TXTing 101: Finding Security Issues in the Long Tail of DNS TXT Records

Olivier van der Toorn
University of Twente
o.i.vandertoorn@utwente.nl

Roland van Rijswijk-Deij
University of Twente
r.m.vanrijswijk@utwente.nl

Tobias Fiebig
TU Delft
t.fiebig@tudelft.nl

Martina Lindorfer
TU Wien
martina@iseclab.org

Anna Sperotto
University of Twente
a.sperotto@utwente.nl

***Abstract*—The DNS TXT resource record is the one with the most flexibility for its contents, as it is a largely unstructured. Although it might be the ideal basis for storing any form of text-based information, it also poses a security threat, as TXT records can also be used for malicious and unintended practices. Yet, TXT records are often overlooked in security research. In this paper, we present the first structured study of the uses of TXT records, with a specific focus on security implications. We are able to classify over 99.54% of all TXT records in our dataset, finding security issues including accidentally published private keys and exploit delivery attempts. We also report on our lessons learned during our large-scale, systematic analysis of TXT records.**

***Index Terms*—DNS, Security, Measurement, Classification**

1. Introduction

The Domain Name System (DNS) is critical to the Internet's infrastructure but has long outgrown its original purpose of resolving names to IP addresses. Applications nowadays rely on DNS to prevent email spoofing (SPF) and to verify SSH (Secure SHell) and TLS (Transport Layer Security) key fingerprints (SSHFP and TLSA). With DNS-over-HTTPS (DoH), it has now even become possible for browsers and browser-based malware to retrieve DNS information directly [1].

Of the existing resource record (RR) types, the TXT RR is the one that provides the most flexibility in terms of content. Its use and format have been subject to changes and debates over the years. While DNS TXT records initially were supposed to only hold descriptive text (RFC 1035), RFC 1464 tried to structure the record type by introducing a key–value store format, which, e.g., SPF and DKIM use. In practice, the freedom of an unstructured resource record remains, which allows anyone to publish any text-based information. This makes TXT records ideal candidates for malicious and unintended practices, yet, the TXT records are generally overlooked in security research.

We would like to thank Daniël Meinsma for starting this research by finding (malicious) PowerShell code samples in TXT records. This work has been partially funded by the EU H2020 projects CONCORDIA (#830927), CyberSecurity4Europe (#830929), and Safe-DEED (#825225). Furthermore, it has been funded by SIDN-fonds, an independent fund on the initiative of SIDN, the registrar for '.nl' domains. The research leading to these results has also received funding from SBA Research (SBA-K1), which is funded within the framework of COMET Competence Centers for Excellent Technologies by BMVIT, BMDW, and the federal state of Vienna, managed by the FFG.

TABLE 1: Overview of TXT records in our dataset.

	Class	Percent	# of Records
A	Standardized	68.95%	50,304,343
B	Non standardized	31.05%	22,655,424
C	- Legitimate, well defined	14.40%	10,504,491
D	- Legitimate, not well defined	15.48%	11,292,795
E	- Unclassified	1.17%	858,138

This paper aims at filling this void, by providing a longitudinal view of how DNS TXT records are used in practice, focusing on unconventional use cases and their security implications. Our dataset consists of all TXT records from OpenINTEL [2], [3], amounting to roughly 75 billion TXT records collected between March 2015 and December 2018. The breakdown of the TXT records in Table 1 shows that the majority has a well-defined purpose, being either standardized or non-standardized (83.35%). This includes email verification and the verification of domain ownership. For 15.48% of TXT records, the underlying use case is not well-defined, but matching them against regular expressions suggests legitimate use cases (e.g., references to DNS services, dates). Finally, 1.17% of the TXT records fall outside the previously mentioned categories. This leads to the question of what type of information is contained in this tail of TXT records.

Our main contributions are: (1) A structured and historical analysis of TXT records spanning more than three years. We highlight changes in how, and how often, TXT records are used. (2) An in-depth analysis of the so-far neglected tail of TXT records, focusing on security implications. (3) Our lessons learned, especially in terms of the amount of manual labor involved, in systematically analyzing security issues in TXT records.

2. Background and Related Work

DNS and TXT Record Use. DNS originally only tied domain names and IP addresses together, but has been continuously extended to keep pace with the technical requirements of the ever changing Internet. Major changes include the introduction of new record types, e.g., SRV, DNSSEC, DNS-over-TCP, and the introduction of DNS-over-HTTPS.

A more subtle way to add new functionality to DNS is overloading existing resource record types. TXT records have been commonly used for this, as they were initially built to hold descriptive (free) text. While some attempts were made to structure (RFC 1464) or discourage (RFC 5507) using TXT records this way, several common applications leverage them. For example, TXT records are used for various forms of email validation and spam prevention, including SPF, DKIM, and DMARC, but DNS TXT records can also be used as a way of finding contacts [4], or to monitor IoT devices [5]. Besides these legitimate use cases, malicious uses include adding large records to create more efficient DNS amplification attacks [6], or creating a command and control channel for malware [7], [8], [9], [10], [11], [12]. Most recently, spam campaigns have started to query DNS TXT records from JavaScript embedded in their HTML payload to dynamically redirect to target URLs [1].

DNS Measurement Studies. Initially, DNS-related measurement studies focused on passive measurements that investigate clients' use of DNS [13], [14], while active measurements provide a better understanding of the operational side of Internet infrastructure. To make such measurements reliable, research have to account for: (1) DNS not necessarily being consistent across several vantage points, (2) the large amount of involved data, and (3) measurements being temporally consistent. To address these issues, van Rijswijk et al. build OpenINTEL, a platform for longitudinal DNS scans [2], [3].

Streibelt et al. [15] actually use inconsistencies in DNS replies to measure DNS. In their specific case they use reply differences based on EDNS0 to measure DNS load balancing (RFC 1794). Studies into the use of *specific* RRs include Fiebig et al. looking into PTR records [16], [17], and Portier et al. taking a first look at TXT records [18]. The latter however focused mostly on quantifying the *well known* parts of TXT record use — mostly email and verification related tokens — and did not explore the *unstructured* tail of TXT records, and what security implications are tied to these records. Portier et al. do mention TXT records may leak information but do not explore what kind of information or in what quantity information is leaked. Even though, our analysis is based on a larger volume of TXT records — 75 billion TXT records collected over a three-year period instead of 1.4 billion records collected over a two-year period — we see comparable results in the high-level classification of records. Portier's 'Protocol enhancement' accounts for 76.60% compared to our 'Standardized' category of 68.95% of TXT records. They reported 15.61% as 'Domain verification', whereas we observed 14.40% in this category. Finally, Portier et al. classified 7.78% of records as unknown, which only amounts to 1.17% in our classification. An in-depth comparison of results is out of scope for this paper.

3. Methodology

Dataset Description. We use data from OpenINTEL for our research. OpenINTEL is an active DNS measurement platform developed at the University of Twente in collaboration with SURFnet, SIDN Labs, and NLnet Labs. Since March 2015, OpenINTEL collects daily DNS snapshots

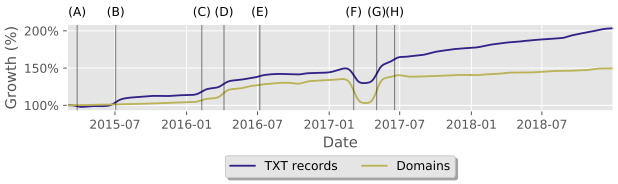


Figure 1: Growth of number of domains and TXT records.

TABLE 2: Dataset statistics.

	Domains/day	TXT records/day	TXT records/domain
min	194M	34M	0
max	325M	73M	848
mean	247M	53M	0.22
std	42M	11M	0.66

and currently queries around 65% of the global DNS name space, covering the zones .com, .net, .info, .mobi, the new ICANN gTLDs, and a set of ccTLDs such as .nl, .se, .ca, .fi, .at, .dk and .nu. We extracted all TXT records gathered between March 2015 and December 2018. As summarized in Table 2, on average we observed 53 million TXT records and 247 million domain names per day. Note that even though every domain only has 0.2 TXT records on average, we see a high variance, with individual domains containing hundreds of records at the apex.

Figure 1 shows the number of TXT records collected over time. The number of TXT records grows by a factor of 2 (from 35 million records in 2015 to 73 million in 2018). By comparison, the overall DNS name space expands only by a factor of 1.5 (from 128 million domains in 2015 to 191 million in 2018). A possible explanation for the increasing use of TXT records are email defenses like SPF and DKIM: the majority of TXT records are related to email, and stricter email policies from large providers (e.g., Google, Microsoft and Yahoo) [19], [20] may explain this rise.

Note that these statistics are subjective to the set of zones measured by OpenINTEL, which has grown over the years [21]. Furthermore, the drop in domains between 2017-03-01 and 2017-05-01 was caused by a major webhoster blocking (and later unblocking) OpenINTEL. Table 3 summarizes events impacting our measurements.

Categorization of TXT Records. To get a structured view on the TXT record ecosystem, we partition the set of TXT records based on regular expressions, and group similar classes into broader categories. We were careful to match a TXT record to a single regular expression, in order to prevent the record from being counted twice. Our process was iterative, we performed a heavy-hitter analysis to identify the current majority classes and built our regular expressions iteratively until the *Other* category did not contain unidentified items anymore. TXT records classified as 'SPF', 'SenderID' or 'DKIM' are part of the *E-mail* category. In addition, we identify the categories *Verification* (e.g., Google site verification, Facebook domain verification), *Patterns* (IP addresses, dates), *Encoded* (usually Base64), *Crypto Coins* and *Miscellaneous* (including service-specific tokens and hosting advertisement slogans). We group the remaining TXT records in the *Other* category, which includes unclassified records.

TABLE 3: Major events impacting our measurements.

*	Date	Event
(A)	2015-03-26	Major cloud provider cleans up SPF records
(B)	2015-07-03	DDoS protection service account hashes peak
(C)	2016-02-09	Start measurement .NL
(D)	2016-04-06	Start measurement .info, .mobi + new gTLDs
(E)	2016-07-07	Start measurement .ca
(F)	2017-03-01	Major webhoster blocks OpenINTEL
(G)	2017-05-05	Major webhoster unblocks OpenINTEL
(H)	2017-06-18	Start measurement .ru + .рф

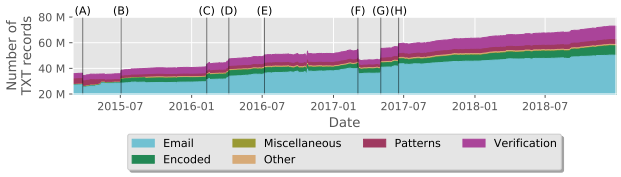


Figure 2: Evolution of TXT records over the span of three years (* annotated with events listed in Table 3).

Table 4 shows the distribution of records over our categories for a single day (2018-12-31), for which 0.46% of the records are left unclassified. Figure 2 shows how our categories evolve over time. We marked events (labeled A–H) that influenced the evolution of categories. Most events relate to the expansion of the OpenINTEL measurements (events C, D, E and H) or the major webhoster temporarily blocking the measurements (events F and G). Other events are large CDNs removing 1.80M SPF records of the form "`v=spf1 -all`" at once (A) and 2.64M account hashes suddenly appearing in TXT records (B).

Reproducibility. We used 101 regular expressions for our classification, which are available on our website: `https://www.tide-project.nl/blog/wtmc2020`.

4. Analysis of the Long Tail

As the structured part of TXT records has already been explored in the past [18], we focus on the *Other* category, i.e., the unstructured tail of TXT records. Figure 3 shows the evolution over time of the number of TXT records in the *Other* category, broken down by the classes identified in Table 4. Although the *Other* category represents on average only 1.28% of TXT records, it has grown significantly over the measurement period from 174K to 858K records (4.9x growth). We identified several events that contribute the most to this trend: In September 2015 (A) and later in August 2018 (D), we witness a rise of TXT records of a single character. We discuss these events in Section 4.1.3. In November 2015 (B) and later in July 2016 (C), there is a sudden rise in the Base64 Encoded MX records, which we discuss in Section 4.1.2.

4.1. Undefined Purpose

4.1.1. BaseN Encoded Records. We observe that 8.17% of all records are encoded with some form of BaseN, e.g., Base64. Portier et al. [18] suggested that a one source of these records is a federation mechanism of Microsoft Exchange Servers [22], which we indeed find for 0.20% of records in our dataset. Furthermore, we found a major

TABLE 4: DNS TXT record categories on 2018-12-31.

Label	# of Records	% of Total	Plot
All Records	72,959,767	100.00%	
E-mail	50,304,343	68.95%	
SPF	49,656,480	68.06%	
DKIM	310,823	0.43%	
SenderID	200,991	0.28%	
DMARC	118,928	0.16%	
Mail Keywords	17,121	0.02%	
Verification	10,504,491	14.40%	
Verif. Keywords	10,504,491	14.40%	
Patterns	3,770,930	5.17%	
Pat. Keywords	3,766,532	5.16%	
Pat. Kwd. Begin	4,396	0.01%	
JWT	2	>0.01%	
Encoded	7,215,892	9.89%	
BaseN	5,957,428	8.17%	
Hash	1,256,263	1.72%	
Account-hash	2,201	>0.01%	
Crypto Coins	89	>0.01%	
OAL	89	>0.01%	
Miscellaneous	305,884	0.42%	
Misc Keywords	200,402	0.27%	
HTTP	47,089	0.06%	
Hosting	19,755	0.03%	
Advertising	15,074	0.02%	
Google	13,899	0.02%	
Domainstatus	9,665	0.01%	
Other	858,138	1.18%	
Unclassified	335,920	0.46%	
Single Char	278,540	0.38%	
Base64 mail	228,672	0.31%	
Empty	14,425	0.02%	
No mail	221	>0.01%	
Javascript	178	>0.01%	
BEGIN	91	>0.01%	
Privkey	63	>0.01%	
Executables	22	>0.01%	
Cmd	6	>0.01%	

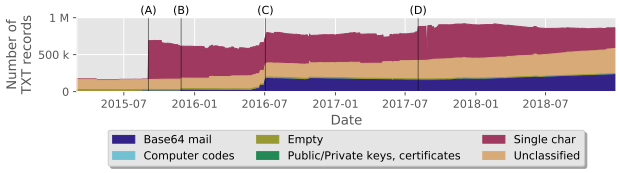


Figure 3: Evolution of selected categories of TXT records over the span of three years.

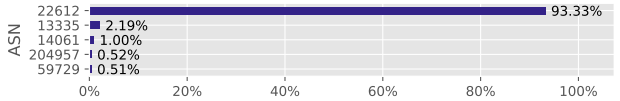


Figure 4: AS numbers with Base64 encoded MX records.

CDN adding Base64 encoded records to zones pointing to them (>0.01% of records in our dataset). After contacting the CDN, they confirmed that they added these records to the zones, but did not disclose the purpose of these records. We did not perform an in-depth investigation of the remaining 7.96% of other BaseN encoded records.

4.1.2. Base64 Encoded MX Records. Of all TXT records 0.31% (228.672) fall in this category. This type of TXT record has seen two sharp increases in use: first on 2015-11-27 (B) when 14,039 records of this type were added, and then between 2016-06-17 and 2016-07-03 (C) when 122,573 records were added.

When decoding these records we observe MX-record-like patterns (priority, host) in 228,631 domains with such a record, 99.86% (228,321) of which have an MX record, yet none of these domains' MX records matches the decoded TXT record. Figure 4 shows the top five AS numbers from which these records originate, with 93.33% (213,088) coming from 'NAMECHEAP' (AS 22612).

These records may be used in an email system where the MX address of a domain is obfuscated, i.e., through a public MX record (the regular MX record), and a 'private' MX record Base64 encoded into the TXT record.

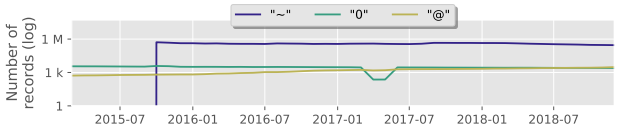


Figure 5: Single character TXT records over time.

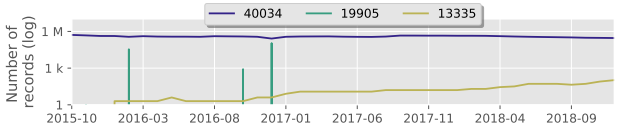
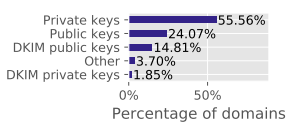


Figure 6: AS numbers hosting '~' TXT records.

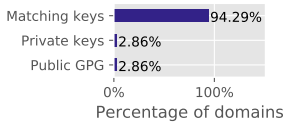
4.1.3. Single Character TXT Records. From the start of the measurement until 2015-09-03 (A) the number of records containing a single character has been stable around 61,000 records. On 2015-09-03 this number increased to 239,302 records, and increases a day later to 526,561 records. We note that two metrics change significantly due to massive addition of these records: Firstly, the composition of characters changes. On 2015-03-01 the most used character is '0'. On 2015-09-03 this changes to '~'. The records added on 2015-09-04 did not change the distribution, as mainly '~' records were added. Additionally, the distribution of characters does not change significantly afterwards, as shown in Figure 5, which presents a snapshot of the distribution of characters on the first of every month. Between 2017-08-02 and 2017-08-04 (D) 66,826 records consisting of a single character have been added. These records mainly consisted of '~' (99.98%).

Due to major rise in the '~' character we investigated where these TXT records originated from. Figure 6 shows that the majority (99.99%) of these '~' records originates from the network of 'CONFLUENCE-NETWORK-INC' (AS 40034), notorious for spreading malware and having many of their IP addresses listed in numerous blacklists [23]. Note, that on 2016-02-01, 2016-10-01, and 2016-12-01 these records also come from 'NEUSTAR-AS6' (AS 19905), a DDoS protection service. Disregarding the '~' records, it is likely that single characters records are the effect of typos when creating TXT records. However, the '~' records seems like a form of domain identification for a major network, as a large part (94.05%) shares the same AS number and the domains point to the same name servers. Both, in the case of 'Base64 encoded MX record' and 'Single character TXT record', we speculate that this particular use of TXT records is heavily guided by management and configuration choices, in which specific service providers or ASes use TXT records as a way to tag their domains.

4.1.4. Other. Finally, we found empty TXT records and records referencing executables. On 2018-12-31 we measured 14,425 empty TXT records, associated with 14,422 domains. These TXT records originate from various AS numbers, with the top contender being 'UNIFIEDLAYER-AS-1' (AS 46606) hosting 12.01% (1.885) of related domains. These observations suggest this phenomenon is the result of management mistakes, potentially caused by improper removal of TXT records, or a default record set for new domains. Empty records are unlikely to have security implications.



(a) Domains with a single key



(b) Domains with two keys

Figure 7: Statistics of domains with keys in their records.

We found three TXT records referencing executables (on 2018-12-31). Two of these point to an URL of a downloadable (Windows) executable. However, when trying to access these URLs we get either a not found or a permission denied error. The third record consists of 'calc.exe'. This is interesting, as researchers commonly use the execution of 'calc.exe' in Proof-of-Concept exploits against Windows systems. However, we were unable to identify the specific use of the identified record.

4.2. Mistakes with a Security Implication

4.2.1. Certificates. The last day of our dataset contains 43 certificates and 17 certificate requests. We processed each of these with *openssl* to verify if the records contain valid certificates or certificate requests and found 16.28% (7) certificates to be valid. The others were truncated, and *openssl* marked them as invalid. Of the certificate requests, 29.41% (5) are valid. Worryingly, one of these certificate requests included the private key.

85.71% (6) of the valid certificates contained references of the domain where the TXT record originated from. As for the certificate requests, those were all issued for the domain where the TXT record originates from. This tells us that most of these certificates and requests are linked to the functioning of the domains.

While domain ownership verification through DNS TXT records to obtain a certificate is common (RFC 8555), performing certificate requests via TXT records is not common practice. Concerning the certificate themselves, RFC 4398 proposed a specific CERT RR to store certificates, but to our knowledge the effort has been abandoned, and the small number of certificates we observe does not indicate that TXT is commonly used for this purpose. While publishing certificates via TXT records is not a security risk in itself, the fact that we found private keys accompanying requests still suggests security-relevant configuration mistakes [24] are being made.

4.2.2. Public and Private Keys. We observed a rise of TXT records containing public and private keys at the end of 2018 (not including the aforementioned certificate containing a private key). This is due to 28 domains which have added key-pairs on 2018-10-19. At the end of the measurement period, the number of domains exposing keys has grown to 89. On the last day of our dataset (2018-12-31), 60.67% (54) of the domains have a single key in their records, and 39.33% (35) of domains have two keys in their records. We have analyzed these keys and classified them into different types, as shown in Figure 7.

Figure 7a shows the key type distribution of domains with a single key. 55.56% (30) of these domains publish a private key in their TXT records, while 24.07% (13) publish a public key. In 16.66% (9) of

the cases, the TXT records is used for DKIM which includes the "----BEGIN RSA PRIVATE KEY----" part. This means that not only the wrong key is being published, but it also renders the DKIM record unusable. One domain publishes a GPG public key through its TXT record. Furthermore, there is one domain with a certificate issue request, with accompanying private key, as mentioned in Section 4.2.1.

Figure 7b shows the key type distribution of domains with two keys in their TXT records. In 94.29% (33) of the cases we found a matching public and private key pair. One domain published the same private key in two separate records, one with, and the other without, a leading dash. One domain published two different public GPG keys in its TXT record.

The fact that we observe a number of private keys is worrying not just in itself, but also because the disclosure of public-private key pairs in practice invalidates security measures as forgery prevention using DKIM. For example, if an adversary has access to a domain's private key used for DKIM signing, they can send emails in that domain's name with the receiving party assuming the origin of the email is legitimate as it is signed with the correct private key. Furthermore, the wrongly posted public keys at least show a misunderstanding of the underlying security technology. We have notified the domain-holders who publish private keys so they can mitigate this issue.

4.3. Malicious Use Cases

4.3.1. Commands. We investigated if the TXT records in our dataset contain commands, specifically commands with malicious intent. In our dataset there are six records containing Command Line Interface (CLI) commands. One record targets Windows with a command to kill Internet Explorer, while the other commands target Linux. Two of the Linux commands aim to test for the Shellshock vulnerability [25], for example:

```
() { :;}; echo "shellshock.fail"
```

Further two records contain `curl` commands. The remaining command forces `apt` (the Linux package manager) to retrieve packages over IPv4, after which it makes sure `curl` is installed, proceeds to download a script from runclound.io (with `curl`) and finally runs it in bash.

These kind of records may be used as (reliable) shell script distribution, since DNS traffic is rarely blocked, while HTTP traffic to specific websites may be dropped.

4.3.2. JavaScript. On the last day of our dataset 172 records contain JavaScript. The ten most common categories in these records are shown in Figure 8: 35.71% (40) of these records are used to load additional JavaScript files; 16.07% (18) contain JavaScript code for analytics purposes, and 16.07% (18) of these records reference Google Ads. Furthermore, these records are used in 15.18% (17) of cases to test if sites are vulnerable to cross-site scripting, typically through calling '`alert()`'.

These types or records may be used to stealthily insert JavaScript code into browsers [26]. As these are dynamically inserted it is unlikely that they leave a long-lasting trace.

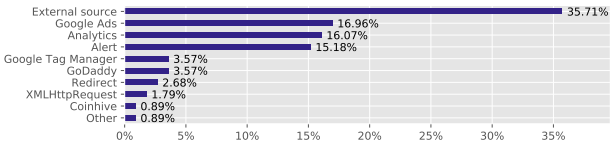


Figure 8: Distribution of JavaScript on 2018-12-31.

```
$a=(new-object net.webclient);
$b=$Env:APPDATA;
$w=$Env:WINDIR;
$c=$b+\'//t.txt\';
$g=$b+\'//t.exe\';
$p=$w+\'//Microsoft.NET//Framework\';
if (gci -Path $p | where {$_.Name -like \'v4*\'}) {
    try {$a.DownloadFile(\'https://filebin.ca/<CODE A>\', $c);
         ren $c t.exe;
         start $g }
    catch {$a.DownloadFile(\'https://files.fm/down.php?i=<CODE B>\', $c);
           ren $c t.exe; start $g }
}
else {
    try {$a.DownloadFile(\'https://filebin.ca/<CODE C>\', $c);
         ren $c t.exe;
         start $g }
    catch {$a.DownloadFile(\'https://files.fm/down.php?i=<CODE D>\', $c);
           ren $c t.exe;
           start $g }
};
sleep 180;
rm $g
```

Figure 9: Malicious PowerShell code.

4.3.3. PowerShell. Finally, we found one case of PowerShell code, see Figure 9 (the actual download URLs have been replaced). This code was hosted by two domains between 2017-06-20 and 2018-06-22. VirusTotal marked the file as malicious. Interesting about the downloaded executable is that it will install a scheduled task to perform additional DNS lookups of the same TXT record, in essence *auto-updating itself via DNS*. This behavior is comparable with the DNSMessenger malware [7], which gathers PowerShell payloads via DNS TXT records.

5. Ethical Considerations

While the TXT records used in this research are, technically, publicly available, we have taken care not to expose information about the individuals, or companies, behind the domains that might expose security vulnerabilities caused by improper use of TXT records. This paper is meant as a learning experience, showing the security pitfalls related to TXT records, rather than blaming parties for their "misconfiguration." We have notified the domain owners with private keys in their TXT records and hope these keys will be revoked and removed from the records.

6. Discussion and Conclusions

In this paper, we explore the unstructured tail of TXT records to uncover uses of TXT records which might have security implications. While analyzing the dataset we became progressively aware of the pitfalls one will encounter when attempting such a task, which we will discuss in this section.

Our analysis, as well as work by Portier et al. [18], show that the majority of TXT records belongs to well defined use cases. Our work builds on this observation and we progressively remove clearly defined categories, an approach that allows us to classify 99.54% of the TXT records in our dataset. For the remaining TXT records, we have not been able to define clear categories, as these

remaining records are highly diverse, both semantically and syntactically. Any further analysis of this category would imply slow manual labor coupled with deep domain knowledge with a likely low "return of investment" in terms of identifying security-relevant records. Analyzing the tail of the TXT records is therefore not only a *needle in the haystack* problem, but it also becomes a *human intelligence* problem.

Still, our quest has not been fruitless. The use of public-private key pairs clearly points to flaws in using and understanding more sophisticated mechanisms such as DKIM. Also, looking for code-specific regular expressions brought to light examples of JavaScript injection and malware auto-updates, which indicate that the DNS is used as a form of malicious code delivery. Albeit rare, finding such samples enhances our understanding of malware behavior and DNS misuse. With DoH we expect this behavior to only increase: for instance, instead of relying on TXT records to deliver miner executables [12], attackers could also distribute cryptomining scripts in the browser [27].

Another major challenge is the lack of context. Our active measurement dataset allows us to perform a wide search of the tail of the TXT records. However, we are not able to see how these records are used in practice, or if they are associated with domain names used for malicious activities. We believe that context information could be provided, for example, by *passive DNS* data. Additionally, active DNS measurements cannot look beyond known labels (second level, in our case). The malicious activity might be 'hiding' at lower levels.

Finally, we note that the remaining 99.54% of the TXT records might not necessarily be secure. Our regular expressions explicitly accommodate for typos which we commonly see in the data ('sfp' instead of 'spf', for example), a common issue in IT operations [24]. The consequence of these human errors might be severe, since they might lead to a false sense of security or, e.g., broken email delivery. A large-scale quantification of these types of errors remains currently open for further analysis.

Mitigation techniques can be viewed in two ways, the first is how to prevent leakage of information through errors. The most common practice here is consistently monitoring the correctness of deployed records. This includes monitoring deployed records for potential information leakage. The second part is preventing, e.g., payload delivery via DNS. Especially with the rise of DoH, this becomes a major challenge for defenders in networks. While, e.g., blocking TXT records in DoH implementations for browsers might sound like a promising prospect at first, this only mitigates a part of the problem, as ultimately all RRs might be abused for malware delivery via DoH.

To support further research in this direction, we provide the 101 regular expressions we used for the classification of DNS TXT records on our website: `https://www.tide-project.nl/blog/wtmc2020`.

References

[1] D. Lopera, "Necurs Spam uses DNS TXT Records for Redirection," https://www.trustwave.com/en-us/resources/blogs/spiderlabs-blog/necurs-spam-uses-dns-txt-records-for-redirection/, 2019.

[2] OpenINTEL, https://www.openintel.nl/.

[3] R. van Rijswijk-Deij, M. Jonker, A. Sperotto, and A. Pras, "A High-Performance, Scalable Infrastructure for Large-Scale Active DNS Measurements," *IEEE Journal on Selected Areas in Communications (JSAC)*, 2016.

[4] P. Papadopoulos, E. Athanasopoulos, A. A. Chariton, and E. P. Markatos, "Where's Wally? How to Privately Discover your Friends on the Internet," *Proc. of the ACM Asia Conference on Computer and Communications Security (ASIACCS)*, 2018.

[5] Y. Jin, M. Tomoishi, K. Fujikawa, and V. P. Kafle, "A Lightweight and Secure IoT Remote Monitoring Mechanism Using DNS with Privacy Preservation," in *Proc. of the IEEE Consumer Communications & Networking Conference (CCNC)*, 2019.

[6] M. Anagnostopoulos, G. Kambourakis, P. Kopanos, G. Louloudakis, and S. Gritzalis, "DNS Amplification Attack Revisited," *Computers & Security*, 2013.

[7] E. Brumaghin and C. Grady, "Covert Channels and Poor Decisions: The Tale of DNSMessenger," https://blog.talosintelligence.com/2017/03/dnsmessenger.html, 2017.

[8] C. J. Dietrich, C. Rossow, F. C. Freiling, H. Bos, M. v. Steen, and N. Pohlmann, "On Botnets That Use DNS for Command and Control," in *Proc. of the European Conference on Computer Network Defense (EC2ND)*, 2011.

[9] J. White, "Pulling Back the Curtains on EncodedCommand PowerShell Attacks," https://unit42.paloaltonetworks.com/unit42-pulling-back-the-curtains-on-encodedcommand-powershell-attacks/, 2017.

[10] H. Ichise, Y. Jin, and K. Iida, "Analysis of DNS TXT Record Usage and Consideration of Botnet Communication Detection," *IEICE Transactions on Communications*, 2018.

[11] C. Mullaney, "Morto worm sets a (DNS) record," https://www.symantec.com/connect/blogs/morto-worm-sets-dns-record, 2011.

[12] Anomali Threat Research Team, "Illicit Cryptomining Threat Actor Rocke Changes Tactics, Now More Difficult to Detect," https://www.anomali.com/blog/illicit-cryptomining-threat-actor-rocke-changes-tactics-now-more-difficult-to-detect, 2019.

[13] H. Gao, V. Yegneswaran, J. Jiang, Y. Chen, P. Porras, S. Ghosh, and H. Duan, "Reexamining DNS From a Global Recursive Resolver Perspective," *IEEE/ACM Transactions on Networking (TON)*, 2016.

[14] D. Tatang, F. Quinkert, N. Dolecki, and T. Holz, "A Study of Newly Observed Hostnames and DNS Tunneling in the Wild," https://arxiv.org/abs/1902.08454, 2019.

[15] F. Streibelt, J. Böttger, N. Chatzis, G. Smaragdakis, and A. Feldmann, "Exploring EDNS-Client-Subnet Adopters in your Free Time," in *Proc. of the ACM Internet Measurement Conference (IMC)*, 2013.

[16] T. Fiebig, K. Borgolte, S. Hao, C. Kruegel, and G. Vigna, "Something From Nothing (There): Collecting Global IPv6 Datasets From DNS," in *Proc. of the International Conference on Passive and Active Network Measurement (PAM)*, 2017.

[17] T. Fiebig, K. Borgolte, S. Hao, C. Kruegel, G. Vigna, and A. Feldmann, "In rDNS We Trust: Revisiting a Common Data-Source's Reliability," in *Proc. of the International Conference on Passive and Active Network Measurement (PAM)*, 2018.

[18] A. Portier, H. Carter, and C. Lever, "Security In Plain TXT: Observing the Use of DNS TXT Records in the Wild," in *Proc. of the International Conference on Detection of Intrusions & Malware, and Vulnerability Assessment (DIMVA)*, 2019.

[19] DirectAdmin, "Sending mail to hotmail, yahoo and gmail," https://help.directadmin.com/item.php?id=207, 2015.

[20] J. van Veen, "How to stop your webshop emails going to the spam or junk folder," https://joostvanveen.com/a-20/how-to-stop-your-webshop-emails-going-to-the-spam-or-junk-folder.

[21] OpenINTEL, "Current Coverage," https://www.openintel.nl/coverage/.

[22] Microsoft, "Exchange Server 2013: Configure a federation trust," https://docs.microsoft.com/en-us/exchange/configure-a-federation-trust-exchange-2013-help.

[23] HostExploit, "World Host Report," http://hostexploit.com/downloads/world_hosts_report_201403.pdf, 2014.

[24] C. Dietrich, K. Krombholz, K. Borgolte, and T. Fiebig, "Investigating System Operators' Perspective on Security Misconfigurations," in *Proc. of the ACM Conference on Computer and Communications Security (CCS)*, 2018.

[25] J. T. Bennett, "Shellshock in the Wild," https://www.fireeye.com/blog/threat-research/2014/09/shellshock-in-the-wild.html, 2014.

[26] SkullSecurity, "Stuffing Javascript into DNS names," https://blog.skullsecurity.org/2010/stuffing-javascript-into-dns-names, 2010.

[27] R. K. Konoth, E. Vineti, V. Moonsamy, M. Lindorfer, C. Kruegel, H. Bos, and G. Vigna, "Minesweeper: An in-depth look into drive-by cryptocurrency mining and its defense," in *Proc. of the ACM Conference on Computer and Communications Security (CCS)*, 2018.