Revocation Statuses on the Internet

Nikita Korzhitskii
Linköping University, Sweden

Niklas Carlsson
Linköping University, Sweden

ABSTRACT

The modern Internet is highly dependent on the trust communicated via X.509 certificates. However, in some cases certificates become untrusted and it is necessary to revoke them. In practice, the problem of secure certificate revocation has not yet been solved, and today no revocation procedure (similar to Certificate Transparency w.r.t. certificate issuance) has been adopted to provide transparent and immutable history of all revocations. Instead, the status of most certificates can only be checked with Online Certificate Status Protocol (OCSP) and/or Certificate Revocation Lists (CRLs). In this paper, we present the first longitudinal characterization of the revocation statuses delivered by CRLs and OCSP servers from the time of certificate expiration to status disappearance. The analysis captures the status history of over 1 million revoked certificates, including 773K certificates mass-revoked by Let's Encrypt. Our characterization provides a new perspective on the Internet's revocation rates, quantifies how short-lived the revocation statuses are, highlights differences in revocation practices within and between different CAs, and captures biases and oddities in the handling of revoked certificates. Combined, the findings motivate the development and adoption of a revocation transparency standard.

1 INTRODUCTION

The modern Internet uses the Web Public-Key Infrastructure (WebPKI) as a foundation to establish trust between clients and servers. In WebPKI, Certificate Authorities (CAs) issue signed X.509 certificates that verify the mapping between public keys and public distinguished names, such as domain names.

In certain cases (e.g., a private key compromise, owner's request, or misissuance by a CA), certificates must be revoked; i.e., rendered invalid. To protect clients and servers from the use of revoked certificates, WebPKI supports several revocation protocols. Currently, revocation statuses of most certificates can be obtained via Online Certificate Status Protocol (OCSP) servers [28], but some CAs continue to support the traditional Certificate Revocation Lists (CRLs) [6] as a complementary option. However, these pull-based protocols raise many security, privacy, and performance issues. Therefore, many browser vendors do not utilize the protocols [23], but instead, they push a proprietary set of revocations to the users [2, 11]. Yet, these push-based revocation mechanisms have their own limitations, which leave secure certificate revocation an open problem [4].

Furthermore, as of today, there does not exist any standardized mechanism in place (similar to Certificate Transparency (CT) [14, 20, 29] w.r.t. certificate issuance) to provide an immutable history of all revocations and corresponding revocation reasons. Consequently, there is no ability to easily study and detect revocation-related misbehavior by CAs (e.g., advertisement of wrong, or contradictory revocation statuses). While many novel WebPKI extensions, revocation protocols, architectures, and transparency schemes have been proposed to address this issue, none have been adopted so far [4]. Instead, we observe that the information about revocations is sparse and most revocation statuses disappear soon after certificate expiration.

In this paper, we make *a case for revocation transparency* by presenting a novel characterization study of the revocation rates on the Internet, the post-expiry life of revocation statuses, and the status-handling practices across CAs. First, we present a measurement methodology that allows us (i) to obtain nearly all revocations performed for the set of certificates expiring during a time window, and (ii) to track the certificate status (using both OCSP and CRL) of such sample sets over 100-day periods, starting at their respective expiration dates[1].

Second, we track all certificates from the Censys dataset [10] that expired between Mar. 2, 2020, and Apr. 1, 2020, and that were valid with respect to Apple's, Microsoft's, or Mozilla's root stores. This time period (see Figure 1) is particularly interesting since the measurement was done prior to and during the mass-revocation event in which Let's Encrypt (LE), the largest CA, initially announced to revoke over 3 million certificates [22] due to a CAA-rechecking bug, but in the end, they revoked only 1.7M certificates [21].

Third, and most importantly, we characterize the revocation-status-handling practices across CAs, including status lifetimes beyond the expiration date and handling differences across CAs and certificate types. We identify classes of behaviors, compare and contrast practices of different CAs, find revocation biases among different sets of certificates, and look closer at some odd CA behaviors (e.g., certificates that switch back to a "Good" status after being advertised as "Revoked"). Across our analysis, we observed highly heterogeneous behaviors among CAs and quick disappearance of revocation statuses. This highlights the lack of a global revocation transparency standard that would otherwise help

---

[1]Currently, CAs must maintain revocation statuses only until certificate expiration [1].
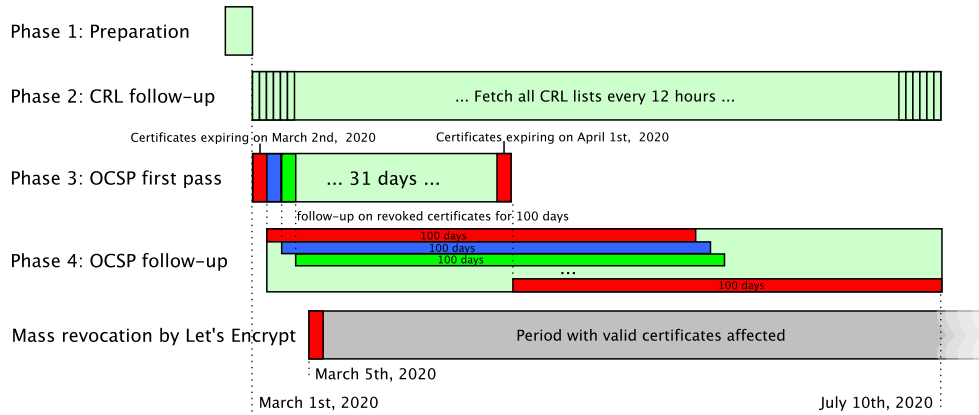
Figure 1: Timeline of the measurement.

to identify and improve odd revocation behaviors, similarly to CT, with its effect on the issuance process. Finally, we share our dataset [19].

Outline: After a brief overview of revocation protocols (Section 2), we present our methodology (Section 3) and characterization results (Section 4). Finally, related work (Section 5) and conclusions (Section 6) are presented.

2 REVOCATION PROTOCOLS

The two primary revocation protocols that CAs typically use are the following:

- **Online Certificate Status Protocol (OCSP):** Using OCSP, a client can request the status of a certificate by providing a serial number and the hashes of the issuer's name and key. The CA-Browser forum requires signed responses to be valid for at least 8 hours, and at most 10 days [1]. OCSP can be used in different ways. For example, *OCSP stapling* allows statuses to be delivered by a web-server, and the *OCSP Must-staple* extension prevents a client from making OCSP requests on their own and enforces a hard-fail policy if the status was not delivered by the web-server. The Must-staple extension is not widely adopted yet [5]. Instead, most browsers typically accept a certificate if they are unable to obtain revocation information [23].
- **Certificate Revocation List (CRL):** CAs maintain signed lists with the serial numbers of revoked certificates, and optionally, corresponding invalidation dates and reason codes for the revocations. CRLs can also be augmented using several extensions (e.g., CRL number, Authority Key Identifier, etc.) [6]. CRLs are required to be reissued at least once every 7 days [1].

Due to the security, privacy, and performance issues with OCSP and CRL, many browser vendors have disabled the above pull-based revocation protocols; instead, they periodically push limited sets of revocations to the clients (e.g., via software updates) [2, 11]. However, this approach has some limitations; e.g., a delay introduced by scheduled updates, and a small coverage of all existing revocations.

WebPKI lacks revocation transparency, and no mechanism similar to CT has been adopted yet. In fact, CAs are not required to maintain revocation statuses for certificates beyond their expiration date [1], and as we show in this paper, most of the time, revocation statuses stop being advertised shortly after certificate expiration. The lack of a transparent and immutable history of revocations complicates keeping CAs accountable for their revocation mishandling.

3 MEASUREMENT METHODOLOGY

We conducted a four-phase measurement campaign (see Figure 1).

1. Preparation: In the first phase, we collect all X.509 certificates (with their parent certificates) found in CT logs [20] and active scans that expire within a period starting from Mar. 2, 2020, to Apr. 1, 2020, using Censys [10]. For the analysis, we only select certificates that are valid with respect to Apple's, Microsoft's, or Mozilla's root stores [18]. From these certificates, we extract all OCSP responder URLs (used in phases 3+4) and CRL URLs (used in phase 2). For every remaining certificate, we then schedule an OCSP first pass (phase 3) 22 hours before its expiration[2], and for every observed CRL, we schedule periodic CRL requests (phase 2).

2. CRL follow-up: During the second phase, we regularly (every 12 hours) fetch all CRL lists using the URLs extracted in the first phase.

---

[2]The interval of 22 hours (slightly less than 24 hours) was selected for performance reasons, after the initial evaluation of our measurement framework.

3. OCSP first pass: In the third phase, we perform an OCSP status lookup for each certificate 22 hours before it expires. If a certificate is found to be revoked during its first pass, it gets scheduled for follow-up checks every 12 hours (phase 4). In the case of an OCSP timeout or an error, the first pass is retried every minute until a revocation status is obtained or the certificate is expired.

4. OCSP follow-up: In the fourth phase, the revocation status of every revoked expired certificate is fetched every 12 hours for 100 days (since the first pass of each individual certificate). We separate OCSP responses into four types: "Good", "Revoked", "Unauthorized", and "Unknown". The first two types ("Good" and "Revoked") are cryptographically-signed responses that definitively specify the status of a certificate. The third type ("Unauthorized") is an unsigned plaintext response. The final category ("Unknown") contains signed "Unknown" statuses (that some CAs deliver) and other unsigned responses.

External effects on the sampling rate: Between May 12, 2020, and May 19, 2020, parallel processes running at our server have temporarily increased the average OCSP inter-request time from 12 hours up to 21.7 hours. Except for this short period, the average OCSP inter-request time was consistently 12 hours ± a few minutes, up until June 21, 2020. Between June 21, 2020, and the end of our measurement period on July 20, 2020, the average inter-request time was roughly 24 hours. Neither of the periods with increased OCSP inter-request times took place during the first month after the expiration date of any of the certificates; hence, the effects do not impact our conclusions.

4 CHARACTERIZATION RESULTS

4.1 High-level breakdown

In total, we collected OCSP status information for 49M certificates. Table 1 provides a breakdown based on whether a certificate was revoked or not, whether the certificate was issued by Let's Encrypt (76.3% of the certificates) or a different CA (23.7%), and whether a Let's Encrypt certificate was part of the above-mentioned mass-revocation event (1.57%). For us to consider a certificate mass-revoked it needed to be (i) on the list of 3M certificates that Let's Encrypt publicized for the event [22] and (ii) to be revoked at the time it expired. We also found that 297K certificates from the list, with expiration dates falling on our first pass period, have never been revoked.

Table 1: Summary of the studied certificates.

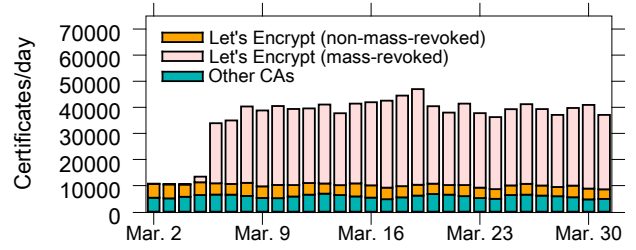| Certificates	Mass-revoked LE	Rest LE	Other CAs	All
Non-revoked	–	36,755,317	11,496,607	48,251,924
Revoked	773,128	129,552	174,712	1,077,390
Revocation rate	100%	0.35%	1.50%	2.18%

Figure 2: Revoked certificates per expiration date.

The timing of the mass-revocation event is particularly interesting since it provides a concrete example of the impact that such events can have on the revocation rate and the lifetime of revocation statuses. Finally, we note that the certificates affected by recent mass-revocation events have been disclosed through website postings of arbitrarily formatted datasets [8, 9, 22].

While the non-mass-revocation-rate of Let's Encrypt was much smaller than for the other CAs (0.35% vs 1.50%), the mass-revocation event increased Let's Encrypt's revocation rate for this period up to 2.40%. The effect is perhaps most noticeable when looking at the number of revoked certificates per day, based on their day of expiry, as shown in Figure 2. Here, starting from Mar. 5, 2020, we can see the impact of the certificates associated with the mass-revocation event. The other two classes of revocations remained relatively stable throughout the measurement period.

We found large variations in the revocation rates of different CAs. Figure 3 shows the number of revoked and non-revoked certificates, broken down per CA. Moreover, we mark the number of revoked certificates listed in the CRLs, in addition to OCSP servers (discussed in Section 4.4). Here, we show all CAs with at least 100 revoked certificates in our dataset, ranked from the one with the most revocations to the one with the least. We also include the "other" category that combines the results for all other CAs. While most CAs have much fewer revoked certificates than non-revoked certificates, there are notable exceptions. Five CAs even had more revoked than non-revoked certificates: Actalis (92.5%), nazwa.pl (66.4%), SwissSign (59.9%), Plex (73.7%), Digidentify (100%). Among the most popular CAs, GoDaddy stands out with 34.5% of its certificates being revoked before expiration.

4.2 Revocation status changes

The revocation statuses provided by OCSP servers often change from "Revoked" to some other status soon after certificate expiry. Figure 4 shows the time that the status remained "Revoked" after the revoked certificates had expired. Here, we filter out any temporary OCSP responses (e.g., unauthorized, unknown) and timeouts whenever we obtained at least one more "Revoked" response.
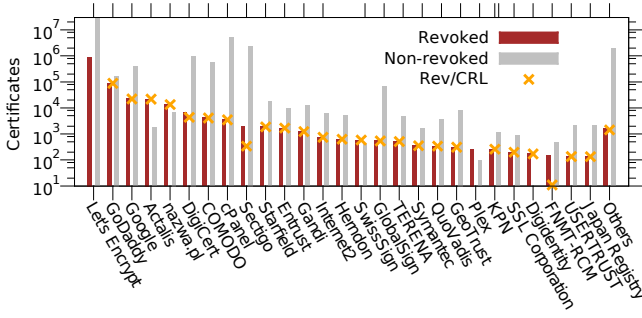
Figure 3: Per-CA breakdown of the number of revoked and non-revoked certificates in the dataset. Revoked certificates found in CRLs are shown with ×. CAs are ordered by the number of revoked certificates, in descending order from left to right. In the following figures, the order is preserved.

Quickly disappearing revocation statuses: Figure 4(a) shows the empirical Cumulative Distribution Functions (CDFs) for four classes of revoked certificates: 2 for Let's Encrypt certificates (mass-revoked and non-mass-revoked) and 2 for certificates by other CAs (with and without Extended Validation (EV)). All certificates by Let's Encrypt changed status within 3 days of expiration. Their mass-revoked certificates had longer status change times than the non-mass-revoked certificates. The CDFs for the other CAs are relatively flat from about two weeks to 100 days. (Note the logarithmic y-axis.) On an encouraging note, the certificate class with the most long-lived revocation statuses is Extended Validation (EV) certificates. This class of certificates should typically endure the most scrutiny.

Some CAs keep the state longer: Figure 4(b) shows the fraction of the certificates issued by different CAs that maintained the revoked status for at least 1 week or 30 days. While many CAs maintained "Revoked" state for very short time periods after certificate expiry (see Figures 4(a) and 4(b)), most of the CAs that did keep the "Revoked" state beyond a week also kept this state beyond 30 days.
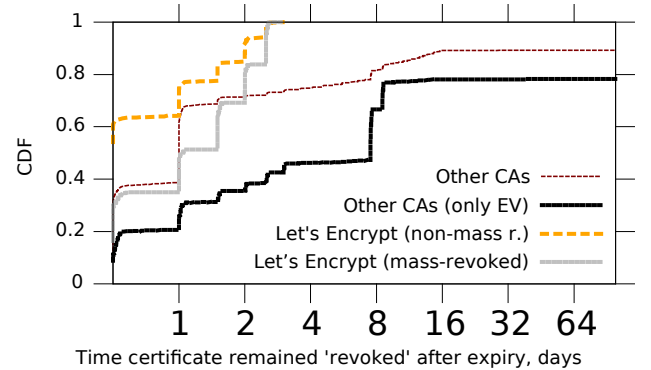
Status response overview: For the revoked certificates, we performed more than 207M OCSP status requests.

All certificates started as "Revoked" and most eventually changed to an unauthorized response (100% of Let's Encrypt certificates and 76.43% of other CAs' certificates). While we only had timeouts for 0.04% of the status requests, the differences between the number of affected certificates were substantial between CAs: only 0.07% of the Let's Encrypt certificates had at least one timeout, compared to 13.98% of the other CAs' certificates. These fractions are non-negligible,
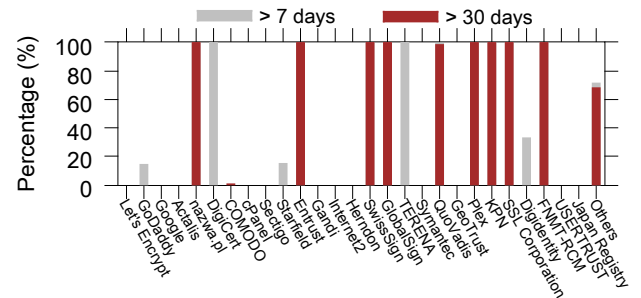
since most browsers soft-fail on an OCSP timeout and continue to establish a potentially-insecure connection. A concerning observation is that 589 certificates issued by 13 CAs (0.34% in the other CA category) switched from "Revoked" status to "Good" (66K responses in total).

Most frequent behaviors: Usually, public certification practice statements of CAs guarantee revocation status preservation for non-expired certificates, but do not specify the CAs' actions after that [13, 15, 32]. We next look at the most frequent CA behaviors. For this analysis, we filtered out temporary status changes whenever we observed the original state again. With this filtering, we observed the following dominating behaviors.

- Let's Encrypt almost always transition statuses from "Revoked" to "Unauthorized". This behavior was observed for 772K (99.86%) of the mass-revoked certificates and 129K (99.88%) of the other certificates revoked by Let's Encrypt. A possible explanation for this behavior is that they respond with code "Unauthorized" as soon as the status record has been removed [7]. Let's Encrypt's current certification practice statement only guarantees that "OCSP responses will be made available for all unexpired certificates" [15].

(a) CDFs

(b) Per-CA breakdown

Figure 4: Time that the revoked certificates remained revoked after expiration.

- Among the other CAs, we observed three dominating behaviors: 133K (76.28%) cases where the CA simply transitioned to "Unauthorized" (like Let's Encrypt), 21K (12.49%) cases where the status always changed to "Unknown", and 19K (10.68%) cases where the "Revoked" status remained for the duration of our measurement period.

Figure 5 breaks down the use of the dominating status change behaviors employed by the different CAs. In addition to the three behaviors mentioned above, we include the "other" behavior category. Most CAs have a dominating behavior that they employ for almost all of their certificates: 15 (out of 26) CAs almost always switch from "Revoked" to "Unauthorized", 9 (out of 26) CAs almost always keep the "Revoked" status for the full 100 day period, Actalis mainly switch certificates from "Revoked" status to "Unknown" (except for 91 cases, when the statuses were switched to "Good", following the intermediate "Unknown" status), Digidentify (who revoke all certificates) always start to timeout, and Japan Registry always switches statuses to "Good". As expected, the "other CA" category (not explicitly listed), contains a mix of behaviors. These results demonstrate the lack of a standard practice w.r.t. revocation statuses after certificate expiration. We have also observed some small differences in the weekly status-change patterns between CAs; however, compared to the differences in issuance timing, these differences are very small. See Appendix A.

Special cases with the "Good" status: 589 revoked certificates switched to status "Good". In almost all cases the servers kept the "Good" status until the end of the measurement period. In 349 of these cases, the status changed directly from "Revoked" to "Good" and in 91 cases an intermediate "Unknown" status was observed. All these cases provide strong motivation for transparent long-term recording of revocation information.

We note that Let's Encrypt and most of the other big CAs did not have any cases with the above strange behavior. Of the CAs with at least 100 revocations, only the following CAs had such cases: GoDaddy (117 cases), Actalis (91), Starfield (9), Entrust (5), and Japan Registry (135). Other CAs (not listed in our figures) with many cases include: "National Institute of Informatics" (91), "SECOM Trust Systems" (70), "ACCV" (54). (The rest of the non-listed CAs had five or fewer revoked certificates changing to status "Good".) Finally, a few certificates in this category stood out more than the others. For example, the list included three EV certificates: one by Entrust for "JPMorgan Chase and Co" ("Revoked" → "Good" → "Revoked"), one by GoDaddy for "Delmarva Broadcasting Company" ("Revoked" → "Unauthorized" → "Good"), and one by Actalis for "Pratiche.it" ("Revoked" → "Unknown" → "Good"). Otherwise, all the certificates in this class include
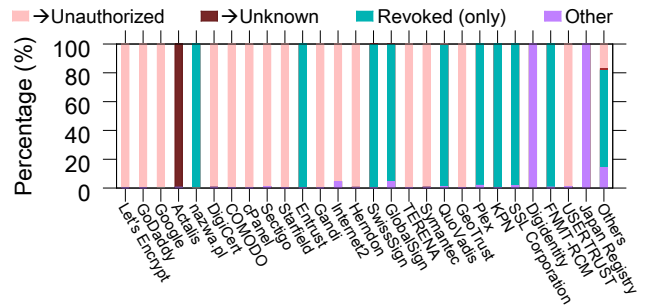

Figure 5: Dominating status change behavior of CAs.

RSA keys with the following key lengths: 1024 (9), 2048 (579), and 4096 (1). Furthermore, only 123 (out of 589) had Signed Certificate Timestamps (SCTs) embedded. We contacted all CAs with the above behavior. A summary of the responses is provided in Appendix B.

4.3 Biases in the revocation sets

Validity period: We have found that the revoked certificates typically have longer validity periods. Figure 6(a) shows CDFs of the validity periods for both revoked and non-revoked certificates for all CAs other than Let's Encrypt. (Since Let's Encrypt always use a 90-day validity period, we kept these certificates separately.) Here, we note a clear shift between the two curves.

Figures 6(b) and 6(c) provide a similar comparison of the (b) revoked and (c) non-revoked certificates on a per-CA basis. Here, we plot the fraction of certificates with validity periods longer than 89 days, 90 days, 1 year (365 days), and 2 years (720 days), respectively. These choices are based on the observation that many CAs use validity periods of either 90 days or 398 days (e.g., steps in the CDFs in Figure 6(a)). For almost all CAs, the fraction of certificates with long validity periods is larger among the revoked certificates (Figure 6(b)) than among the corresponding CA's non-revoked certificates (Figure 6(c)). This is in part an effect of CA/Browser Forum conventions [1] and decisions by individual browsers [3, 12, 24] forcing CAs to use shorter certificate validity periods. Another reason is that older certificates have had more time to become compromised. It could also be an indication that CAs apply increasingly stricter security policies (e.g., to comply with CT [20]).

Public key types: The modern WebPKI relies on EC (Elliptic Curve) [17] and RSA (Rivest–Shamir–Adleman) [26] public-key cryptography. Here, we compare the use of different key types and key lengths. While RSA 2048 is the dominating public key among both revoked (90.44%) and non-revoked (80.81%) certificates, there are significant differences in the revocation rates of certificates including different key types. For example, certificates with RSA 3072
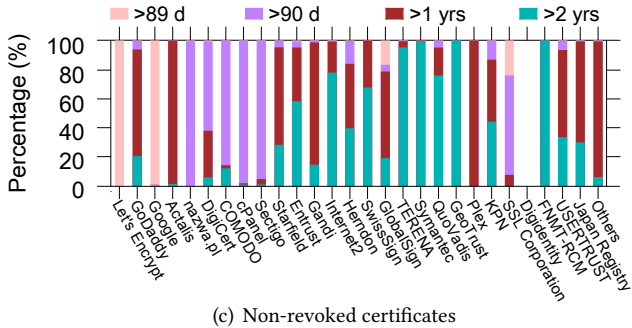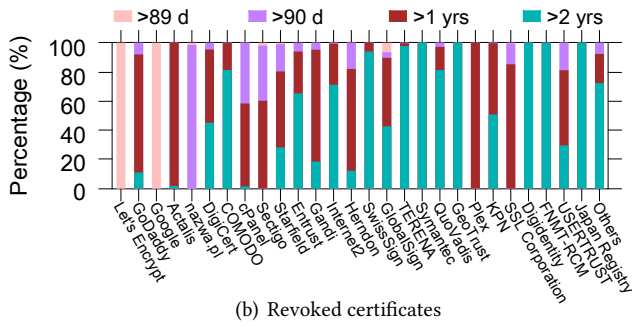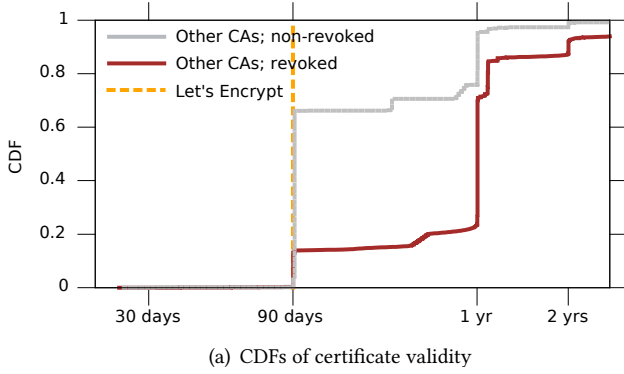
(a) CDFs of certificate validity

(b) Revoked certificates

(c) Non-revoked certificates

Figure 6: Validity periods for different categories of revoked and non-revoked certificates.

(4.55% revocation rate), EC 521 (80.49%) and RSA with key lengths other than the three most common lengths (6.67%) all have revocation rates well above average. In contrast, EC 256 (0.14%), EC 384 (0.62%) and RSA 4096 (1.48%) all have revocation rates below average. These differences are also present when looking at certificates of Let's Encrypt and other CAs separately.

SCT and EV usage: To measure the CT compliance we looked at the use of Signed Certificate Timestamps (SCTs). While all certificates issued by Let's Encrypt have embedded SCTs, other CAs do not always embed the timestamps. Furthermore, among the certificates issued by other CAs, the fraction of certificates that do not contain SCTs was much

greater among the revoked (10.04%) than non-revoked certificates (1.91%). In addition to having longer validity periods, some of the older non-expired certificates lack embedded SCTs. Owners and issuers of these certificates may be replacing them with certificates that better meet recent browser requirements [3, 25]. We have also observed significantly higher revocation rates among EV certificates. For example, 2K (10.77%) out of the 18K observed EV certificates were revoked. Furthermore, for CAs other than Let's Encrypt, 1.08% of the revoked certificates are EV certificates and 0.14% of the non-revoked certificates are EV certificates.

4.4 CRL-based analysis

For the 2K CRL URLs extracted from the certificates of interest, we collected 644K CRL snapshots. Combined, these snapshots included CRL entries for 170K (15.8%) of the revoked certificates found using OCSP. Let's Encrypt's decision not to implement CRL contributes to the small fraction. Here, we focus on the certificates with at least one CRL entry and one OCSP "Revoked" status.

Timing analysis: On average, revocation statuses disappear even faster from CRL lists than from OCSP responders. For example, only in 26.5% of the cases did we observe the revocation status in the CRLs after the expiration date of the certificates, and only for 2.9% did we observe the status being preserved longer than a week after expiration. This may be an attempt to reduce the size of the CRLs. However, since the majority of the revocations happen early in the lifetime of the certificates (e.g., the median normalized lifetime is 13.8%) there is still a significant time period over which certificates are included in the CRLs. This is illustrated in Figure 7(a), which shows the normalized timing of revocations and when the CRL entries are last observed in our dataset. Here, all values are normalized relative to the total intended validity period (i.e., "NotBefore" and "NotAfter" corresponds to the values 0 and 1, respectively). As implied by Little's law, the average size of a CRL (e.g., measured as entries per CRL) is equal to the average time that the entries remain in the CRL (e.g., measured in days) times the average rate that certificates are being added to the CRL (e.g., revocations per day), CRL sizes therefore easily become very large. Indeed, the average CRL size was 7K entries and the largest CRL contained 1.1M entries at its peak. Figure 7(b) shows CDFs and CCDFs for both individual measurements (all) and when using the observed peak size (max_t). We also observed some CRLs that did not appear to delete entries and roughly 0.94% of the certificates remained in the CRLs for the full duration of our measurement.

Revocation reasons: Figure 8 breaks down on a per-CA basis the percentage of certificates for which (i) we did not

find any CRL entry, (ii) we found CRL entries without a revocation reason, or (iii) we found a revocation reason for. The overall percentages for particular reasons (over all certificates with CRL entries) are provided in the figure's key. The four dominating CRL behaviors that we observed were: (i) some CAs did not use CRLs (Let's Encrypt, Plex) or only used it to a limited degree (e.g., Sectigo, FNMT-RCM), (ii) 17 CAs used CRLs for the majority of their revocations but did not provide any revocation reason, (iii) three CAs almost always used "Cessation Of Operation" as revocation reason (Go-Daddy, Google, Starfield), and (iv) three CAs almost always specified "Superseded" as the revocation reason.

Overall, most revoked certificates were not included in CRLs and 19.6% of CRL entries contained no revocation reason. Our results show that the practices of CAs are highly heterogeneous and revocation statuses are not persistent; thus, we argue that the Internet would benefit from a revocation transparency standard.

5 RELATED WORK

A number of studies have measured the revocation rates on the Internet. Liu et al. [23] performed several IPv4 HTTPS scans and found that a large fraction of served certificates was revoked (8%), while CRLSets [11] by Google was only covering 0.35% of all revocations. Chung et al. [5] evaluated the performance of OCSP responders by sending OCSP
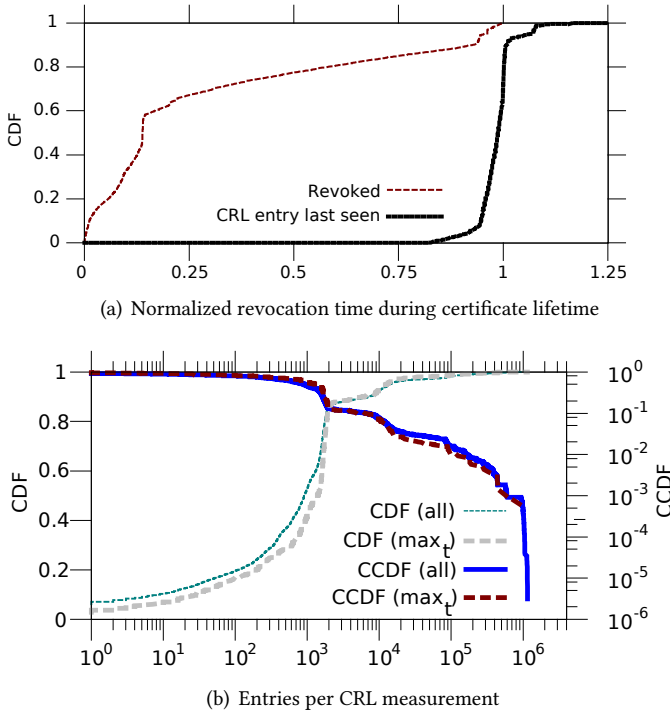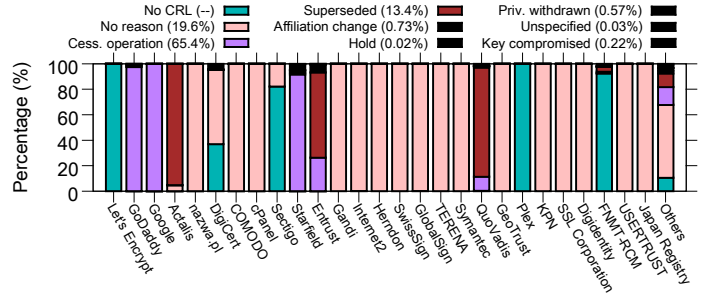

Figure 8: CRL reasons for revocation.

requests from geographically separated locations. They concluded that OCSP responders were not sufficiently reliable to support *OCSP Must-staple* extension. Zhu et al. [33] found OCSP latency to be "quite good", and showed that 94% of OCSP responses are served using CDNs. Moreover, only 0.3% of certificates were found to be revoked at that time (2015). Smith et al. [31] propose an efficient scheme to disseminate revocations. In the process, they measured revocation rates and found that in the absence of a mass-revocation event, the revocation rate on the Internet was 1.29%. This is similar to what we observed. The above works perform OCSP status checks before certificate expiration, while we check the certificates the day before their expiration and onward. Revocation effectiveness at the code-signing PKI was measured in [16], and a number of security problems related to revocations were identified. A recent survey and a comprehensive framework for comparison of implemented and proposed revocation/delegation schemes are provided in [4].

Other community efforts and data sources: The CA/Browser forum specifies some requirements that motivated our measurement design, including the requirement that "revocation entries on a CRL or OCSP Response MUST NOT be removed until after the Expiry Date of the revoked Certificate" [1]. We used the Censys search engine, backed by Internet-wide scanning [10], to obtain all certificates for our study. Some other online services also provide revocation statuses. For example, crt.sh [30] fetches known CRLs regularly, and performs OCSP requests on-demand. Until Aug. 2020, Internet Storm Center [27] was regularly fetching several CRLs; however, they did not monitor all CRLs present in our dataset and did not capture the mass-revocation by Let's Encrypt.

6 CONCLUSION

In this paper, we have presented the first characterization of the revocation status responses provided by OCSP and CRL responders from the time of certificate expiration and beyond. We described a measurement methodology, which allowed us to look at the revocation rates on the Internet from a new perspective; we quantified how short-lived the

(a) Normalized revocation time during certificate lifetime

(b) Entries per CRL measurement
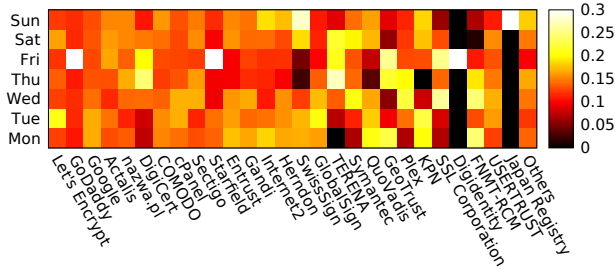
Figure 7: Distributions for measured CRLs.

revocation statuses are, and highlighted differences in status handling practices of different CAs. We found that most CAs remove revocation statuses very soon after certificate expiration. Some CAs do not provide CRL entries for all revoked certificates and/or remove entries from the CRLs before certificate expiration. The CA-dependent differences highlighted throughout the paper (e.g., revocation status lifetimes, usage of reason codes, and abnormal behavior of switching certificates from "Revoked" to "Good" status) capture a highly heterogeneous landscape that lacks a revocation transparency standard. Finally, we argue for the deployment of revocation transparency and demonstrate the global impact of the mass revocation event, which took place during our measurement campaign. We compared the characteristics of the mass-revoked certificates with the characteristics of other revoked and non-revoked certificates issued by Let's Encrypt and the rest of the CAs, and found a limited number of biases, e.g., the biggest differences in the revocation rates depend on the origin CA, key type, EV policy, and presence of embedded SCTs.
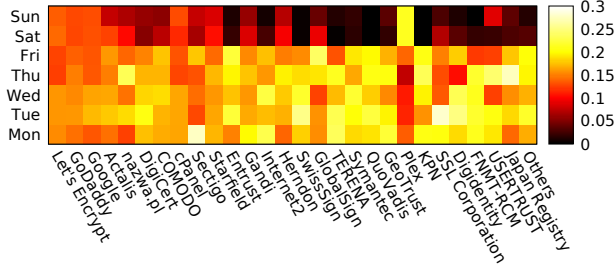
ACKNOWLEDGMENT

REFERENCES

[1] Baseline Requirements for the issuance and management of publicly-trusted certificates, v1.7.2 (2020), https://cabforum.org/baseline-requirements-documents/
[2] OneCRL (CA/Revocation Checking in Firefox) (2020), https://wiki.mozilla.org/CA:RevocationPlan#OneCRL
[3] Apple: About upcoming limits on trusted certificates (2020), https://support.apple.com/en-us/HT211025
[4] Chuat, L., Abdou, A., Sasse, R., Sprenger, C., Basin, D., Perrig, A.: SoK: Delegation and Revocation, the Missing links in the Web's Chain of Trust. In: Proc. IEEE EuroS&P (2020)
[5] Chung, T., Lok, J., Chandrasekaran, B., Choffnes, D., Levin, D., Maggs, B.M., Mislove, A., Rula, J., Sullivan, N., Wilson, C.: Is the Web ready for OCSP Must-Staple? In: Proc. IMC (2018)
[6] Cooper, D., Santesson, S., Farrell, S., Boeyen, S., Housley, R., Polk, W.: Internet X.509 Public Key Infrastructure certificate and Certificate Revocation List (CRL) profile. RFC Editor, RFC 5280 (May 2008)
[7] Deacon, A., Hurst, R.: The Lightweight Online Certificate Status Protocol (OCSP) Profile for High-Volume Environments. RFC Editor, RFC 5019 (September 2007)
[8] DigiCert: DigiCert: Delay of revocation for EV audit inconsistency incident (2020), https://bugzilla.mozilla.org/show_bug.cgi?id=1651828
[9] DigiCert: Inconsistent EV audits (2020), https://bugzilla.mozilla.org/show_bug.cgi?id=1650910
[10] Durumeric, Z., Adrian, D., Mirian, A., Bailey, M., Halderman, J.A.: A search engine backed by Internet-wide scanning. In: Proc. ACM CCS (2015)
[11] Google: CRLSets, https://dev.chromium.org/Home/chromium-security/crlsets, last accessed: September 2020
[12] Google: Certificate lifetimes (2020), https://chromium.googlesource.com/chromium/src/+/master/net/docs/certificate_lifetimes.md
[13] Google Trust Services: Certificate Policy v1.3, OID 1.3.6.1.4.1.11129.2.5.3; Last accessed: Jan. 21, 2021.
[14] Gustafsson, J., Overier, G., Arlitt, M., Carlsson, N.: A first look at the CT landscape: Certificate Transparency logs in practice. In: Proc. PAM (Mar 2017)
[15] Internet Security Research Group: Certification Practice Statement, Version 3.0 (Oct 2020), http://cps.letsencrypt.org
[16] Kim, D., Kwon, B.J., Kozák, K., Gates, C., Dumitras, T.: The broken shield: Measuring revocation effectiveness in the Windows code-signing PKI. In: Proc. USENIX Security (Aug 2018)
[17] Koblitz, N.: Elliptic curve cryptosystems. Math. of C. **48**(177) (1987)
[18] Korzhitskii, N., Carlsson, N.: Characterizing the root landscape of Certificate Transparency logs. In: Proc. IFIP Networking (June 2020)
[19] Korzhitskii, N., Carlsson, N.: Dataset for "Revocation Statuses on the Internet" (2021), https://www.ida.liu.se/~nikca89/papers/pam21.html
[20] Laurie, B., Langley, A., Kasper, E.: Certificate Transparency. RFC 6962 (2013)
[21] Let's Encrypt: 2020.02.29 CAA Rechecking Bug, https://community.letsencrypt.org/t/2020-02-29-caa-rechecking-bug/114591/3
[22] Let's Encrypt: Download affected certificate serials for 2020.02.29 CAA Rechecking Incident (Mar 2020), https://letsencrypt.org/caaproblem/
[23] Liu, Y., Tome, W., Zhang, L., Choffnes, D., Levin, D., Maggs, B., Mislove, A., Schulman, A., Wilson, C.: An end-to-end measurement of certificate revocation in the Web's PKI. In: Proc. IMC (2015)
[24] Mozilla: (2020), https://blog.mozilla.org/security/2020/07/09/reducing-tls-certificate-lifespans-to-398-days/
[25] O'Brien, D.: Certificate Transparency Enforcement in Chrome and CT Day in London (2018), https://groups.google.com/a/chromium.org/d/msg/ct-policy/Qqr59r6yn1A/2t0bWblZBgAJ, last accessed: Jan. 2021
[26] Rivest, R.L., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public-key cryptosystems. Communications of the ACM **21**(2), 120–126 (1978)
[27] SANS Internet Storm Center: SSL CRL activity, https://isc.sans.edu/crls.html, last accessed: September 2020
[28] Santesson, S., Myers, M., Ankney, R., Malpani, A., Galperin, S., Adams, C.: X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. RFC Editor, RFC 6960 (2013)
[29] Scheitle, Q., Gasser, O., Nolte, T., Amann, J., Brent, L., Carle, G., Holz, R., Schmidt, T.C., Wählisch, M.: The rise of Certificate Transparency and its implications on the Internet ecosystem. In: Proc. IMC (2018)
[30] Sectigo: Certificate search, https://crt.sh, last accessed: September 2020
[31] Smith, T., Dickinson, L., Seamons, K.: Let's Revoke: Scalable Global Certificate Revocation. In: Proc. NDSS (2020)
[32] Starfield Technologies, LLC: Certificate Policy and Certification Practice Statement, Version 4.9 (October 2020), http://certificates.godaddy.com/repository/, last accessed: Jan. 21, 2021.
[33] Zhu, L., Amann, J., Heidemann, J.: Measuring the Latency and Pervasiveness of TLS Certificate Revocation. In: Proc. PAM (2016)
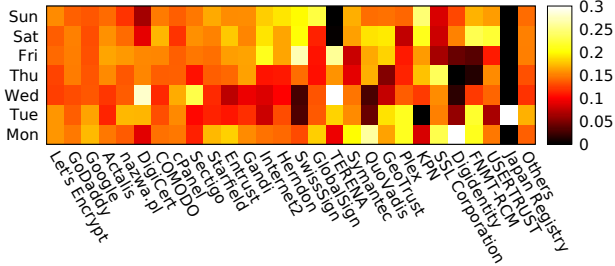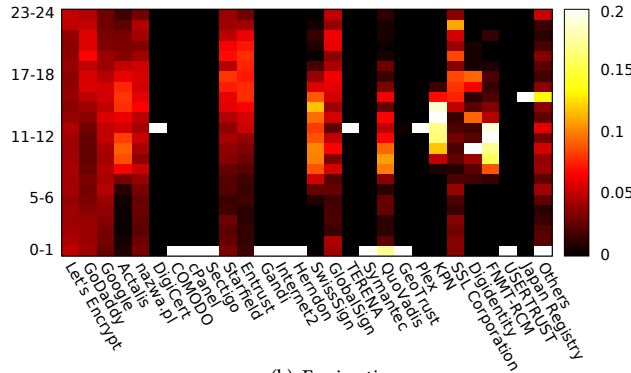
(a) End of revocation status

(b) Start of certificate validity period

Figure 9: Weekly distribution of certificate-validity-start day for the revoked certificates and last-status-change day (from "Revoked" to something else).

(a) Expiry day

(b) Expiry time

Figure 10: Expiration time of revoked certificates.

APPENDIX A. EXTRA ON CA BEHAVIOR

We have already seen that different CAs have different practices for handling of revocation statuses. Here we present the obtained day-of-week distributions that capture *when* CAs change the "Revoked" status to something else (Figure 9(a)); compare this to the distribution of the first certificate validity day (Figure 9(b)). We note the weak weekly patterns. While more than half of the CAs issue significantly fewer certificates with start dates during weekends (dark areas for Sat/Sun in Figure 9(b)), we did not observe such weekly patterns for the revocation status changes. Instead, only a few CAs have spikes of revocation status changes on a certain day (white squares in Figure 9(a)). For example, Starfield, GoDaddy (part of Starfield), and Digidentify update most of their statuses on Friday, and Japanese Registry on Sunday (Monday Japanese time). The distributions suggest that the relation between last-status-change and certificate-validity-start days is not straightforward. Some of the CAs have even weekly distributions for both processes, which may suggest higher levels of automation (e.g., Let's Encrypt, Google, Actalis, cPanel, Gandi, Herndon). Among the large CAs, DigiCert stands out with their pronounced weekly patterns for both processes. Similarly, there are differences in the daily and hourly distributions of the expiry times selected for certificates (Figure 10). Here, some of the large CAs (e.g., Let's Encrypt, GoDaddy, Google, GlobalSign) spread expiry times both across the week and the hours of the days, whereas other large CAs (e.g., DigiCert, Comodo, cPanel, Sectigo) always set certificates to expire at the same time of day. These differences may not have major security implications, however, they demonstrate the lack of a standardized policy for managing the statuses of expired certificates.

APPENDIX B: RESPONSES FROM THE CAS

We contacted 8 organizations that operate the CAs for which we observed at least one status change from "Revoked" to "Good". However, we did not find a contact email for one CA that no longer operates: AT&T Wi-Fi Services. We received responses from 5 organizations: Starfield (GoDaddy), Japan Registry, Entrust, ACCV, and Atos. The CAs that responded confirmed that they had issued the certificates in question and provided varying explanations for their behavior. Two CAs argued that their use of "Good" statuses was motivated by the standard [28], which states that "at a minimum, this positive response [i.e., a "Good" response] indicates that no certificate with the requested certificate serial number currently within its validity interval is revoked." One of these two CAs also stated that they "are going to consult with the community to clarify the requirements, and then, follow it." We believe that CAs should avoid changing the status of revoked certificates to "Good" at any time.