Longitudinal Analysis of Wildcard Certificates in the WebPKI

David Hasselquist*†, Ludvig Bolin*, Emil Carlsson*, Adam Hylander*,
Martin Larsson*, Erik Voldstad*, Niklas Carlsson*
*Linköping University, Sweden
†Sectra Communications, Sweden

***Abstract*—The use of wildcard certificates and multi-domain certificates can impact how sensitive a certificate is to attacks and how many (sub)domains and machines may be impacted if a private key is compromised. Unfortunately, there are no globally agreed-upon best practices for these certificate types and the recommendations have changed many times over the years. In this paper, we present a 10-year longitudinal analysis of the usage of wildcard certificates and multi-domain certificates on the internet. Our analysis captures and highlights substantial differences in the heterogenous wildcard and multi-domain certificate practices. The results also show that there are several ways that CAs and domain owners have chosen to improve their practices, with many appearing to reduce the number of domains (and subdomains) for which each certificate is responsible.**

I. Introduction

X.509 certificates and the trust that we place in them are at the heart of internet security. However, not all certificates are managed the same and can be trusted the same. Two often forgotten aspects of certificate management that play big roles in how sensitive a certificate is to attacks and how many subdomains (and machines) that may be impacted if a private key is compromised are (1) the use of wildcards and (2) how many domains are included using the subject alternative name (SAN) extension of the certificate.

Certificate validation: At a high level, an X.509 certificate issued by a Certificate Authority (CA) can be used by a webserver as proof that the domain/subdomains that it belongs to is in ownership of the private key associated with the public key found in the certificate. If a browser trusts the certificate, the browser will allow the user to access the website as normal.[1] However, if the certificate cannot be trusted or does not validate the server's domain, the client is stopped from entering the webpage (or warned not to). The exact set of subdomains that a certificate validates is therefore important.

Wildcards and multi-domain certificates: To allow a certificate to validate several domains, certificates sometimes use wildcards (e.g., *.example.com would allow any subdomain of example.com to use the certificate), list several domains in the SAN extension, or both. Unfortunately, there is no agreed upon best practice and the recommendations and best-practices around wildcards and multi-domain certificates have changed many times over the years. For example, since initially being introduced in RFC 882 [1], there have been many revisions to how wildcards are supported, including some larger revisions seen in RFC 4592 [2]. With a lack of agreed upon best practices, we have observed that different CAs and domain owners have taken highly different approaches and that these approaches have changed over time. With these choices, combined with how the corresponding certificates and keys are managed, having significant security implications, it is important to understand both the current patterns and trends in the usage of wildcards and multi-domain certificates.

Contributions: To address the above gap in the research literature, in this paper, we present a longitudinal analysis of the usage of wildcard certificates and multi-domain certificates on the internet over the past 10 years. For our primary analysis, we collect and use bi-annual snapshots of the newly issued certificates submitted to the most popular certificate transparency (CT) logs [3]. For this dataset, we then compare subsets of certificates along five different dimensions: (1) the domain popularity of the domains that a certificate validates, (2) the CA issuing the certificate, (3) the certificate validation type, (4) the validity period, and (5) the key type of the public key of the certificate. To obtain a broader perspective on the overall trends, we also collect and analyze certificate data extracted from all certificates listed in the database of crt.sh [4] for a handcrafted selection of popular domains and the certificates observed in the scans performed by Rapid7 [5]. Our analysis provides valuable insights into the heterogenous nature of current wildcard and multi-domain certificate practices.

Example findings: Our analysis captures and highlights substantial differences in wildcard and multi-domain certificate practices. While some of these differences appear to be related to policy differences of individual CAs and how the CAs' practices have changed over time, other differences cannot. Instead, we observe how certain subsets of certificates (e.g., based on domain popularity or the certificate's validation type, validity duration, or the key type) appear to employ quite different wildcard strategies, and that these practices can change quickly from year to year. Yet, most subsets have increased their wildcard usage over the last few years, while simultaneously reducing the number of domains that they include in the SAN and the number of wildcards that they include per wildcard certificate. Overall, our analysis demonstrates that CAs and domain owners may employ several strategies to improve their

[1] For a browser to trust a certificate, the certificate must be signed by a root certificate stored in a browser's trust store or an intermediate certificate linked through a verifiable certificate chain to a trusted root certificate.

practices by reducing the number of domains/subdomains per certificate, and hence each certificate's attack surface.

Outline: After introducing the datasets (§ II), we present a high-level analysis of three datasets (§ III), before looking closer at the impact of five different factors (§ IV to § VIII). Finally, we present related work (§ IX) and conclusions (§ X).

II. Datasets and Summary Stats

For our analysis, we use one primary dataset (CT logs [3]) and two complementing datasets (crt.sh [4] and Rapid7 [5]). We next describe each dataset and how they were collected.

A. Datasets

CT log dataset: Certificate Transparency (CT) was introduced in 2013 to improve web security and reduce the risk of certificate misuse [3]. The use of CT has been widely adopted and most browsers today require a certificate to be logged in publicly append-only CT logs. For example, Google Chrome requires all new certificates to be logged in two or three distinct and recognized CT logs [6], out of which at least one needs to be operated by Google. Most certificates are therefore logged in Google's CT logs. For this reason, we use the certificates from Google's widely used Argon logs (sharded into several logs based on the expiry dates) for our analysis. For older certificates, we use Google's log Pilot, Aviator, and Rocketeer, going back to the start of the log usage in 2013.

To limit the dataset, we collect data from two weeks each year between 2013 and 2022. Based on the findings by Korzhitskii and Carlsson [7], we choose weeks 6 and 32 as they fairly represent a normal week in terms of deviations and activity, and do not coincide with any major events or holidays. To get a fair comparison and only show the most relevant certificates, we remove all certificates logged later than one month after they have been issued. In total, our CT log dataset (2013–2022) consists of over 197M certificates.

Crt.sh dataset: Crt.sh [4] is an interactive tool where users can search for certificates in CT logs. Currently, crt.sh monitors 43 CT logs ranging from large operators like Google to smaller operators like TrustAsia. In our study, we use data from crt.sh as a complimentary dataset for a popularity-based domain comparison. Using Certwatch [8], we access the crt.sh database and extract certificates for 500 domains based on the popularity of the domains on the Tranco ranking. In total, our crt.sh dataset (2013–2021) consists of over 6.2M certificates.

Rapid7 dataset: Rapid7 is a security company with a long history of developing many of the industry's most used forensic and ethical hacking tools [5]. Using the data provided through their Open Data initiative [5], we extract the certificate and wildcard usage on the web. In total, our Rapid7 dataset (2013–2020) consists of over 105M certificates.

B. Summary statistics

Table I summarizes the number of wildcard certificates of different types observed in each dataset. Here, we distinguish between certificates that have a wildcard in the SAN extension, in the subject field, or in at least one of the two places. Looking at the overall wildcard usage, we observed the highest relative usage in the crt.sh dataset (49.1%), followed by the CT log dataset (18.2%), and finally the Rapid7 dataset (4.7%). The big differences can be explained by the sampling techniques of the different datasets and how that relates to the biases we found in the wildcard usage seen within different subsets of the certificates. For example, the crt.sh dataset is heavily skewed towards the most popular domains, many of which are using much more wildcards than a random certificate. Similarly, the difference between the Rapid7 and the CT log datasets can be attributed to Rapid7 missing many multi-domain certificates (many sharing CDN infrastructure) and being somewhat older (we see an increase in wildcard certificates the past few years).

TABLE I: Summary statistics for our three datasets.

	CT log	**crt.sh**	**Rapid7**
# certificates in dataset	197 545 653	6 221 376	105 568 228
# certificates with wildcard in SAN	35 366 096	3 052 845	4 690 749
# certificates with wildcard in subject	15 608 533	2 555 871	3 382 763
# certificates with wildcard somewhere	36 007 424	3 053 086	4 923 358

When comparing the three different rows, the biggest differences in certificates in subject vs. in total were observed for the CT log dataset: 7.9% vs. 18.2%. For the others the differences are 41.1% vs. 49.1% (crt.sh) and 3.2% vs. 4.7% (Rapid7). Again, these differences can be explained by the relative biases of the sample sets. We next look closer at how the use of the wildcards have changed over time.

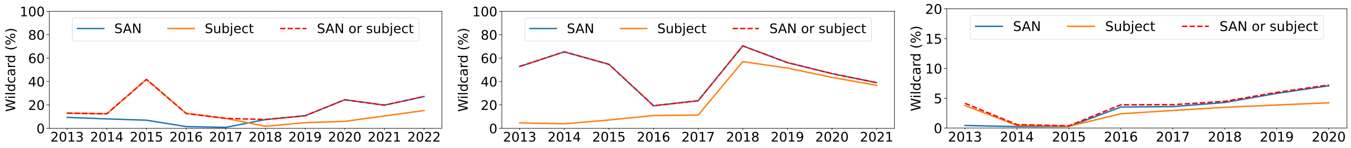
III. High-level longitudinal analysis

Figure 1 shows the yearly percentage of certificates for each of the three wildcard types. Here, we note that the SAN extension has been mandatory during our study (decided by the CA/Browser forum in 2012 [9]) and that Chrome removed support for validating against the subject in 2017 [10], effectively making the SAN extension the only option.

For both the CT log and the Rapid7 dataset, we observe a clear shift from most wildcards being in the subject to being found in the SAN. In fact, for the past few years, all wildcard certificates have indeed had wildcards in their SAN. This shift matches well with Chrome removing support for validating against the subject in 2017 [10]. Yet, the non-negligible number of wildcards seen in the subject is interesting.

Also, the average number of domains per SAN differs substantially between the datasets. These yearly values are shown in Figure 2, and we again note that the differences can be explained by biases in the sampling associated with each dataset (e.g., crt.sh contains mostly certificates of popular domains and the CT logs perhaps providing the most representative sampling of a random certificate).

In the following sections, we use the CT log dataset to evaluate the impact of several underlying factors on the observed differences in the usage of wildcards and multi-domain certificates. In particular, we consider the impact of the domain popularity, which CA issued the certificate, the certificate validation type, the validity period, and the certificate key type.



(a) CT log

(b) crt.sh

(c) Rapid7 (note y-axis)

Fig. 1: Yearly wildcard usage for the three datasets.

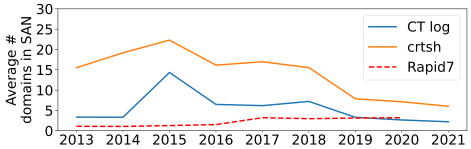


Fig. 2: Yearly average number of domains in SAN per dataset.

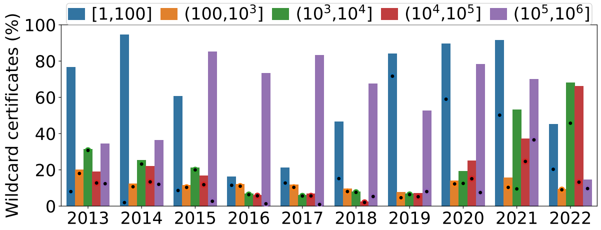


Fig. 3: Percentage of wildcard certificates issued for domains with different popularity using CT log dataset. We distinguish between wildcards in SAN (bars) or in the subject (markers).

IV. Popularity-based Domain Comparison

To study differences in how the wildcard usage has changed over time for websites with different popularity, we use the rankings provided with the Tranco top-1M list. For our primary dataset (i.e., the CT log dataset), we simply check if a certificate would validate one of the domains on this ranking list (accounting for domain-name matching but ignoring validity periods, revocations, etc.), and in the case it does, we assign it to the popularity range including that domain's Tranco ranking. Here, we use the following popularity ranges: [1,10^2], (10^2,10^3], (10^3,10^4], (10^4,10^5], (10^5,10^6]. After doing this for all certificates, we calculate yearly statistics for each set of certificates belonging to the popularity categories of interest.

Differences in the proportion of wildcard certificates: Figure 3 shows the percentage of wildcards certificates for the different popularity categories. Here, we consider two types of wildcard certificates: (1) certificates with a wildcard in the SAN (shown as bars) and (2) certificates with wildcards in the subject field (shown as markers). Regardless of metric, we observe very big variations in the wildcard usage both over the years and across domains. Perhaps the most noticeable trend is that the most popular domains (ranks 1–100) were the biggest users of wildcards in the SAN both the first three years (2013–2015) and again towards the end of the period (2019–2021). Looking at the percentage of certificates with wildcards in the subject the trend is a bit different. Here, the most popular domains initially have the smallest fraction of wildcard certificates (2013–2014) but then quickly becomes the category with the biggest fraction (2016–2021).

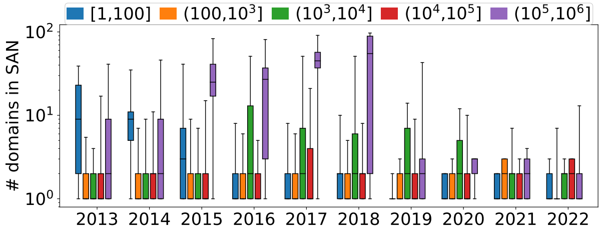
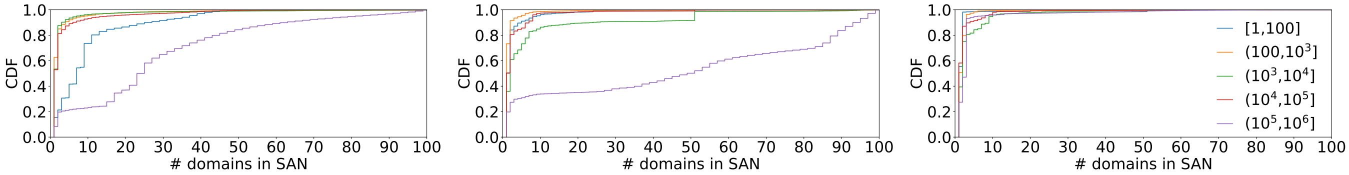


Fig. 4: SAN usage per popularity category and year.

Among the other popularity ranges, the patterns are less clear, although a few things stands out. First, the range (10^5,10^6] is the only category that has the highest yearly wildcard usage (2015–2018 in CT dataset and 2016–2017 in crt.sh dataset) as observed in the SAN. However, this popularity class never has the highest wildcard usage in the subject. Second, looking at the CT log dataset, for the intermediate popularity ranges (10^3,10^4] and (10^4,10^5], we see a steady increase in the wildcard usage in the SANs for the last four years (2019–2022). However, this trend is not consistent when looking at the wildcards in the subject field or when looking at only the last 100 domains in each range (e.g., as done in our crt.sh dataset or using this filtering on the CT log dataset). We therefore consider these observations less significant and instead use them to highlight that there are big variations in the wildcard usage between domains (as well as how many certificates are associated with each ranking range).

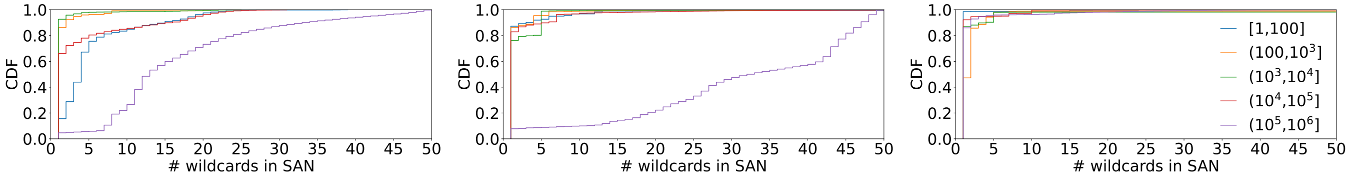
Popularity-based differences in number of domains per SAN: While we have observed a general reduction in the number of domains per SAN since 2018, there are some significant differences in how many domains each SAN contains for each popularity category and how this have changed over time. This is illustrated by the high variations in Figures 4 and 5.

Figure 4 presents a box-plot over the yearly statistics for each popularity category. Here, we show the 5^{th} percentile (bottom marker), 25^{th} percentile (bottom of box), median (middle marker), 75^{th} percentile (top of box), and 95^{th} percentile (top marker). We first note that the most popular domains (ranks 1–100) and the (10^5,10^6] category again stand out. Initially, for the first two year (2013–2014) the top-100 domains (i.e., ranks [1,100]) saw the highest number of domains per SAN, after which the certificates for domains with ranks (10^5,10^6] saw the highest for the next four years (2015–2018). Second, and perhaps more noticeable, is the trend that we have seen a reduction of the number of domains per SAN, starting at somewhat different times for the different popularity categories. For example, the general high-level trend (with some smaller variation) has been relatively consistent for the



(a) 2013 to 2015 (b) 2016 to 2018 (c) 2019 to 2021

Fig. 5: CDF of the number of domains in SAN for different popularity categories (split into three consecutive 3-year intervals).



(a) 2013 to 2015 (b) 2016 to 2018 (c) 2019 to 2021

Fig. 6: CDF of the number of wildcards in SAN for different popularity categories (split into three consecutive 3-year intervals).

top-100 domains over the full time period, whereas it can be seen as starting in 2016 for $(10^3,10^4]$ and in 2018 for $(10^5,10^6]$. For the other categories there are no clear trends as the number of domains per SAN have been relatively small throughout the time period, with only small variations here and there (including recent smaller spikes 2021 and 2022).

The high-level temporal trends are perhaps best summarized using the Cumulative Distribution Function (CDF) plots shown in Figure 5. Here, we show CDFs of the number of domains per SAN for different popularity categories as split over three 3-year intervals: 2013–2015, 2016–2018, and 2019–2021. We note that the CDFs of almost all popularity categories have been shifted towards the left as we go from the oldest 3-year period (2013–2015) to the most recent 3-year period (2019–2021). This observation is consistent also if extending the last time period to include 2022.

Popularity-based differences in wildcards per SAN: We see similar trends in the number of wildcards per SAN (for the certificates that have wildcards) as we observe for the number of domains per SAN of the different popularity categories. This is illustrated in Figure 6 using CDFs. We again note a clear shift towards the left (lower number of wildcards) when going from the first 3-year period (2013–2015) to the most recent (2019–2021). While we observe relatively more wildcards for the top-100 domains in the first period (2013–2015) the biggest wildcard usage is observed for the domains with rankings $(10^5,10^6]$ for each of the first six years (2013–2018) regardless of which percentile was considered. For the last four years there have been no clear trend as all popularity classes have seen reduced wildcard usage.

V. CA-based Analysis

We next consider the certificates issued by the eight most popular CAs observed in the CT log dataset.

Differences in the proportion of wildcard certificates issued by top CAs: Figure 7 shows the percentage of wildcards certificates in the SANs (bars) and in subject fields (markers) of the certificates issued each year by each of the top-8 CAs. To put these numbers in context, Figure 8 shows the fraction of all observed certificates that were issued by each of the eight CAs, as calculated for each of our snapshots (2 times per year). This plot clearly shows the introduction of Let's Encrypt (2016) and how it quickly took over as the top-CA, how DigiCert has seen a steady drop (from being the top-CA the first two years), and how CloudFlare took an increasing market share 2019–2020, just to see a significant drop in 2021.

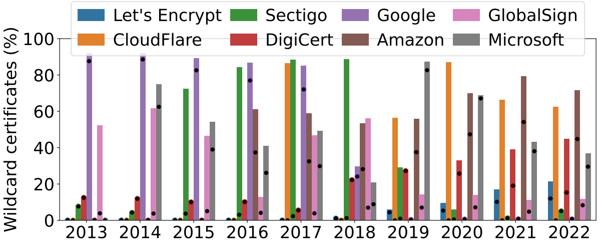


Fig. 7: Percentage of wildcards certificates in SAN (bars) and in subject field (markers) per CA.

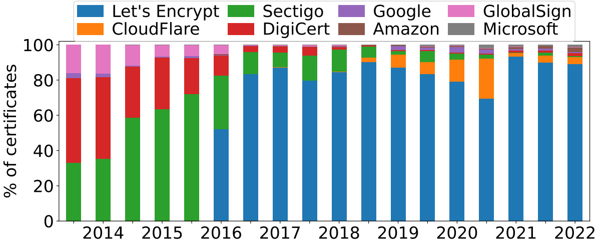
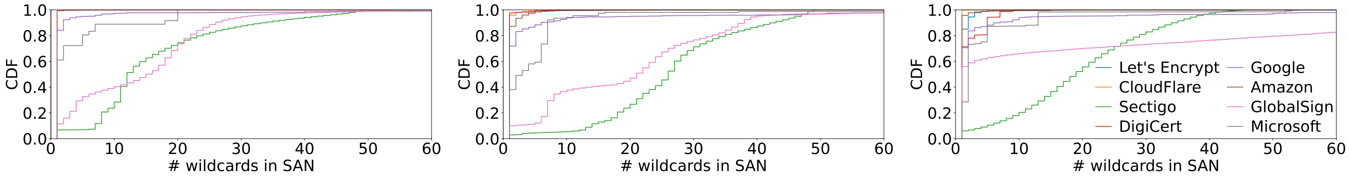


Fig. 8: Relative yearly certificate frequency per top-CA.

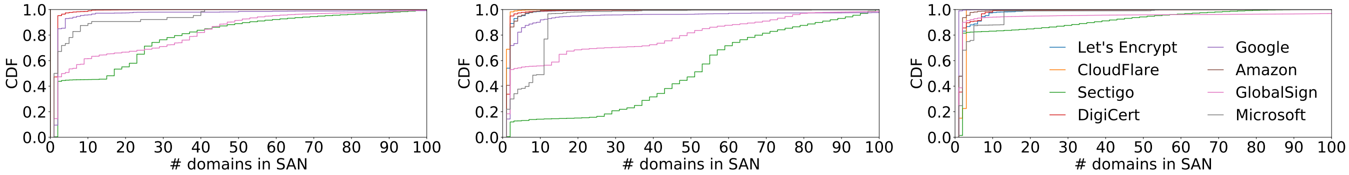
Now, looking at the wildcard usage, we note that Google went from having the highest (2013–2016) or third highest (2017) to quickly having the lowest (2019–2022) wildcard usage of all the CAs. This shows a clear policy shift in their wildcard usage. In contrast, CloudFlare and Amazon have had among the highest wildcard usage over this later time period (2019–2022). DigiCert and Let's Encrypt have seen perhaps the steadiest increases in overall wildcard usage (as seen in the SANs), with their increases starting in 2018.

Looking at the percentage of certificates with wildcards in the subject field, we note significant differences between these two CAs over this period, as DigiCert's wildcard usage in



(a) 2013 to 2015 (b) 2016 to 2018 (c) 2019 to 2021

Fig. 9: CDF of the number of wildcards in SAN for the top-8 CAs (split into three consecutive 3-year intervals).



(a) 2013 to 2015 (b) 2016 to 2018 (c) 2019 to 2021

Fig. 10: CDF of the number of domains in SAN for the top-8 CAs (split into three consecutive 3-year intervals).

the subject have gone down the last three years (2020–2022) while that of Let's Encrypt have increased at the same rate as in the SANs. Only CloudFlare appears to have avoided using wildcards in the subject field. While they consistently have had zero such certificates, the other top-CAs have always had some certificates with wildcards in the subject field (for the years we observed certificates for them). Of course, it is unclear how much should be read into the wildcards in the subject field seen these later years, as Google Chrome removed support for the subject field in 2017 [10]. Yet, it is interesting that the CAs clearly have handled the use of wildcards in this field very differently over this time period.

Wildcards per certificate: We also observe big variations in the number of wildcards per certificate. Figure 9 shows CDFs of the number of wildcards per SAN with at least one wildcard for three consecutive 3-year periods. Sectigo and GlobalSign again stand out, as they have by far the most wildcards per such certificate. Interestingly, both these CAs only show limited reductions the last three years. The case of GlobalSign is particularly interesting as we simultaneously observe an increase in the number of certificates with a single wildcard (following the y-axis to the left) and an increase in the number of certificates with many wildcards (e.g., 17% of the certificates in 2019–2021 had more than 60 wildcards). In contrast, the distributions remain much more similar over time for Sectigo, with only a small fraction of certificates having a single wildcard and a relatively s-shaped distribution spanning the full spectrum between (roughly) 1 to 50 wildcards.

Domains per SAN: Another way to increase the number of subdomains that a certificate can be used for is by including many domains in the SAN. Figure 10 shows CDFs of the number of domains per certificate for three consecutive 3-year intervals. While most CAs follow the general trend of a reduced number of domains per SAN and all CAs had a median below three for the last three years (2019–2021), two CAs consistently have used more domains per SAN: Sectigo and GlobalSign. Of these, Sectigo stands out the most, going from a median of 17 (2013–2015) to 51 (2016–2018) and now back to a median of two domains per SAN (2019–2021).

VI. Impact of Certificate Type

Certificates can generally be categorized into three different types, with validation checks done in accordance with Baseline Requirement (BR) [9]. The simplest checks are done for Domain Validated (DV) certificates. While many CAs do additional checks, DV certificates are typically issued after the domain owner has answered an email, authenticating its domain ownership. With Organization Validated (OV) certificates, the company or individual owner is manually validated by the CA. These certificates are typically recommended (as a minimum) for servers running e-commerce transactions [11]. Finally, Extended Validated (EV) certificates corresponds to the highest tier of validation. These certificates require additional validation steps and more precautions may be required to ensure their integrity [11], [12]. Naturally, EV certificates are the most expensive certificates and DV certificates can these days be obtained for free (e.g., via Let's Encrypt). In the past, Individual Validated (IV) certificates have also been observed. However, we did not observe any such certificates in our dataset and therefore exclude them from our discussion.

To determine the certificate type of each certificate, we use the Object Identifiers (OIDs), described in the BRs [9] and in the EV SSL Certificate Guidelines [12]. In the case that a certificate is missing an OID matching one of the known types, we assign it to the "other" category.

Fraction of wildcard certificates per certificate type: Figure 11 shows the percentage of certificates with at least one wildcard in the SANs (bars) and in the subject fields (markers) broken down per certificate type. While we include an "other" category here, it primarily includes older certificates that our automated method did not classify. Based on prior work, we expect most of these to be DV certificates. The larger fraction of "other" among the early years can be seen in Figure 12, where we plot the fraction of all certificates in each snapshot (two per year) that were classified as each type. Ignoring the "other" category, we note several big changes in the fraction of certificates with wildcards among the OV certificates, with the biggest spike in 2020 reaching 80% compared to a low

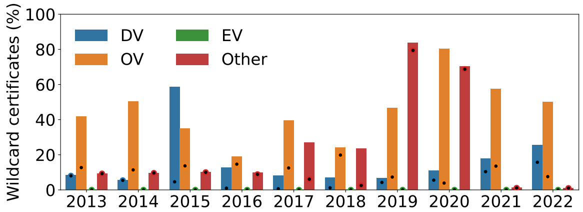


Fig. 11: Percentage of wildcards certificates in SAN (bars) and in subject field (markers) per certificate type.

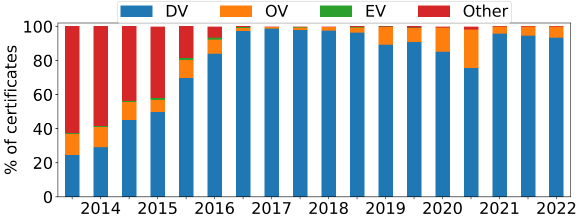
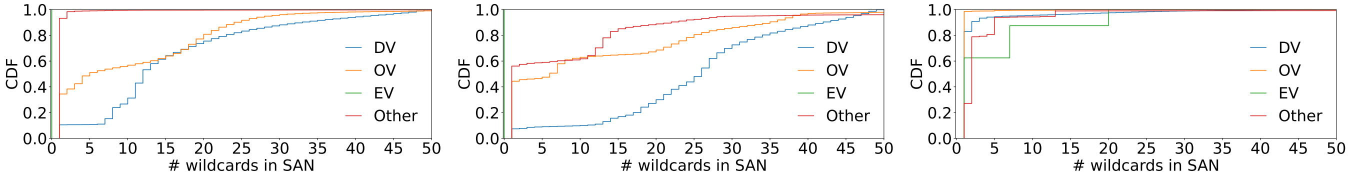


Fig. 12: Relative certificate frequency per certificate type.

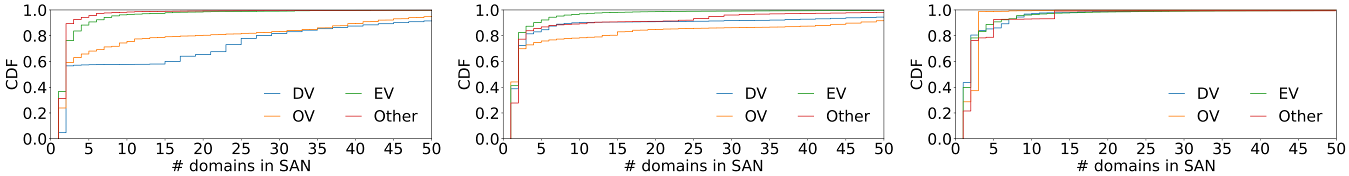


(a) 2013 to 2015

(b) 2016 to 2018

(c) 2019 to 2021

Fig. 13: CDF of the number of wildcards in SAN for each certificate type (split into three consecutive 3-year intervals).



(a) 2013 to 2015

(b) 2016 to 2018

(c) 2019 to 2021

Fig. 14: CDF of the number of domains in SAN for each certificate type (split into three consecutive 3-year intervals).

of 19% in 2016. With exception to a big one-year spike in 2015 (59%), we observe a relatively lower wildcard usage among DV certificates than OV certificates (e.g., lower usage in SAN in 9 out of 10 years and lower usage in subject in 9 out of 10 years). However, for DV certificates we do observe a steady increase in wildcard usage over the past five years (starting 2018) whereas we have seen a slight drop in wildcard usage among the OV certificates over the past three years (since the peak in 2020). The non-visible wildcard usage among EV certificates (consistently below 0.04%) is due to EV certificates only offering wildcards for certificates for the `.onion` suffix (used by an anonymous onion service, known as "hidden service", reachable via Tor).

Wildcards per certificate: Figure 13 shows CDFs of the number of wildcards in the SAN per certificate type for three consecutive 3-year intervals of the CT log dataset. Here, we again focus on the DV (blue) and OV (orange) curves, as the EV curve only captures the behavior of a very small number of `.onion` certificates. For both DV and OV certificates, we note a big decrease in the number of wildcards per certificates over the last 3-year period. For example, for the first 3-year period, more than 20% of the wildcard certificates belonging to these two types included more than 20 wildcards, whereas the 80^{th} percentile now have shifted to the value 1. For DV certificates, 83% of the wildcard certificates now only have a single wildcard in SAN, and for OV certificates, 98.6% of the wildcard certificates only have a single wildcard. This shows a clear trend in the usage of less wildcards per certificate.

Domains per SAN: We also observe a decrease in number of domains per SAN for both the DV and OV types. This is illustrated in the CDFs shown in Figure 14. Here, we note that EV certificates consistently have used a small number of domains per certificates (curve pushed towards the left) but that DV and OV certificates have reduced the number of domains per SAN they include substantially and now (for the last 3-year period) have a very similar distribution as the EV certificates. In fact, the distributions of EV and DV are very close to each other and the tail of the OV curve is shifted even further to the left (i.e., lower numbers).

VII. Impact of Validity Period

The maximum validity period of a certificate is determined by the difference between the first date it is valid (seen in the notBefore field) and the last day it is valid (seen in the notAfter field) [13]. Here, we note that the notBefore time does not necessarily coincide with the issue date, and the validity periods reported should best be seen as approximations (especially as the issue time are not included in the certificates). Having said that and noting that the procedures can differ somewhat between CAs, we note that the issuing date typically is the same or very near the date set in the notBefore field.

Since most browsers (or other clients) do not check whether every certificate has been revoked, the expiry date of the certificate act as a type of failsafe (in that compromised certificates cannot be used longer than their expiry date). For this and other related reasons, it has long been argued for the use of shorter validity periods. However, it is not until recently that the browser vendors started to require use of validity periods no longer than 398 days, when Apple (March 2020), Chrome (June 2020), Mozilla (July 2020) and the finally BR

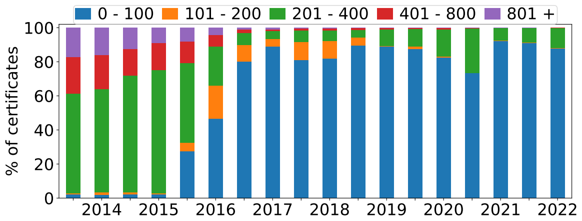


Fig. 15: Relative certificate frequency per validity period.

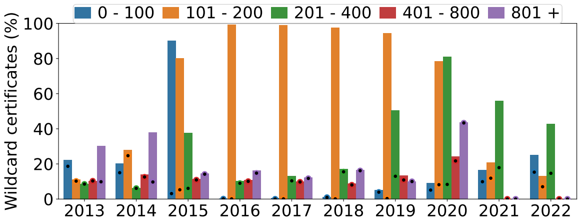
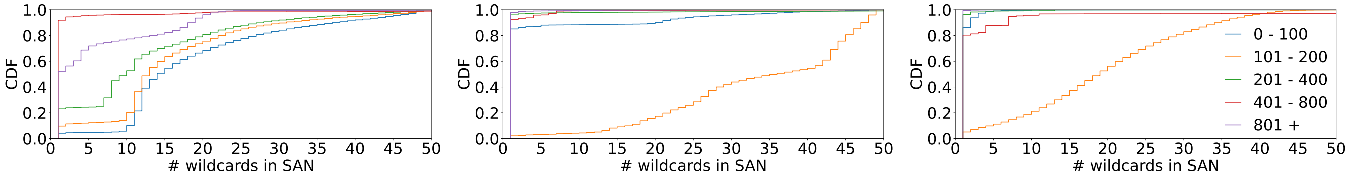


Fig. 16: Percentage of certificates with wildcard in SAN (bars) and in subject field (markers) per validity period (in days).

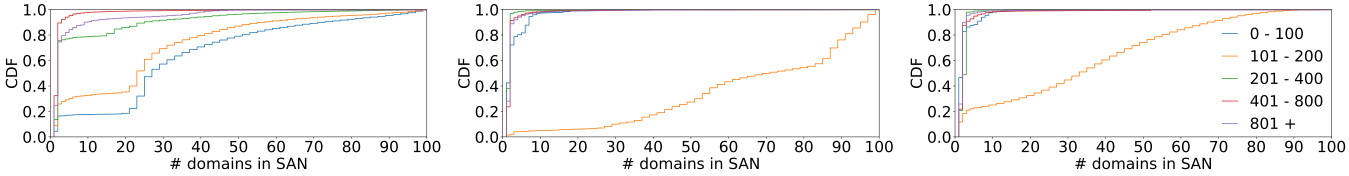


(a) 2013 to 2015

(b) 2016 to 2018

(c) 2019 to 2021

Fig. 17: CDF of the number of wildcards in SAN per validity period category (split into three consecutive 3-year intervals).



(a) 2013 to 2015

(b) 2016 to 2018

(c) 2019 to 2021

Fig. 18: CDF of the number of domains in SAN per validity period category (split into three consecutive 3-year intervals).

(July 2020) decided to reduce the maximum allowed lifetime of newly issued certificates to 398 days (instead of 825 which was the maximum allowed before that).

The push towards shorter validity periods is clearly visible in Figure 15, where we show the relative frequency that certificates with different validity periods were issued and observed in the CT log dataset each snapshot (2 per year). Here, we see that the certificates with validity periods of more than 800+ days quickly were pushed out, followed by the 400–800 day certificates being pushed out. However, since the introduction of Let's Encrypt, or for the past six years (2016–2022), certificates with validity periods less than 100 days have dominated. Most of these certificates have a validity period of 90 days and are issued by Let's Encrypt (who always use a 90-day validity period). The statistics for the certificates with validity period of less than 100 days therefore very much resembles those of Let's Encrypt.

Fraction of wildcard certificates associated with different validity periods: Figure 16 shows the percentage of certificates with wildcards in the SAN (bars) and in the subject field (markers). We note that neither of the two certificate classes with validity periods that no longer are allowed (i.e., 800+ and 400–800) stands out over the periods they saw substantial usage. Of the other classes, the middle range (i.e., 101–200) has seen the most dramatic changes with a big increase in fraction of wildcard certificates between 2015 and 2020, coinciding with the period of its biggest usage (Figure 15). While most of these certificates are issued by Sectigo under the name COMODO, other CAs significantly contributed to this class and its outstanding behavior. For example, during 2015–2020, COMODO issued certificates were responsible for between 54–98% of the yearly certificates in this category.

Wildcards and domains per certificate: Figures 17 and 18 show the CDFs of the number of wildcards in the SAN per wildcard certificate and the number of domains in the SAN per certificate, respectively. In both cases, we see a clear trend that both the number of wildcards per certificate and the number of domains in the SAN per certificate have reduced over time for all categories except the certificates with a validity period of 101–200 days. While the number of wildcards per wildcard certificate has gone up slightly for the certificates with validity periods of 401–800 days, we note that this subset represents a very small number of certificates in the last three years (2019–2021). We therefore caution from placing too much judgment on these certificates but note that these certificates also were issued by CAs that were among the last to issue certificates with a validity period longer than the 398-day cap.

VIII. Impact of Key Type

The strengths of the keys being used can provide some indication of the security concerns of the websites using them. For this analysis, we first split the most commonly observed key types into five classes from most to the least secure: (1) ECDSA 384, (2) RSA 4096, (3) ECDSA 256/RSA 3072, (4) RSA 2048, and (5) RSA 1028. We note that we observed consistently low usage of the most secure key (i.e., ECDSA 384) in all snapshots (less than 0.04%) and only a few instances of the least secure key (i.e., RSA 1024) in the initial two snapshots. Instead, the majority of the certificates use keys

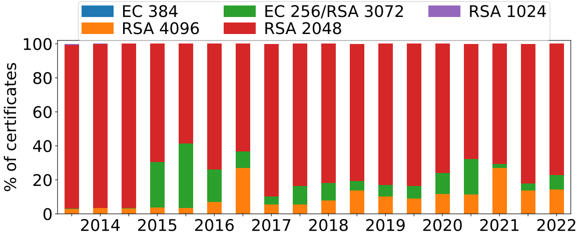


Fig. 19: Relative certificate frequency per public key cipher.

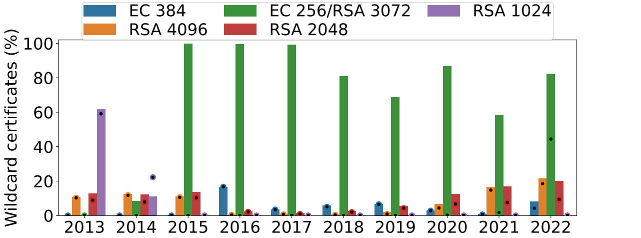
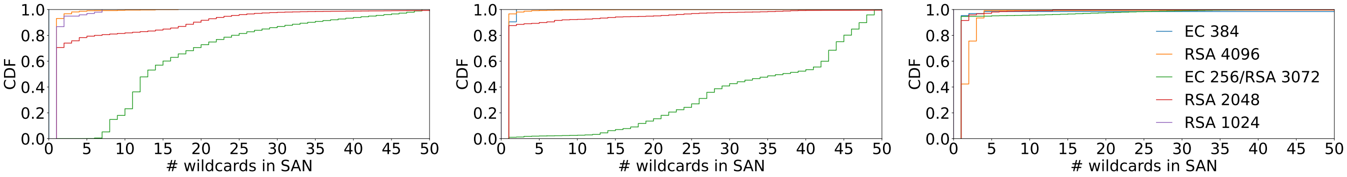


Fig. 20: Percentage of certificates with wildcards in SAN (bars) and in subject field (markers) per public key cipher.

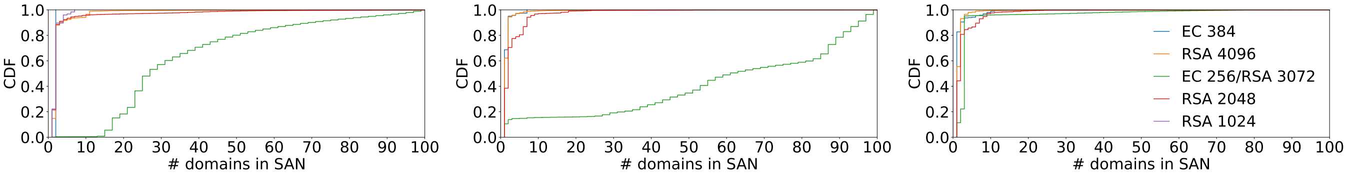


(a) 2013 to 2015

(b) 2016 to 2018

(c) 2019 to 2021

Fig. 21: CDF of the number of wildcards in SAN for certificates with public key ciphers of different security levels.



(a) 2013 to 2015

(b) 2016 to 2018

(c) 2019 to 2021

Fig. 22: CDF of the number of domains in SAN for certificates with public key ciphers of different security levels.

with the three intermediate security levels. Figure 19 shows a breakdown for each snapshot (2 per year). We note that RSA 2028 consistently has been the most common key type being used, although the share of the other two have together made up between 20–40% since around 2015, and the relative share of RSA 4096 has gone up relatively steady over this period (with exceptions for two spikes: late 2016 and early 2021).

Fraction of wildcard certificates associated with different key types: Focusing on the three main classes observed, we find the class with ECDSA 256/RSA 3072 certificates to consistently use the largest fraction of wildcard certificates since it started seeing significant usage (2015). The percentages of wildcard certificates of each class are shown in Figure 20. While the fraction of wildcard certificates of this category has gone down somewhat since 2015 (when it was close to 100%), the fraction have stayed above 58% (2021) all years since then. In contrast, the certificates with RSA 2048 and RSA 4096 keys have never seen a bigger fraction of wildcard certificates than 21% (2022). Having said that, both these subsets of certificates have seen a steady increase in wildcard usage since 2017. The usage of wildcard certificates among the middle category (ECDSA 256/RSA 3072) is dominated by COMODO/Sectigo (2015–2018) and CloudFlare (2019–2022) issued certificates.

Wildcards and domains per certificate: Figures 21 and 22 show CDFs of the number of wildcards in SAN per wildcard certificate and the number of domains in SAN per certificate, respectively. The certificates with the keys with the mid-most ranked security level (i.e., ECDSA 256/RSA 3072) again stands out from the rest. For other categories, the number of wildcards in SAN per wildcard certificate and the number of domains in SAN per certificate is relatively low, with medians of either 1 or 2 for each category and time interval. In contrast, the numbers are much greater for this category, with CDFs substantially shifted to the right (bigger values) and the median number of wildcards per wildcard certificate being 13 (2013–2015) and 37 (2016–2018) for the first two time periods, and then quickly reducing to one (2019–2021) for the last 3-year period. Similarly, the median number of domains per certificate for this category is 27 (2013–2015) and 61 (2016–2018) for the first two time periods, and quickly reduce to three (2019–2021) for the last 3-year period. These observations show that even if the wildcard usage still is higher among this category, the number of wildcards these certificates use per wildcard certificate and the number of domains they include in SAN per certificate has substantially confirmed to the lower values seen for the other certificates the last three years.

IX. Related Work

With almost all web communication today using HTTPS, certificate handling has become a very important topic.

Certificate discrepancies: Kumar et al. [14] study the lacking practices at CAs by examining CA issued certificates and their compliance with the BR. While larger CAs mostly issue correct certificates, many mid-sized CAs issue certificates with varying errors, and some smaller CAs have non-conforming problems in every issued certificate. Examples include failure to fully populate the SAN extension, encoding the wrong data, or inclusion of invalid DNS names. Bruhner et al. [15] study the effects of discrepancies in certificate replacement policies among the top-issuing CAs but do not report results for wildcard usage. Heinl et al. [16] evaluate CA trustworthiness using a set of objective criteria but only evaluate four CAs.

CT logs: Gasser et al. [17] study security practices of CAs using CT data and internet measurements. By tracking certificates in CT logs and comparing them to the BR, they conclude that more than 600M CT log entries (approximately 907k certificates) violate the BR. Various other works have characterized the CT logs themselves [7], [18]–[20]. Some works suggest that CT logs are too transparent as they may reveal sensitive information about certificate holders. Scheitle et al. [21] study the leakage of Fully Qualified Domain Names (FQDN) in CT logs, and demonstrate that adversaries may use CT logs to effectively discover new FQDNs to attack.

Wildcard certificates: The use of wildcard certificates may be beneficial against the FQDN attacks as new subdomains can be hidden in the CT logs. On the other hand, wildcards may also impose a security risk. Some works leverage wildcard certificates to perform various attacks [22]–[24]. Brinkmann et al. [25] argue that wildcard certificates can enable cross-protocol attacks. As a single wildcard certificate can validate multiple subdomains, a MITM attack can redirect user traffic meant to another subdomain, violating the TLS authentication and opening up attack vectors.

Certificate revocation: Other works have focused on certificate revocation [26]–[29]. For example, Liu et al. [26] show that a large percentage of served certificates are revoked and that browsers need to do more to verify the revocation status before serving the website. Korzhitskii and Carlsson [28] present a longitudinal analysis focusing on certificate revocation statuses and show several shortcomings in current revocation handling within and between CAs. While this shows an increased attack window for compromised certificates, these works do not consider the impact or use of wildcards.

X. Conclusions

In this paper, we have presented a longitudinal analysis of the wildcard usage on the internet over the past 10 years. The work highlights substantial differences in wildcard practices between different subcategories of certificates that cannot be attributed only to individual CAs or to policy suggestions by browser vendors and other big players in the WebPKI landscape. Instead, we have observed different subsets slowly conforming along two primary dimensions (i.e., number of wildcards per wildcard certificate or number of domains per SAN) while they still may employ quite different wildcard strategies and that these practices can change quickly from year to year. This shows that there are several ways that CAs and domain owners choose to improve their practices and reduce the number of domains/subdomains in each certificate. Regardless if the attack surface is reduced by decreasing the number of wildcards per certificate or the number of subdomains included in the SANs, these results show that the CAs and domain owners' efforts overall are reducing the number of subdomains that may be impacted by an attacker that obtains the private key for the wildcard certificate or multi-domain certificate. Here, reduced wildcard usage plays a particularly important role as wildcards can even allow attacks against subdomains that do not yet exist.

Acknowledgment

This work was partially supported by the Swedish Foundation for Strategic Research (SSF) and the Wallenberg AI, Autonomous Systems and Software Program (WASP) funded by the Knut and Alice Wallenberg Foundation.

References

[1] P. Mockapetris, "Domain names," RFC 882, 1983.
[2] E. Lewis, "The role of wildcards in the domain name system," RFC 4592, 2006.
[3] B. Laurie *et al.*, "Certificate transparency," RFC 6962, 2013.
[4] Sectigo Limited, "Certificate search," https://crt.sh, 2023.
[5] Rapid7, "Open data," https://opendata.rapid7.com, 2023.
[6] Google, "Chrome certificate transparency policy," https://googlechrome.github.io/CertificateTransparency/ct_policy.html, 2023.
[7] N. Korzhitskii and N. Carlsson, "Characterizing the root landscape of certificate transparency logs," in *Proc. IFIP Networking*, 2020.
[8] Crtsh, "certwatch_db," https://github.com/crtsh/certwatch_db, 2023.
[9] CA/Browser forum, "Baseline requirements for the issuance and management of publicly-trusted certificates," 2022.
[10] J. Medley, "Deprecations and removals in chrome 58," https://developer.chrome.com/blog/chrome-58-deprecations/, 2017.
[11] CA/Browser forum, "Information for the public," https://cabforum.org/info-for-consumers, 2022.
[12] ——, "EV SSL Certificate Guidelines," 2022.
[13] "Internet X.509 public key infrastructure certificate and certificate revocation list (CRL) profile," RFC 5280, 2008.
[14] D. Kumar, Z. Wang, M. Hyder, J. Dickinson, G. Beck, D. Adrian, J. Mason, Z. Durumeric, J. A. Halderman, and M. Bailey, "Tracking certificate misissuance in the wild," in *Proc. IEEE S&P*, 2018.
[15] C. M. Bruhner, O. Linnarsson, M. Nemec, M. Arlitt, and N. Carlsson, "Changing of the guards: Certificate and public key management on the Internet," in *Proc. PAM*, 2022.
[16] M. P. Heinl, A. Giehl, N. Wiedermann, S. Plaga, and F. Kargl, "MERCAT: A metric for the evaluation and reconsideration of certificate authority trustworthiness," in *Proc. CCSW*, 2019.
[17] O. Gasser, B. Hof, M. Helm, M. Korczynski, R. Holz, and G. Carle, "In log we trust: Revealing poor security practices with certificate transparency logs and internet measurements," in *Proc. PAM*, 2018.
[18] J. Gustafsson, G. Overier, M. Arlitt, and N. Carlsson, "A first look at the CT landscape: Certificate transparency logs in practice," in *PAM*, 2017.
[19] C. Nykvist, L. Sjöström, J. Gustafsson, and N. Carlsson, "Server-side adoption of certificate transparency," in *Proc. PAM*, 2018.
[20] N. Korzhitskii, M. Nemec, and N. Carlsson, "Postcertificates for revocation transparency," Tech report (arXiv), 2022.
[21] Q. Scheitle *et al.*, "The rise of certificate transparency and its implications on the Internet ecosystem," in *Proc. IMC*, 2018.
[22] M. Marlinspike, "More tricks for defeating SSL in practice," 2009.
[23] R. Roberts, Y. Goldschlag, R. Walter, T. Chung, A. Mislove, and D. Levin, "You are who you appear to be: A longitudinal study of domain impersonation in TLS certificates," in *Proc. ACM CCS*, 2019.
[24] M. Zhang *et al.*, "Talking with familiar strangers: An empirical study on https context confusion attacks," in *Proc. ACM CCS*, 2020.
[25] M. Brinkmann, C. Dresen, R. Merget, D. Poddebniak, J. Müller *et al.*, "ALPACA: Application layer protocol confusion-analyzing and mitigating cracks in TLS authentication." in *Proc. USENIX Security*, 2021.
[26] Y. Liu, W. Tome, L. Zhang, D. Choffnes, D. Levin, B. Maggs, A. Mislove, A. Schulman, and C. Wilson, "An end-to-end measurement of certificate revocation in the web's PKI," in *Proc. IMC*, 2015.
[27] J. Larisch, D. Choffnes, D. Levin, B. M. Maggs, A. Mislove, and C. Wilson, "CRLite: A scalable system for pushing all TLS revocations to all browsers," in *Proc. IEEE S&P*, 2017.
[28] N. Korzhitskii and N. Carlsson, "Revocation statuses on the Internet," in *Proc. PAM*, 2021.
[29] A. Halim, M. Danielsson, M. Arlitt, and N. Carlsson, "Temporal analysis of X.509 revocations and their statuses," in *IEEE EuroS&PW*, 2022.