Investigating Certification Authority Authorization Records' Effect on Existing Certificates

Till Pinke
University of Twente
P.O. Box 217, 7500AE Enschede
The Netherlands
t.e.pinke@student.utwente.nl

ABSTRACT

Due to attacks on Certificate Authorities undermining the security provided by TLS certificates, auditing frameworks are gaining traction. Two of these are Certificate Transparency, which publicly display certificate issuances, and Certificate Authority Authorization Records that document the authorities permission to issue certificates to domains. This paper aims to investigate how existing certificates are affected by new CAA records. We combine data from both CAA records and CT logs at scale to identify cases in which certificates are retroactively affected by updated CAA records. Then we check upon these anomalies with a TLS scan to investigate whether these certificates are still in use. We also investigate patterns and differences between CA operators and domain types regarding these occurrences. As there is little existing research in this area and CAA adoption has been relatively recent it is important to investigate edge cases in such a technology. We find that only 33% of all CAA updates affect certificates after they have been issued while 2.7% are retroactive and conflict with the issuer of the certificate. Among these anomalies the `.pl`, `.in` and `.io` top level domains appear more frequently as well as certificates issued by GoDaddy, GeoTrust and to a lesser extent GlobalSign and Amazon, while Let's Encrypt and CloudFlare are examples of CAs which appear very rarely among anomalies. Performing a TLS scan on identified cases reveals that the majority of certificates associated with these anomalies are no longer in use.

Keywords

CT Logs, CAA, HTTPS Security

1. INTRODUCTION

Nowadays TLS certificates are utilized in multiple applications to ensure safety for users, for example verifying the connection with a website. However, these certificates are not a perfect solution. The certificate storage on the user end could be poisoned, the issuing Certificate Authorities (CA) can be compromised, or certificates can be issued mistakenly due to exploits being abused [18]. In order to make the process more transparent and secure, multiple methods have been developed for this purpose. One of these technologies, which is gaining traction recently, are Certification Authority Authorization (CAA) records [15]. We investigate specific edge cases and quantify them by combining these two technologies for the first time. Our expectation is that future efforts can build on our research.

In short, CAA records are essentially public logs that indicate which CAs are authorized to issue certificates for a domain. Problems may arise when CAA records with contradicting information are added after a certificate has already been issued. For example the existing certificate's CA does not appear on the CAA records for the respective domain. This does not directly invalidate the certificate as only the CAA records at the time of issuance are relevant. Nonetheless, it is of interest to investigate these occurrences and analyse them based on various criteria, such as CAs, top level domains or number of CAs authorized by CAA records.

Thus, in this paper we study the following main research question: *How do CAA record adoption and policy changes affect existing certificates?* and split it up into the following three sub-questions.

1. How often does a new CAA policy affect a preexisting certificate?
2. Do the number of these occurrences differ between top level domains and CAs?
3. What happens to affected certificates afterwards?

In Section 2 an overview over the involved technologies is given in case the reader is not acquainted with them as well as provide relevant references for information on these topics. The tools used, process of combining the data sets and limitations associated with this are explained in Section 3 and the results of this are discussed in Section 4. In Section 5 we give an overview of related works and further readings on the topic of CAA records and CT logs. Finally we draw an overall conclusion in Section 7.

2. BACKGROUND

2.1 TLS Certificates

The Transport Layer Security (TLS) standard is widely used for providing secure communication over networks. It is based on chains of certificates utilizing public-key cryptography, as seen in figure 1. Any certificate is expected to be signed by a root certificate, which forms the basis of each certificate chain. The certificate used to sign traffic is the leaf certificate, titled end entity certificate in the given figure. When a certificate is checked for validity this chain is traversed backwards and the signatures are verified at each step.

34^{th} Twente Student Conference on IT Febr. 2^{nd}, 2020, Enschede, The Netherlands.

These certificates use the the X.509 standard which we will use to extract meta data about the certificates. This includes validity ranges, in the form of `not_before` and `not_after` fields, information about the issuer as well as a serial number to uniquely identify the certificate. The standard has more fields, however the mentioned ones are the most important.

2.2 CT Logs

To make the TLS certificate issuance more transparent Google has started the usage of CT logs [3] and is still considered the main driver in this technology field. As a result of this more CAs, such as CloudFlare and DigiCert, joined in on keeping logs of issued certificates [6]. These logs are a public append-only list of certificate issuances backed by Merkle Trees [17]. This means that every time a certificate is issued by a CA they make an entry in this public list. Once this is done it cannot be reverted and remains as a permanent record. The public availability of these logs enables third party auditors to verify and find misissuances of certificates. We will assume the role of such an auditor and thus utilize these logs as one of the main data sets.

There are two different ways of managing such a log. Firstly the log is kept open and certificates are collected without discrimination until the log is closed. This is the case for the two logs that we will utilize in this paper, namely the Google Rocketeer and Google Pilot logs. The second way is called temporal sharding, in which certificates are sorted into categorizes based on their expiry year. The combination of these smaller shard logs provides the full log. Examples of this are the Google Argon and DigiCert Yeti logs. The main advantage of the first method lies in its simplicity, but it does not scale very well. The larger a log becomes the more difficult it is to perform reasonable maintenance on them. As a solution to this temporal sharding distributes certificates across logs and defines specific cut offs at which a log is no longer continued to limit their growth.

The CT log standard can be found in almost any browser, for example with Chromium having the requirement of every TLS certificate to be present in an approved CT log [11]. The requirements for having a Chromium approved log are strict, therefore not every log is accepted. Other browsers, such as Firefox, do have this feature as well but their policies are not as strict as Chromium's [4], as each browser deploys their own criteria for this technology.

2.3 CAA Records

CAA records are a DNS service that allows domain name holders to add records which indicate the CAs which are allowed to issue certificates for the associated domain [10][15]. This standard features both standard issue and wildcard domain types for fine grained specification of authority for certificate issuance. Its purpose is to, similarly to CT Logs, enable third party auditing, help avoid misissuance, and overall limiting attack surface by reducing the number of CAs that can issue certificates. In the following example record Digicert is authorized to issue certificates for the domain `example.com`, while no authority is allowed to issue to the wildcard URL `*.example.com`.

```
example.com CAA 0 issuewild ";"
example.com CAA 0 issue "digicert.com"
example.com CAA 0 iodef "mailto:root@example.com"
```

Adoption is not as advanced as for CT Logs yet, but a 2017 ballot [14] made it mandatory for CAs to check CAA records before issuing a certificate. This does not mean that every domain has CAA records associated with it. In fact the majority does not and a 2018 paper by Scheitle *et al* [21] found that only six of the largest DNS operators allowed for their customers to configure these records.

3. METHODOLOGY

In this section we will explain the the methodology for this paper. This includes how the data sets are prepared, as well as the steps required to combine and analyse them to answer the research questions. To achieve this we use PySpark [2] running on a Hadoop [1] cluster to process the large amount of data in an efficient manner. The complete process is documented and executed via a Jupyter [8] notebook.

3.1 Data Sets

To answer the main research question we combine two data sets. The first one being a daily recording of CAA records which contains 908,336 unique domains between 2017 and 2020. The second data set is the union of multiple CT logs. Since there exists a large number of logs to choose from [6] and the amount of data they contain is quite large not all of them can be included. The Chromium CT logs policy [11] provides a list of logs which are trusted by Chromium. If a certificate is not present within one of these logs it will not be accepted by the browser. Therefore, the CT logs with the largest number of certificates, Google-Pilot and Google-Rocketeer, to cover a large amount of certificates.

3.2 Data Preparation & Cleaning

First both data sets have to be prepared accordingly. For the CAA records only domain name, date of recording and the associated CA value are used as input. We take the following steps to prepare the data for further analysis:

1. Group by domain name and date
2. Aggregate individual CAs to sets
3. Group by domain name and set of CAs
4. Aggregate groupings to minimum date

The resulting rows of the table contain the domain name, a set of authorized CAs and the first date of recording of this type of record. This process does ignore CAA record updates which revert to an old state, but since less than one percent of domain names' CAA records are updated more than twice this rarely occurs and can thus be neglected.

The CT data prep is simpler as it is given in the form of logs instead of daily recordings of a database. There are no duplicates to remove except for splitting the data into normal certificates and wildcard certificates. Relevant columns are the domain name, validity range of the certificate, a unique way of identifying the certificate in the form of a serial number and the issuing organisation.

3.3 Combining the Data Sets

To answer research question 1 the two data sets are to be combined. We achieve this by executing the following steps:

1. Join on domain name
2. Keep certificates where CAA record is later than estimated issuance

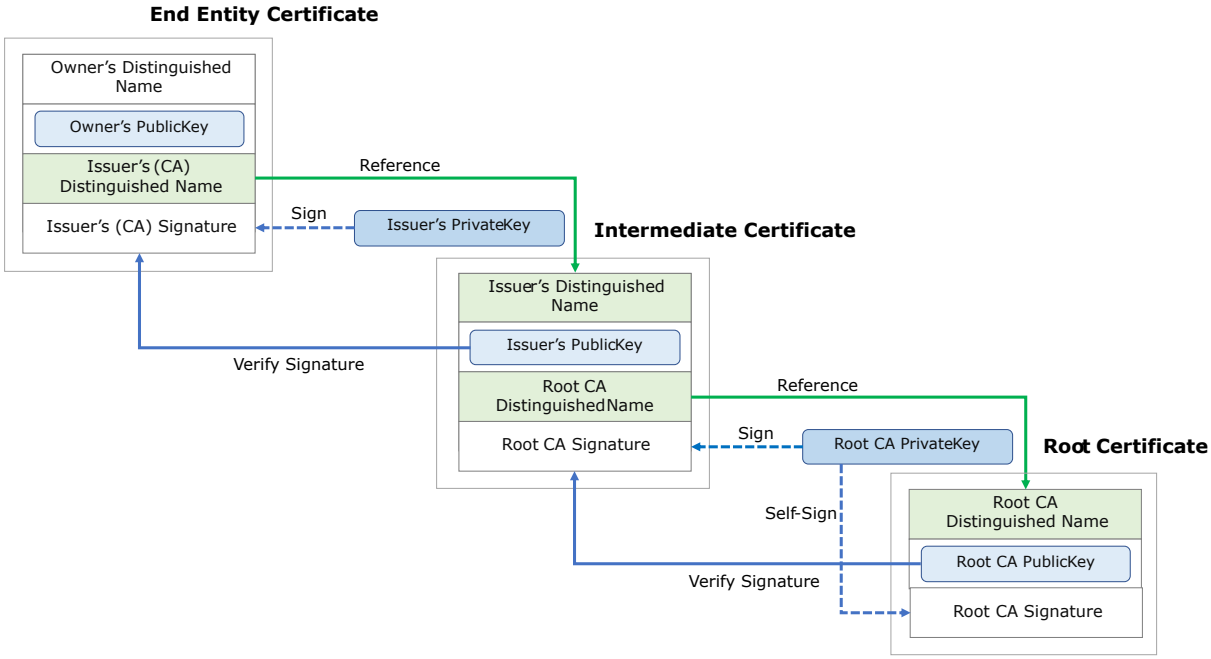

Figure 1: A model of the certificate chain structure [22]

3. Check if certificate issuer is authorized by CAA records

By executing each of these step individually we can then use the intermediate results to gain more insight on the distributions of these anomalies.

For this analysis we consider an anomaly as a CAA record, being updated for a domain, where the issuer of a certificate for this domain is not included in the updated record. For instance the domain `example.com` has `letsencrypt.org` and `digicert.com` in its new CAA record of type *issue*. However, a certificate for this domain is already issued by Sectigo. Since a url associated with Sectigo is not present in the CAA record this combination of certificate and CAA record update is recorded as an anomaly. The certificate does not have to be in use, but it would still be valid as per its expiry date. This is not only limited to leaf certificates. The RFC [15] also allows a certificate if it was signed by an authorized party higher up the certificate chain. Therefore, we will not count these case as anomalies.

3.4 Analysis of Anomalies

In order to analyse the anomalies correctly we quantify them by the following criteria:

- Top Level Domain (TLD)
- CAs authorized in CAA records
- Issuing CAs
- Number of CAs authorized

We compare the resulting numbers to all CAA updates to find irregularities. This way certain CAs or TLDs, which are more prone to anomalies, can be identified.

3.5 Certificate Investigation

By filtering out the certificates which have expired by the current date we get all theoretically still valid certificates. Utilizing the OpenSSL python library [7] we do a scan to retrieve the certificate currently in use through the standardized HTTPS port 443. We then extract the serial number, convert it to hexadecimal and compare it to the one mentioned in the identified anomalies. If they match the certificate is still in use, otherwise it has been updated. In case the number of certificates is too large to achieve this in a feasible amount of time we investigate this for a subset only, which we select on the basis of longest remaining lifespan for a certificate. For the purpose of simplification this is reduced to a true or false result. Only if the certificate is still present the scan will return true. In all other cases we do not consider the certificate to be in use anymore. For more advanced analysis it could be checked whether the website is still available, CA has changed etc., which we leave for future work.

3.6 Limitations

The main limitation with identifying these anomalies is the fact that the exact date of issuance for a certificate cannot be determined. Instead the *not_before* date is used to approximate this [21]. Another issue is that the domain holder can modify CAA records only for a short period of time when the certificate is issued. The CT logs themselves are also not as reliable. Since only two logs are used some anomalies may not be found as even a larger number of logs does not ensure coverage [19].

Another issue to be solved is the encoding of CAs between CAA records and CT logs. While the CAA records contain a URL the CT logs will not always map directly to it. Even websites that provide a list of valid CAA identifiers [9] does not provide all associated CA owner names which are actually used in certificates. This requires a manual mapping between the two, so as a compromise only CAA URLs with more than a hundred occurrences are mapped to their respective organization. Additional obvious cases in the spotted anomalies are added to this mapping if found. Due to this false positives can occur and depending on the size of the results infeasible to re-

solve. If the resulting set of anomalies is small enough, a manual inspection of a sample set of them can be used to update the mappings.

4. RESULTS

4.1 Research Question 1

How often does a new CAA policy affect a preexisting certificate?

As a result of the query complexity, constraints have to be put on the input data. Due to the preparation of CAA records their size is limited compared o the CT logs. Therefore, a large range of recordings is included, dating between 2017 and 2020. The largest portion of data stems from the CT logs. To limit this and filter out irrelevant certificates only ones issued after 2016 are included. Figure 2 gives an overview of the number of certificates in each log per year. There may be duplicates between the logs so the total number of certificates covered is lower than the sum between the two logs. However this can be neglected, as the fraction of certificates affected by CAA records is rather small, as seen in figure 3. In the last four years less than one percent of all certificates had an associated CAA record. This number is not surprisingly low as a 2018 empirical survey by Ruohonen [20] showed that only 1.6% of Alexa's 1M list have specified CAA records for their domains. This is further reduced by the fact that not all CAA records are covered by the certificates present in the two CT logs.

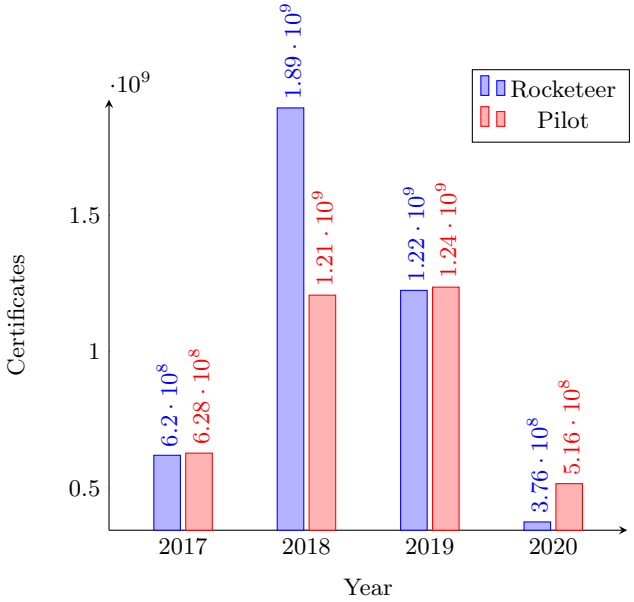

Figure 2: Number of non-wildcard certificates per year in the Rocketeer and Pilot CT logs

Figure 4 shows the number of times a CAA update has affected a certificate. The blue bar indicate the total number of relevant CAA updates, while red and grey are the updates out of the total count which fall under their respective categories. The red updates standing for a post issuance update, but during the lifespan of a certificate, while the grey ones are the part of these retroactive updates which were identified as anomalies as described in Section 3.3. A small quantity of the total occurrences has been retroactive while an even smaller percentage was identified as an anomaly. Table 1 gives the exact numbers for this graph. Between 2017 and 2020 the percentages of anomalies are 3.07%, 2.74%, 2.48% and 6.27% respectively and for retroactive CAA updates 13.54%, 27.24%, 40.05% and 78.57%. Due to 2020 being the last measured year its data points deviate a lot from the other years. Since no certificates from 2021 can be recorded there is a lack of proactive CAA updates in 2020. This results in a larger proportion of retroactive certificates and therefore also anomalies.

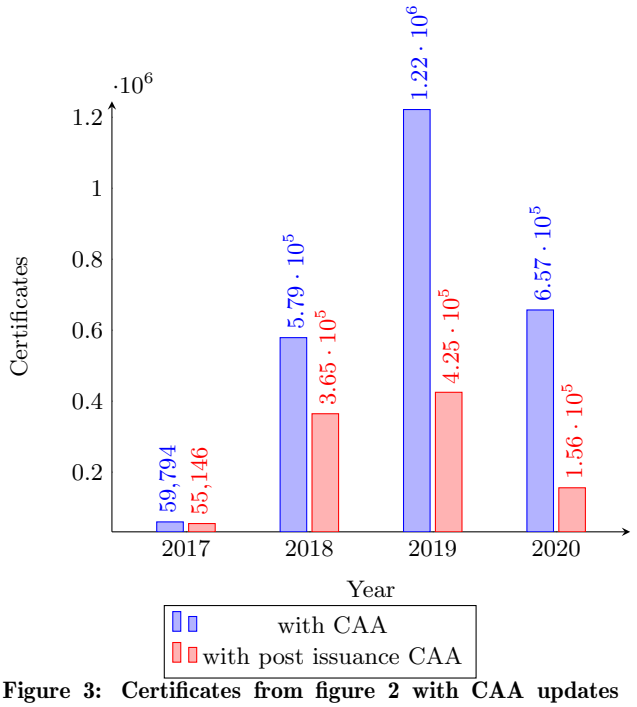

Figure 3: Certificates from figure 2 with CAA updates and how many of them have received CAA updates post issuance

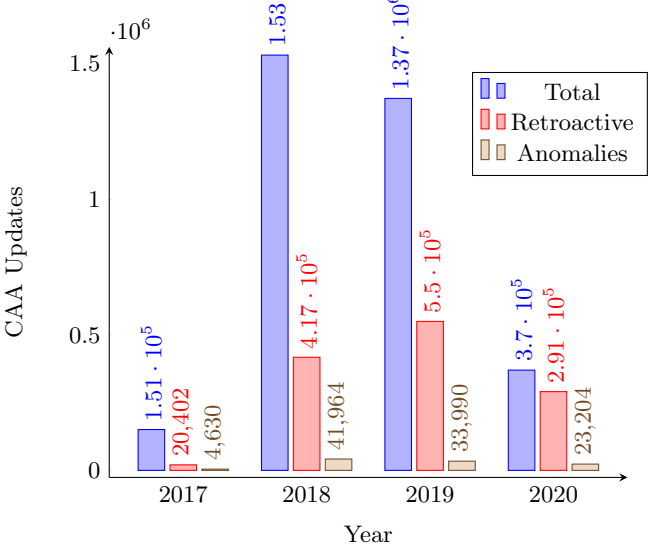

Figure 4: Number of times a certificate has been affected by a CAA update grouped by the type of update

Table 1: The exact numbers as displayed in figure 4

	Total	Retroactive	Anomalies
2020	369,801	290,537	23,204
2019	1,372,215	549,565	33,990
2018	1,531,868	417,276	41,964
2017	150,646	20,402	4,630

Out of these CAA updates there are two different kinds, ones where the issuer of the certificate is listed in the CAA records and ones where one of the issuer is higher up on the certificate chain is noted. The distribution of these can be seen in figure 5. The majority of certificates with about 80% are directly issued while for the other 20% a root or intermediate certificate's issuer is present in the CAA records.

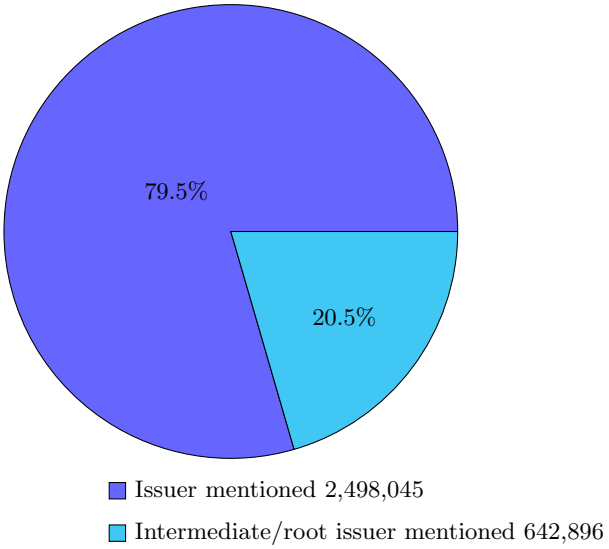

Figure 5: Distribution of CAA updates where the certificate issuer is authorized by CAA records versus the case of an intermediate or root certificate's issuer being mentioned instead

Due to the unreliable data set of 2020, to get a good estimation on how often a CAA update affects a preexisting certificate and whether it can be considered an anomaly we only use the years 2017, 2018 and 2019. 32.32% of CAA updates affect a certificate retroactively while 2.67% of CAA updates are retroactive and theoretically conflict with an existing certificate.

4.2 Research Question 2

Do the number of these occurrences differ between top level domains and CAs?

We analysed the anomalies according to the criteria from Section 3.4. The results of this can be found in tables 2, 3, 4 and 5. For all of these tables the total column documents the number of occurrences among anomalies. The ratio column shows how many CAA updates of this type are considered anomalies. Ideally all of these ratios should be equal. Smaller or larger values indicate a pattern, or an anomaly amongst anomalies.

Top level domain types, as seen in Table 2, are spread evenly, however `.pl`, `.in` and `.io` stand out with larger than 5% ratio. For CAs authorized by CAA records, see Table 3, amongst the most common CAs Sectigo stands out with over 15%. The patterns in issuing CAs from Table 4 are more significant. Here GoDaddy, GeoTrust, GlobalSign and Amazon appear unusually often. Table 5 shows the most common number of CAs authorized in the relevant CAA updates and is not as conclusive. The only unusual value would be for the cases where ten CAs are authorized. However this only takes place 23 times which is negligible. The spike in ratio with a single authorized CA could indicate that a new certificate by a different CA has been issued for the domain, which is most likely the cause of the majority of anomalies.

Table 2: Ten most common top level domains among anomalies and their ratio to all CAA updates

TLD	Total	Ratio in %
.com	46,268	3.14
.pl	13,703	5.73
.org	3,675	3.13
.net	3,332	2.24
.ru	2,933	2.41
.in	2,865	5.51
.de	2,367	2.10
.nl	1,785	4.27
.io	1,687	5.51
.br	1,472	1.55

Some of these findings may be attributable to the incomplete mapping as mentioned in Section 3.6 Limitations. However a manual inspection of anomalies for certificates issued by GoDaddy and GeoTrust confirm that this is not the case. Another conclusion we can draw from combining the tables is about Let's Encrypt. Certificates issued by Let's Encrypt are rarely identified as anomalies while they are listed the most in CAA records compared to other CAs. One explanation for this could be that domain holders authorize Let's Encrypt by default.

Table 3: Top CA URLs specified in CAA records among anomalies and their ratio to all CAA updates

CA	Total	Ratio in %
letsencrypt.org	74,252	2.76
digicert.com	35,352	1.82
comodoca.com	24,794	1.27
globalsign.com	8,930	1.19
sectigo.com	5,702	15.10
certum.pl	5,152	2.27
amazon.com	3,725	5.29
godaddy.com	2,211	4.73
amazonaws.com	2,032	7.19
amazontrust.com	1,883	7.96

To answer research question 2 we find that the `.pl`, `.in` and `.io` top level domains are more likely to be identified in an anomaly. These domain names follow the ISO 3166-1 alpha-2 standard [5]. From this we find that all three of these are country TLDs, `.pl` belonging to Poland, `.in` to India and `.io` to the British Indian Ocean Territory. However, the `.io` domain is usually not associated with its country due to it commonly being used for domains in the tech industry and the British Indian Ocean Territory having no permanent residents. As to why these domains stand out is not very clear. In the case of India and Poland it is possible that these countries utilize a different set of local CAs instead of the big players which are not covered by the methodology. For the case of `.io` domains one possibility is the rapid nature of the tech industry. Domains can be taken over and therefore the CA can also change,

the hosting service could be switched etc. The exact reason for these data points is unclear, we can only make an educated guess as an in-depth investigation would be out of scope for this paper.

Table 4: Top CAs among anomalies and their ratio compared to all CAA updates

CA	Total	Ratio in %
COMODO CA Limited	32,095	4.32
GlobalSign nv-sa	12,103	14.86
GoDaddy.com, Inc.	10,365	43.89
nazwa.pl sp. z o.o.	9,855	5.64
DigiCert Inc	9,043	9.83
Let's Encrypt	6,319	0.45
CloudFlare, Inc.	4,917	1.23
Amazon	3,929	16.03
Sectigo Limited	1,592	5.03
GeoTrust Inc.	1,381	41.63

In a similar sense, GoDaddy and GeoTrust as well as to a lesser extent GlobalSign and Amazon are identified as issuing CAs which appear more frequently amongst anomalies than all CAA updates. Let's Encrypt and CloudFlare are the opposite. For these CAs it is relatively uncommon to have an anomalous CAA update associated with their certificates. For the case of GeoTrust a dispute with Google in 2017 [12] rendered the GeoTrust root certificate untrusted. It is likely that this caused the majority of certificates based on this root to be switched out, possibly to a different CA, in turn resulting in a larger number of anomalies. It is unclear however, as to why the other CAs stand out in particular.

Table 5: Most common number of authorized CAs in anomalies and how this compares to all CAA updates

Number of authorized CAs	Total	Ratio in %
1	65,917	4.91
4	14,548	1.58
2	13,531	2.66
3	7,555	1.29
5	989	2.24
6	626	5.92
7	298	4.07
8	139	3.45
9	59	3.98
10	23	13.14

4.3 Research Question 3

What happens to affected certificates afterwards?

After checking which certificates are valid we find 9,216 anomalies which fulfill this criterion. Executing a TLS scan on the domains reveals that the majority of certificates associated with anomalies are no longer in use. Only in 9.74% of anomalous cases the associated certificate is still in use. Figure 6 shows this distribution and the exact numbers. It is important to recall for this section that an updated CAA record does not directly invalidate certificates, as they are only to relevant at time of issuance. It can however hint towards a certificate update, domain transfer or similar events.

In Figure 7 the number of days since an anomaly has occurred is plotted for the 818 cases in which the certificate is still in use. On average anomalies are 324 days old for these cases, with half of data points being located between 203 and 468 days. It can be seen that the box is skewed

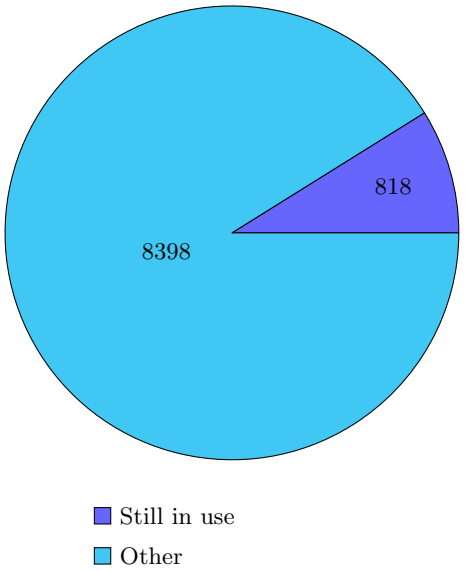

Figure 6: Number of anomalies for which the associated certificates are still in use and other cases

towards a lower number of days. This indicates that the longer it has been since an anomaly occurred it is less likely that the certificate will still be utilized.

It is not possible for us to create a similar boxplot for other cases as we do not have records of the exact dates at which certificates were no longer used or a website was taken down etc. Therefore, we cannot make a reasonable comparison to the average time it takes to remove a certificate. One would assume that this is done at the same time as updating the CAA record since it involves a conscious manual modification of permissions. However, as we already saw, this is not always the case, though this number is very small compared to the total number of CAA updates.

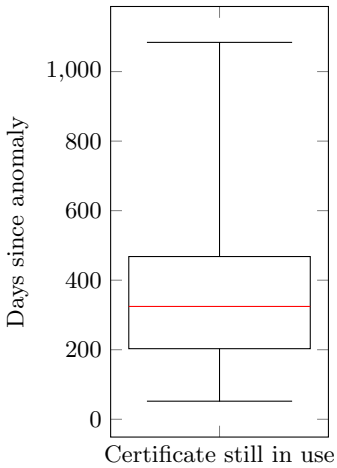

Figure 7: Number of days since an anomaly for cases where the certificate is still in use

5. RELATED WORK

We do relative work across two dimensions. To the best of our knowledge ours is the first study to investigate the combination of CT logs and CAA records and their retroactive effect on each other. Therefore related work on this research is mostly separated into the CT and CAA side. In between these two fields the work done on review-

ing CT logs is far more extensive than on CAA records as the technology is slightly older and adoption is more advanced.

5.1 CT

Google is currently the main driving force of CT. They provide documentation and insights on their project website, giving an overview of the integrity, functionality as well as a comparison to other technologies in the same category [3].

Other sources focus on the reliability of these log operators. A conference paper by Li *et al.* [19] analyses this exact concept. By looking at 88 logs and the service provided by third parties that monitor these logs they find that even with multiple monitors it is not guaranteed to discover the full set of certificates for a domain. It also highlights the scope of such an operation. The amount of data recorded exceeds 28 GB daily in 2019, which is a number that only continues to grow. Due to the combination of these factors one must consider the unreliability of CT logs as an additional factor in this paper.

A 2018 paper about tracking certificate misissuance by Kumar *et al.* [16] is about the correctness of certificates. It shows that the percentage of erroneous certificates is shrinking over time. However, large CAs contribute to this by having a very low number of incorrect certificates, pushing the overall percentages down. As a result of this, incorrect certificates correlate to other mistakes in the same field, which could also imply patterns for CAA records and the CAs associated with them.

5.2 CAA

A 2019 empirical study by Ruohonen [20] analyzed the adoption of CAA with a similar data set to the one that is used in this paper. The results of this show a variety of facts about CAA records, for example that the majority only authorizes a single CA to issue certificates for a domain while disallowing wildcard issuance. We also used a similar set of criteria to analyse the anomalies as specified in Section 3.4 such as top level domain distribution. Their exists a difference between the results, which may be attributable to different data sources or a shift in the use of the CAA technology.

In a 2018 article about analysing CAA records, Scheitle *et al.* [21] a multifaceted analysis of the adoption of CAA records is conducted. In one of these sections the role of a third party auditor is assumed which is very similar to the process of this paper. The authors use the CAA and CT data to identify certificates that have been misissued and why the misissuance occurred. They also reveal a multitude of limitations associated with this approach and how to mitigate them, for example, the lack of a concrete issuance date for certificates, which is solved by approximating it with the *not valid before* property. These issues are listed in Section 3.6. The analysis we did, while using a different approach and bringing distinct results, follows a similar process and uses the same data sets, therefore also being fallible to the same issues.

6. FUTURE WORK

Future work on this topic can aim to investigate the cause of these patterns. A reason for this could be that certificates issued by these CAs are more prone to being replaced. The number of anomalies could also be refined by discarding affected certificates which are not in use anymore at the time of a CAA update. This would require more complex processing of the large CT log data set and would also bring its own limitations with it. Another improvement was already mentioned in Section 3.5. The investigation of these anomalies can be improved further by finding the edge cases that lead to the continued deployment or lack thereof of a certificate. Examples of this are a change in CA or the website being taken down. Another way of building on this paper is a general scale up such as adding more CT logs to cover a larger number of certificates. Alternatively an extra data dimension can be added in the form of certificate revocation information from technologies such as OCSP [13]. A simpler way to achieve a similar effect is performing TLS scans every day after a CAA record update has been detected until a certificate is no longer in use. With this a comparison data set for Figure 7 can be constructed.

7. CONCLUSION

In this paper we investigated the ways in which CAA records can affect existing certificates after they have been issued. For this we defined an anomaly as the case of a CAA record being updated with the result of this being a conflict between the issuer of an existing certificate and the new set of authorized CAs. This is not saying that the old certificates are invalidated, as the CAA records are only valid at time of issuance. We found that these anomalies do happen on a non negligible basis and also recognized patterns related to the top level domains and CAs associated with the effected certificates.

As the main result we determined that CAA updates affect a certificate retroactively in 33.3% of cases while 2.7% result in a previously described anomaly. These numbers represent CAA updates only and not all certificates. In fact only 1.6% of the Alexa's 1M list of domains was covered by the provided CAA dataset. Furthermore we found patterns within these anomalies, mainly the three top level domains `.pl`, `.in` and `.io` appear more frequently than other TLDs. In a similar way we found GoDaddy and GeoTrust certificates representing a much larger number of anomalies than expected from the set of all CAA updates. To a lesser extent the same applies to GlobalSign and Amazon. On the other side of the scale fall Let's Encrypt and CloudFlare, being underrepresented amongst anomalies. After investigating these anomalies we found that the majority of certificates are no longer in use. Only 9.74% of anomalies that had theoretically valid certificates had them still deployed. Even so, with a mean time of 324 days since an anomaly it appears that these certificates are mostly ignored, forgotten or other circumstances. Based on these numbers we can assume that the majority of CAA updates also imply the obsoletion of an existing certificate.

All things considered we find that anomalies occur rarely and in an even smaller number of events a potentially obsoleted certificate is still in use. But, as CAA records are only relevant for CAs at issuance time no real security implications arise from this. Because of this no hard conclusions can be drawn from an identified anomaly. However, as we demonstrated in this paper, it is still possible to use CAA records as an auditing tool similar to CT logs, even though this is not its intended purpose. We are still able to reduce the number of certificates to investigate with relatively simple analysis based on the extra data dimension. Since CAA records have little to now downsides associated with them while providing additional security and auditing capabilities, we encourage further adoption of this technology.

8. REFERENCES

[1] Apache Hadoop. `https://hadoop.apache.org/`.

[2] Apache Spark™ - Unified Analytics Engine for Big Data. https://spark.apache.org/.

[3] Certificate Transparency. https://www.certificate-transparency.org/home.

[4] Expect-CT - HTTP | MDN. https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Expect-CT.

[5] ISO - ISO 3166 — Country Codes. https://www.iso.org/iso-3166-country-codes.html.

[6] Merkle Town. https://ct.cloudflare.com/logs.

[7] OpenSSL Cryptography and SSL/TLS Toolkit. https://www.openssl.org/.

[8] Project Jupyter. https://www.jupyter.org.

[9] Recognized CAA domains. https://ccadb-public.secure.force.com/ccadb/AllCAAIdentifiersReport.

[10] What's a CAA record? - DNSimple Help. https://support.dnsimple.com/articles/caa-record/.

[11] chromium/ct-policy. https://github.com/chromium/ct-policy, 2020.

[12] R. Chirgwin. Google to kill Symantec certs in Chrome 66, due in early 2018. https://www.theregister.com/2017/09/12/chrome_66_to_reject_symantec_certs/.

[13] S. Galperin, S. Santesson, M. Myers, A. Malpani, and C. Adams. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP.

[14] K. Hall. [cabfpub] Results on Ballot 187 - Make CAA Checking Mandatory. https://archive.cabforum.org/pipermail/public/2017-March/009988.html, Mar. 2017.

[15] J. Hoffman-Andrews, P. Hallam-Baker, and R. Stradling. IETF RFC8659 DNS Certification Authority Authorization (CAA) Resource Record. https://tools.ietf.org/html/rfc8659.

[16] 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 *2018 IEEE Symposium on Security and Privacy (SP)*, pages 785–798, May 2018. ISSN: 2375-1207.

[17] A. Langley, E. Kasper, and B. Laurie. IETF RFC6962 Certificate Transparency. https://tools.ietf.org/html/rfc6962.

[18] N. Leavitt. Internet Security under Attack: The Undermining of Digital Certificates. *Computer*, 44(12):17–20, Dec. 2011. Conference Name: Computer.

[19] B. Li, J. Lin, F. Li, Q. Wang, Q. Li, J. Jing, and C. Wang. Certificate Transparency in the Wild: Exploring the Reliability of Monitors. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*, CCS '19, pages 2505–2520, New York, NY, USA, Nov. 2019. Association for Computing Machinery.

[20] J. Ruohonen. An Empirical Survey on the Early Adoption of DNS Certification Authority Authorization. *Journal of Cyber Security Technology*, 3(4):205–218, Oct. 2019. arXiv: 1804.07604.

[21] Q. Scheitle, T. Chung, J. Hiller, O. Gasser, J. Naab, R. van Rijswijk-Deij, O. Hohlfeld, R. Holz, D. Choffnes, A. Mislove, and G. Carle. A First Look at Certification Authority Authorization (CAA). *ACM SIGCOMM Computer Communication Review*, 48(2):10–23, May 2018.

[22] Yuhkih. Certificate trust chain. https://commons.wikimedia.org/wiki/File:Chain_Of_Trust.svg, Sept. 2020.