Pochat, T. Van Goethem, and W. Joosen, "Funny Accents: Exploring Genuine Interest in Internationalized Domain Names," in *Proc. PAM Conference*. Springer, 2019.
[15] Z. Durumeric, E. Wustrow, and J. A. Halderman, "ZMap: Fast Internet-Wide Scanning and its Security Applications," in *Proc. 23rd USENIX Security Symposium*, 2013, pp. 605–620.
[16] S. Schelter and J. Kunegis, "Tracking the Trackers: A Large-Scale Analysis of Embedded Web Trackers," in *Proc. 10th International AAAI Conference on Web and Social Media*, 2016.
[17] T. Vissers, W. Joosen, and N. Nikiforakis, "Parking Sensors: Analyzing and Detecting Parked Domains," in *Proc. NDSS Symposium*. Internet Society, 2015, pp. 53–53.
[18] (2019) Shavar Tracking Protection Lists. [Online]. Available: https://github.com/mozilla-services/shavar-prod-lists
[19] (2019) Python SPF Package. [Online]. Available: https://pypi.org/project/pyspf/
[20] S. Scheffler, S. Smith, Y. Gilad, and S. Goldberg, "The Unintended Consequences of Email Spam Prevention," in *Proc. PAM Conference*. Springer, 2018, pp. 158–169.
[21] O. Cetin, C. Ganan, M. Korczyński, and M. van Eeten, "Make Notifications Great Again: Learning How to Notify in the Age of Large-Scale Vulnerability Scanning," in *Workshop on the Economy of Information Security*, 2017.
[22] ICANN. (2018, May) Temporary Specification for gTLD Registration Data. ICANN. [Online]. Available: https://www.icann.org/resources/pages/gtld-registration-data-specs-en
[23] Z. Durumeric *et al.*, "Neither Snow Nor Rain Nor MITM...: An Empirical Analysis of Email Delivery Security," in *Proc. ACM IMC Conference*. ACM, 2015, pp. 27–39.
[24] E. Rescorla, "RFC 8446: The Transport Layer Security (TLS) Protocol Version 1.3," Internet Requests for Comments, 2018. [Online]. Available: https://tools.ietf.org/html/rfc8446
[25] H. Hu, P. Peng, and G. Wang, "Towards Understanding the Adoption of Anti-Spoofing Protocols in Email Systems," in *Proc. SecDev Conference*. IEEE Computer Society, 2018, pp. 94–101.