An Internet-wide View on HTTPS Certificate Revocations: Observing the Revival of CRLs via Active TLS Scans

Markus Sosnowski, Johannes Zirngibl, Patrick Sattler, Juliane Aulbach, Jonas Lang, and Georg Carle
Technical University of Munich, Germany
TUM School of Computation, Information and Technology; Department of Computer Engineering
Chair of Network Architectures and Services
{*sosnowski, zirngibl, sattler, aulbach, langj, carle*}*@net.in.tum.de*

***Abstract*—A global decentral Public Key Infrastructure (PKI) is a key element of trusted and secure communication over the Internet. Such a PKI enables trust inference through digital signatures. However, the irrevocable nature of signatures and the complexities involved in distributing revocation information pose significant challenges. Recent updates to the root store policies of Mozilla and Apple now mandate that each Certificate Authority (CA) must publish Certificate Revocation Lists (CRLs) on the Common CA Database (CCADB) as of October 2022. This policy shift enables new approaches for acquiring a comprehensive view of certificate revocations within the Transport Layer Security (TLS) ecosystem. This work investigates the impact of the new CRLs on certificate revocation research, whether they are sufficient to gain a comprehensive view, and how the current revocation methods compare. We conducted weekly Internet-wide TLS measurements to collect X.509 certificates over port 443 for two years starting in March 2022. These scans resulted in 1.1 billion valid leaf certificates, including 4.5 million revoked certificates we identified using the Online Certificate Status Protocol (OCSP), CRLs, CCADB CRLs, and OCSP stapling. Our findings show that acquiring a comprehensive view of certificate revocations is challenging, primarily via the OCSP. Compared to the other methods, our analyses indicate that the CCADB CRLs provided the most complete view of global certificate revocations. They covered nearly the entirety of valid leaf certificates, found 44% more revocations than alternative methods, and less than 0.3% of the revocations were exclusively visible via the OCSP or conventional CRLs.**

1. Introduction

Secure and trusted communication is essential to our modern Internet. A foundation for achieving such communication is a global X.509 Public Key Infrastructure (PKI) that can authenticate and infer a level of trust in a communication partner. Certificates assure that the party we interact with is the one they claim to be through cryptographic signatures, preventing man-in-the-middle attacks. Therefore, these X.509 certificates play a crucial role in Transport Layer Security (TLS) and authenticate almost any website visitable over the Internet. However, digital signatures cannot be undone, so there must be other means to revoke the given trust. Currently, there are two main approaches: Certificate Revocation Lists (CRLs) and the Online Certificate Status Protocol (OCSP). CRLs contain a list of all revoked certificates from one Certificate Authority (CA), and the OCSP can be used to actively request the revocation status of a single or few certificates. CAs embed the support for either method in each certificate. However, neither of the approaches is ideal, and both have performance, reliability, or privacy issues [9].

To collect Internet-wide revocations, CRLs are ideal because a single Hypertext Transfer Protocol (HTTP) request provides the necessary information for each CA. Creating a single database of all revocations can be used to better understand the certificate ecosystem or to develop and maintain novel approaches for distributing revocation information like CRLite [26]. However, only a fraction of valid certificates have CRLs included by the CAs.

Starting October 1, 2022, new root store policies from Mozilla [33] and Apple [2] announced that CAs must publish CRLs in the Common CA Database (CCADB) for every intermediate certificate that the respective root stores should trust. These new CRLs contain the revoked certificates issued by the intermediate, effectively forcing CAs to provide CRLs if they have not already done so.

This policy shift provides new possibilities to collect and analyze certificate revocations on a large scale. Suddenly, a single repository contains references to CRLs covering all relevant TLS ecosystem revocations.

This paper examines the impact the CCADB CRLs have on certificate revocation research, whether the information provided is sufficient to collect a comprehensive and global view, and how the different revocation methods compare. We conducted a long-term study of certificates collected via active Internet-wide TLS measurements and revocations collected through the OCSP, CRLs, CCADB CRLs, and stapled OCSP responses.

We provide the following contributions:

- *i)* reasoning about a data collection pipeline that can be used to collect a global view on certificate revocations;
- *ii)* analysis covering $1.1\,\mathrm{G}$ (1.1×10^9) valid certificates and their revocation statuses collected in the last two years between March 1, 2022, and January 4, 2024;
- *iii)* comparison of current revocation methods, reasons, and differences among top issuers; and
- *iv)* analysis of a single Internet-wide snapshot comparing OCSP stapling with the other revocation methods and inspecting actively used revoked certificates.

2. Background

Enabling trusted communication over the Internet is a challenging task. Cryptographic algorithms can verify whether the private part of an asymmetric key pair made a digital signature using the public part only. If this can be verified, we call the signature valid. If we trust a party in control over a private-public key pair, we can transfer this trust to any content signed with this key pair. However, it is infeasible to manually define the trustfulness of every party we interact with on the Internet. PKIs are a concept to transitively pass trust using digital signatures from a set of roots to intermediate and leaf parties, assuming each involved party includes only other trusted parties and no party gives away their private key. CAs are the trusted parties allowed to function as issuers passing trust to their subjects. TLS [35] is currently the *de facto* standard for encrypted communication on the Internet [25]. It mainly uses an X.509 PKI [4] to authenticate peers and to ensure a server can only serve content for the domain names it was allowed to. Each TLS server identifies itself with an X.509 certificate. Such certificates are basically public keys enriched with additional data like domain names and a signature from the same or another certificate. Servers must provide all intermediate certificates necessary to verify the chain of trust up to one of the root certificates provided by the user, embedded in the application, or included in the Operating System (OS). This way, trust for a TLS server can be determined quickly and in a decentralized manner.

However, a digital signature cannot be undone, and another channel for up-to-date revocation information is needed. The two main approaches to distributing certificate revocation are CRLs [4] and the OCSP [36]. The following sections describe the approaches in more detail.

2.1. Certificate Revocation Lists (CRLs)

A CRL [4] is a list containing the serial numbers and revocation times from every revoked certificate a CA has issued. Optionally, they can include a reason for each revocation. Certificates can be removed from the list after they expire [5]. CRLs must be downloaded separately for each CA; the available endpoints are typically included directly in a certificate. According to the CA/Browser Forum Baseline Requirements [5], a certificate must contain the HTTP URL of the CA's CRL service. Like a certificate, CAs sign their CRLs. Recently, Apple and Mozilla updated their root store policies [2, 33], such that each included CA must additionally provide their CRL endpoints through the CCADB. The CCADB is a central repository for root and intermediate certificates included within the products and services of the CCADB Root Store Operators [32]. The CCADB mainly covers HTTPS certificates but also contains code-signing and S/MIME certificates. In contrast to the CRLs distribution endpoints embedded in the certificates, CAs can split their CRLs across multiple endpoints and publish all of them in the CCADB. To obtain the complete list of revocations, the CRLs from these endpoints have to be combined. However, checking revocations with CCADB CRLs is more complex because it is unclear which CRL might contain a specific certificate revocation. After all, a CCADB entry contains only information for intermediate certificates, which means the issuing certificate has to be known. Either type of CRL has the downside of getting very large, and it can be inefficient for an application to download the entire CRL every time it wants to check a single certificate. This drawback is even more severe in face of mass-revocation events like after the Heartbleed vulnerability [42]. To make revocation checking more efficient for applications, OCSP was designed.

2.2. Online Certificate Status Protocol (OCSP)

The OCSP [36] allows applications to check the revocation status of a single or few certificates. Like the CRL distribution points embedded in the certificates, CAs can include an OCSP endpoint in their issued certificates. Applications can send OCSP requests to these endpoints to receive a signed response. The response notifies the application whether the certificate is revoked; in case of revocation, it can include a time and a reason. However, the OCSP has privacy issues because when users browse the Internet, their browser should make OCSP requests for almost every site they visit, effectively enabling CAs to track the browsing behavior of single users. Additionally, Chung *et al.* [9] observed outages of OCSP responders during their study; hence, their availability cannot be guaranteed, and applications tend to treat an unavailable OCSP responder as a successful validation. Attackers could exploit this behavior by intercepting OCSP messages. Further, fetching the responses takes time and can slow down the loading time of websites. Therefore, OCSP stapling has been designed.

OCSP stapling [35] is an extension to the OCSP protocol where the TLS server requests a signed OCSP response from the CA beforehand and forwards it to clients during the TLS handshake. This approach can speed up the loading time of websites and hide the user behavior from the CA. However, Chung *et al.* [9] and Sosnowski *et al.* [39, 40] showed that the presence of stapled OCSP responses is non-deterministic; hence, they cannot be guaranteed nor enforced due to the implementations of web servers and browsers treating them just as a nice-to-have feature.

2.3. Derived Revocation Approaches

The distributed nature of the OCSP and CRLs makes it difficult and costly for end devices to collect the revocation data. Several approaches were proposed to improve this by providing an aggregated list of revocations. Mozilla developed OneCRL [19] and the Chromium Projects developed CRLSets [41]. Both are a list of revocations curated by the browser vendors and pushed to the user in a single format. This format allows for fast and efficient revocation checking; however, the approaches are currently only used for intermediate certificates or emergencies. A CRLite approach was proposed by Larisch *et al.* [26] and later adopted by Mozilla [22]. They used Bloom filters to achieve scalable and efficient revocation checks. However, all of these aggregation approaches have the prerequisite that every revocation must be known first. Until recent changes in the CA policies of Apple and Mozilla, not all

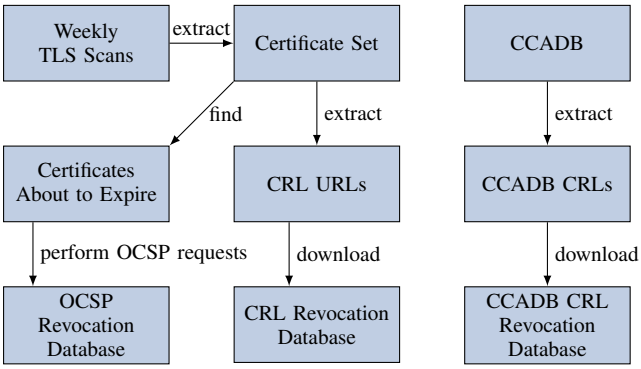

Figure 1. Data collection pipeline used to collect certificate revocations for this work over time. Each step was performed daily.

CAs provided CRLs (including Let's Encrypt [16]), and it was challenging to collect this data across the Internet.

3. Methodology

Tracking certificate revocations on the Internet can be challenging because of its decentralized nature. No central repository exists; instead, multiple methods have been developed to distribute this information. Therefore, we implemented our own data collection pipeline to create a dataset that is as comprehensive as possible with the possibilities we have.

Figure 1 shows an overview of our data collection pipeline. Our analyses are primarily based on a comprehensive collection of X.509 certificates for three reasons:

i) OCSP information can only be requested for a known certificate.
ii) CRLs do not contain the actual certificates (only the serial number), preventing further studies.
iii) We focus our analysis on certificates used on the Internet.

Hence, the pipeline started with our weekly Internet-wide TLS measurements that we used to extract and generate a single set of X.509 certificates. The pipeline worked as follows: as soon as a new TLS scan was finished, we extracted all newly observed X.509 certificates and added them to a single certificate set. From this set, we identified all certificates that were going to expire, and requested their revocation status from the provided OCSP endpoint. Only checking expiring certificates allowed us to limit the necessary OCSP requests, reduce the load on the OCSP responders, and collect the revocation data in a reasonable amount of time. We extracted all CRL distribution points from our certificate set and downloaded each CRL daily. The CRLs endpoints listed in the CCADB can be downloaded independently from the certificates; hence, we created a script that, independently from the rest of the pipeline, would access the CCADB, extract the CCADB CRLs, and download each CRL daily.

Section 3.1 gives details about our weekly TLS scans, section 3.2 provides further information about implementing the different revocation methods, and section 3.3 describes how we matched the collected raw revocation data to the individual certificates.

3.1. Collecting Certificates With Weekly Internet Measurements

To collect certificates used on the Internet, we conducted weekly TLS scans. We focused on servers with an open port 443, the standard HTTPS port, expecting a high rate of servers providing X.509 certificates. We used ZMap [13] on the complete IPv4 address scanning for open ports. Then, we used the goscanner [18] to perform a TLS handshake and collect the certificates. To collect further certificates provided only through requesting the respective domain name, we additionally relied on DNS resolutions (A and AAAA records for IPv4 and IPv6 addresses, respectively) for more than $600\,\mathrm{M}$ domains collected from sources such as:

- full zonefiles from the Centralized Zone Data Service (CZDS), $1.1\,\mathrm{k}$ in total, *i.a.*, *.com*, *.net*, and *.org*;
- top lists, *i.a.*, Umbrella [10], Majestic [30], Chrome UX Report [20], Chromium HSTS Preload [7], Cloudflare Radar [11]; and
- domains extracted from certificates in Certificate Transparency (CT) logs

With the help of the DNS resolutions, we performed TLS scans a second time, using the domains as Server Name Indication (SNI). However, we filtered the resolutions for servers with an open port 443 to prevent unnecessary requests. For IPv4, we already know the open ports from our ZMap scans; for IPv6, we performed an additional port scan with ZMapv6 [17]. We did not exclude addresses from aliased prefixes to cover Content Delivery Networks (CDNs), *e.g.*, Fastly or Cloudflare as reported by Zirngibl *et al.* [43].

The goscanner checks during scan time whether the certificate provided by the server is *valid*, utilizing only functions provided by the standard Golang library. Essentially, it confirms that the certificate meets basic X.509 requirements, a chain of trust exists to the Debian X.509 root store relying only on the provided peer certificates, and ensures that the server performed a correct signature with the certificate's private key. Moreover, the library checks if the requested domain appears as a Subject Alternative Name (SAN) when scans are conducted with SNIs.

3.2. Collecting Internet-wide Revocation Information

We collected revocation information through four methods: OCSP requests, stapled OCSP responses, CRLs included in certificates, and CRLs listed in the CCADB.

All entries in the CCADB are publicly available, and we downloaded each listed CRL. Similarly, we downloaded every CRL we could extract from the collected certificates. We only considered HTTP CRL distribution points because all certificates we analyzed contained an HTTP CRL distribution point, and only $0.01\,\%$ additionally provided a Lightweight Directory Access Protocol (LDAP) endpoint. Collecting revocation data through the OCSP is more complex because we had to actively request the information for each certificate. We requested the revocation status for each certificate only twice to limit the necessary requests. The first time was 30 days, and

the second time was one day before they were going to expire. The main reasons for performing only two requests were to limit the load on OCSP responders and to acquire the data in a reasonable time. For a comprehensive view of OCSP revocations, a single request right before expiry should be enough because revocation is permanent (unless the CA returns a "certificateHold" reason); hence, the status should remain for the whole certificate's lifetime. However, the request should not be made too late, as the statuses disappear quickly after certificate expiration [23]. There is a trend towards short-lived certificates: the CA/Browser Forum [5] restricts the lifetime of certificates to less than 398 days, and authorities like Let's Encrypt go beyond that and issue certificates for only 90 days [1]. Therefore, we should be able to collect most revocations over a reasonable amount of time.

3.3. Assigning Revocation Information to Certificates

Our data collection pipeline can collect large amounts of revocation data over time. However, this information consists only of the revoked serial numbers and not the certificates. Serial numbers are unique among a single CA; hence, we had to create a mapping of the CRLs to the individual certificates to compare them globally.

We could perform OCSP requests for concrete certificates only. Hence, it was trivial to assign the response to said certificate: we only checked whether the serial number in the response was the same as the one in the certificate. Similarly, we matched CRLs to certificates depending on the CRL distribution point included in the respective certificate. If we could download and successfully parse a CRL, we flagged the certificate either as "revoked" or "not revoked" depending on whether or not the CRL contained the certificate's serial number. Matching CCADB CRLs was more complex because we first had to find the entry for the issuing certificate listed in the CCADB that contained the link to the respective CRL. We matched the Authority Key Identifier in a certificate to the Subject Key Identifier in potential issuing certificates and confirmed these candidates by validating their signatures. Both identifiers are X.509 extensions indicating a suitable issuing certificate [4]; however, only checking the signature can guarantee the relation. All valid leaf certificates we analyzed contained these extensions. If one of the CCADB CRLs contained the serial number of the checked certificate, we flagged it as "revoked".

3.4. Ethical Considerations

Our measurements follow the ethical considerations and best practices proposed by Dittrich *et al.* [12] and Partridge *et al.* [34]. We focused on publicly visible services, collected no user data, scanned with a limited rate, maintained a custom blocklist, and informed about our research with expressive rDNS records, websites on our scanning machines, and WHOIS information. In case someone reached out to us, we responded to all requests and, if requested, included their IP addresses and domains in our blocklist, preventing further scans to target their infrastructure. We rate-limited our OCSP requests to 100 requests per CA to avoid disturbing their operation. We confirmed this rate with Let's Encrypt because they were the CA we had to query the most.

TABLE 1. COLLECTED X.509 CERTIFICATES FROM OUR WEEKLY INTERNET MEASUREMENTS FROM 2022-03-01 TO 2024-01-04, BROKEN DOWN INTO VARIOUS CATEGORIES.

	Certificates	Fraction of Parent
Total (with expiry date)	1.20G	
⌞ Leafs	1.19G	98.1%
⌞ Valid	1.12G	94.5%
⌞ with OCSP endpoint	1.12G	100.0%
⌞ with CRL(s)	241.51M	21.5%
⌞ with CCADB CRL(s)	1.12G	100.0%
Valid Leafs with Revocation Information Obtained		
⌞ via OCSP	674.20M	60.0%
⌞ via CRLs	240.76M	21.4%
⌞ via CCADB CRLs	1.12G	100.0%

4. Results

Using the methodology from the previous section, we collected a vast data set of X.509 certificates used in the TLS ecosystem and their revocations. The data set allowed us to compare the effectiveness of the different methods CAs use to announce revocations.

Table 1 gives an overview of the certificates collected with the help of our weekly Internet-wide IPv4 and IPv6 scans (with and without SNI) between March 1, 2022, and January 4, 2024. We collected more than $1.1\,\mathrm{G}$ valid leaf certificates for our following analyses. In the table, *Fraction of Parent* lists the percentage in relation to the parent category; *e.g.*, $94.5\,\%$ of all leaf certificates were valid. We counted a certificate as valid if our TLS scanner was able to verify the validity during scan time at least once. Because we partitioned the certificates in our data collection pipeline according to the expiration date, the total number of certificates contains no certificate without an expiration date. However, the following analyses are based only on valid certificates (*cf.*, section 3.1) that must contain such a date. We saw not a single certificate that was valid but did not contain an OCSP endpoint. However, only $22\,\%$ of these certificates offered a CRL. However, the root store policy from Mozilla and Apple changed and forced CAs to provide CRLs through the CCADB after October 1, 2022. With the help of the information in the CCADB, we could match almost all valid leaf certificates to one or multiple CCADB CRLs. We started downloading CRLs and performing OCSP requests daily after July 21, 2022. To ensure a comprehensive dataset from the start, we used certificates from TLS scans from the previous four months to collect CRLs and perform OCSP requests. This four-month period was chosen as a provisional measure, inspired by Let's Encrypt's policy of restricting their certificates to a 90-day validity period [1]. We collected the CCADB CRLs beginning October 16, 2022. Despite our efforts, the data indicates that we were unable to successfully collect a revocation status for every certificate and method. We performed OCSP requests twice: once for certificates that expired in 30 days and the second time if they expired the next day. Hence, not all certificates were checked, and sometimes, we received unsuccessful

TABLE 2. OBSERVED VALID LEAF CERTIFICATES REVOKED AFTER THE POLICY CHANGE TOGETHER WITH THE PROVIDED REASON.

Revocation Reason	OCSP	CRLs	CCADB
None	61.91%	9.90%	34.38%
Cessation Of Operation	23.85%	43.45%	30.76%
Superseded	7.52%	40.35%	30.00%
Affiliation Changed	4.64%	5.13%	3.52%
Key Compromise	1.75%	1.13%	1.30%
Unspecified	0.31%	0.00%	0.00%
Privilege Withdrawn	0.03%	0.04%	0.04%
Certificate Hold	0.00%	0.00%	
Total	1.52M	3.10M	4.48M

responses from the CA containing no information. In rare cases, the download of a CRL failed, or our library could not parse a response.

4.1. Collected Revocation Data

With our Internet scans, we could identify around four million certificates revoked at some point during our study. This does not mean we scanned them while they were revoked; such cases are presented later in section 4.4.

4.1.1. Overview. Table 2 shows the total number of revoked valid leaf certificates we could identify according to the respective revocation mechanism after the policy change on October 1, 2022. We can see that we retrieved the highest number of revocations via the CCADB CRLs. In total, we observed $4.5\,\mathrm{M}$ revoked certificates, *i.e.*, $0.37\,\%$ of our total observed certificates. Note that we performed OCSP requests for only around $\frac{3}{5}$ of the certificates. Additionally, CAs can provide reason codes for their revocations as defined in RFC 5280 [4]. According to the Mozilla root store policy [33], CAs should only use the reasons *Key Compromise*, *Privilege Withdrawn*, *Cessation of Operation*, *Affiliation Changed*, and *Superseded*. The first two can be seen as more critical reasons as they indicate potential misuse of a certificate, either because the key was (or could be) compromised, the certificate owner intentionally misused the certificate, or it contained misleading information. The latter three reasons can be seen as less harmful because they describe circumstances in the normal life cycle of certificates, *e.g.*, discontinued websites, changing subject names, or certificates being replaced by updated versions. As expected, most revocations fall into one of the three less harmful reasons. However, we can also see that the majority of revocations do not have a reason at all. We observed only a single certificate with the reason code of *Certificate Hold* (the only temporary revocation method) via OCSP and the CRLs; however, not over the CCADB CRLs. The issuer was "NETLOCK Kft.", a minor issuer from which we collected only $1.45\,\mathrm{k}$ certificates. Note that it is not guaranteed that the provided reason is correct as they are defined purely by the respective CA.

4.1.2. Collected Revocations Over Time. All the revocations we collected contained a revocation date. Similar to the revocation reason, this date is set by the CA and does not have to correlate with the time the revocation was published. Figure 2 presents the number of revoked valid leaf certificates we observed according to the revocation time set by the CA. Some of the collected certificates had already been revoked for a long time. Our earliest revocation was from April 27, 2020, but the figure shows only revocations starting in 2022. We can see that we were able to collect the most revocations each day from the CCADB CRLs after the root store policy change from Apple and Mozilla. We observed several revocation bursts, the biggest one in November 2023 (*i.e.*, from the 23th to 25th). These were almost completely caused by revocations from GoDaddy and Let's Encrypt; however, we were not able to identify the reason for these events. We can see two drops in the number of OCSP revocations during the last year, *i.e.*, between `2023-03` and `2023-06`, and between `2023-09` and `2023-11`. The two drops were caused by an unnoticed failure of our OCSP fetching service lasting several weeks each. In retrospect, we could no longer check these certificates because they have already expired. These two events highlight the difficulty in creating a comprehensive set of revocations via OCSP and the huge advantage of the CCADB CRLs.

To conclude, we showed that the CCADB CRLs are able to provide the most comprehensive view on revocations after the policy change on October 1, 2022. Moreover, we saw indicators that the certificate ecosystem is dominated by major players, analyzed in the following section.

TABLE 3. TOP ORGANIZATIONS IDENTIFIED FROM THE ISSUER NAME AND THEIR REVOCATIONS OBSERVED AFTER THE POLICY CHANGE. WHILE ALL CERTIFICATES SUPPORTED OCSP, NOT ALL INCLUDED CRL ENDPOINTS IN THEIR CERTIFICATES.

Organization	Revocations			Certificates	
	OCSP	CRLs	CCADB	Total	with CRLs
Let's Encrypt	567.0k		1.1M	796.4M	
Google Trust [...]	232.9k	429.5k	444.2k	93.5M	100%
cPanel, Inc.	132.0	202.0	205.0	58.0M	100%
Sectigo Limited	39.0k	5.1k	96.7k	54.8M	1%
GoDaddy.com, Inc.	490.5k	2.3M	2.3M	36.1M	100%
Cloudflare, Inc.				30.2M	100%
DigiCert Inc	65.7k	202.0k	278.7k	22.0M	19%
ZeroSSL	30.5k		47.7k	11.9M	
Amazon	3	13	13	8.3M	100%
Microsoft [...]	337.0	517.0	292.0	2.3M	100%

4.2. Top Certificate Organizations

The certificate ecosystem is dominated by major organizations. We identified the organization through the *Issuer* field embedded in the certificates. The top 10 are listed in Table 3. To better compare the revocation methods, we only considered revocations after the policy change.

Note that the provided name does not have to correlate with the actual CA issuing the certificate. This is discussed by Ma *et al.* [29] and Ayer [3]: in short, CA certificates are valuable, and private keys can be passed on to other organizations; in these cases, the issuer's name might contain outdated information because a certificate cannot be changed after receiving its signature. Additionally, some CAs use white-labeled intermediate certificates to issue certificates for a client. For example, Cloudflare appears as an organization; however, the actual CA controlling the private key of this certificate is DigiCert. Table 3

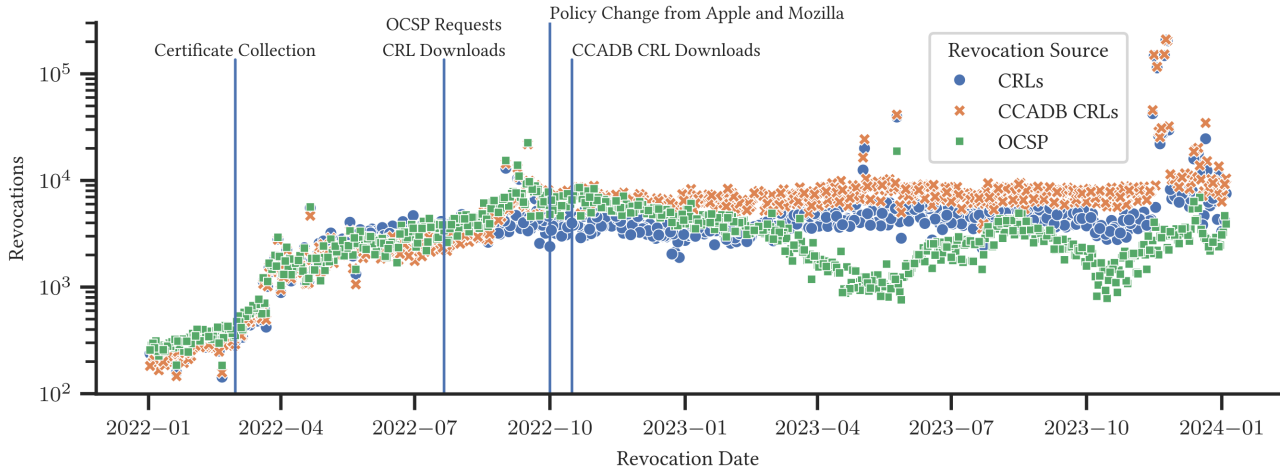

Figure 2. Number of revocations according to the revocation date provided by the CAs for observed valid leaf certificates. Only showing dates starting 2022. We marked the date at which we started collecting the certificates and when we started collected data from the respective revocation sources.

TABLE 4. CCADB CRLs REVOCATION REASONS FOR THE TOP 10 ISSUING ORGANIZATIONS (IDENTIFIED FROM THE ISSUER NAME) WITHOUT CLOUDFLARE (FROM WHICH WE OBSERVED NO REVOCATION) AFTER THE POLICY CHANGE. EMPTY CELLS INDICATE THAT THERE ARE EXACTLY ZERO OBSERVATIONS BEFORE ANY ROUNDING WAS APPLIED.

Revocation Reason	Let's Encrypt	Google Trust [...]	cPanel, Inc.	Sectigo Limited	GoDaddy.com, Inc.	DigiCert Inc	ZeroSSL	Amazon	Microsoft [...]
None	94.41%	0.01%	84.39%	34.30%	0.32%	96.46%	99.83%	92.31%	5.82%
Superseded	2.60%	0.00%		64.47%	52.62%	2.34%			18.84%
Key Compromise	2.02%		9.27%	0.79%	1.45%	0.06%	0.16%		63.36%
Cessation Of Operation	0.97%	99.99%		0.37%	39.13%	1.04%		7.69%	5.14%
Privilege Withdrawn			6.34%	0.04%	0.00%	0.07%	0.00%		
Affiliation Changed				0.02%	6.47%	0.03%			6.85%
Total	1.12M	444.18k	205.00	96.68k	2.31M	278.65k	47.68k	13	292

shows that the revocation behavior is highly dependent on the organization. This already starts with the support for CRLs; some organizations do not support them at all and others only for some of their certificates, explaining the low number of CRL revocations in general. We observed no revocation for Cloudflare and just a few from Amazon after the policy change.

Analyzing the top issuing organizations reveals significant differences in their revocation practices. To investigate these practices further, we listed the reason codes the top 10 issuing organizations provided in their CCADB CRL revocations in Table 4. We focused only on the CCADB CRLs to simplify the analysis because our previous analyses showed that they provided the most comprehensive view. The table reveals that Let's Encrypt was the primary source of empty reasons, likely due to customers needing to provide them. We can see that not all issuers make use of the *Cessation of Operation*, *Affiliation Changed*, and *Superseded* reasons that describe circumstances in the expected lifetime of certificates.

In conclusion, the major certificate issuers treat revocations differently. Some do not utilize CRLs, while others apply them only to a subset of their certificates. Additionally, there are cases where no certificate revocation was observed. Moreover, their usage of revocation reasons differs significantly, and not all revoke certificates as part of the standard certificate life cycle. Overall, the CCADB CRLs provided the most complete data set.

TABLE 5. COMPARING THE CONSISTENCY OF REVOCATIONS COLLECTED THROUGH DIFFERENT MEANS AFTER THE POLICY CHANGE.

Method A	Method B	Only A	Only B	$A \wedge \neg B$	$\neg A \wedge B$	$A \wedge B$
OCSP	CCADB	0	2.9M	11.2k	104.1k	1.5M
CRLs	CCADB	4	1.4M	11.3k	32.7k	3.1M
CRLs	OCSP	2.2M	675.2k	32.0k	529	841.1k

4.3. Consistency Across Revocation Methods

Measuring the number of revocations that we were able to collect through the different revocation methods allowed us to understand how comprehensive the collected data was. However, this does not reveal if the revocations are actually the same.

We compared each combination of OCSP, CRLs, and CCADB CRLs and presented the results in Table 5. In the table, we compare two methods, A and B, and list the number of valid leaf certificates each method identified as revoked after October 1, 2022. "Only A" means that method A was the only available method for a given certificate, and the status of that certificate was revoked. "$A \wedge \neg B$" means it was revoked through method A but not via method B, even though there was revocation information available through method B. "$A \wedge B$" indicates

TABLE 6. BREAKDOWN OF THE SERVERS IN OUR LATEST INTERNET SCAN, THE VALIDITY OF THEIR CERTIFICATES, THE USE OF OCSP STAPLING, AND PROBLEMS WE WERE FACING ANALYZING THE STAPLED RESPONSES.

Category	Targets	Fraction of Parent
Total	733.46M	
⌞ with valid certificates	515.11M	70.23%
⌞ providing stapled OCSP responses	224.16M	43.52%
⌞ ASN.1 unparseable	149	0.00 %
⌞ unsuccessful OCSP response	4.88k	0.00%
⌞ successful OCSP response	224.15M	100.00%
⌞ invalid signature	51.83M	23.12%
⌞ wrong Serial Number	15.68k	0.01%

that it was revoked via both methods. The same principle applies to the remaining columns.

We had to add the "only" categories because, for several certificates, we were able to obtain revocation information only through one method. We expected to observe some inconsistencies. On the one hand, CRLs contain only revoked certificates and no information about valid ones. We counted each certificate as not revoked via a CRL if its serial number was not listed in the CRL, and we successfully collected revocation information from this CRL at least once. Still, it is possible that downloading or parsing later versions of the CRL failed, and we could not collect the revoked statuses. On the other hand, the OCSP can confirm a certificate is not revoked; however, it is very expensive to constantly check certificates, and it is possible that the certificate got revoked after we checked it via the OCSP and before we checked it a second time right before it expired. This could explain some of the inconsistencies when comparing the different methods.

Overall, we can confirm that the CCADB CRLs provided the most comprehensive view on certificate revocations. We saw $44\,\%$ more revocations from the CCADB CRLs, and less than $0.3\,\%$ of revocations were only received through OCSP or regular CRLs.

4.4. Revocations Observed During Scan-Time

Until this point we analyzed certificates and their revocations throughout the last year. We tracked the certificates even when they disappeared from the Internet. In this section, we only used our latest Internet scan (conducted from December 26 to 31, 2023) to analyze revoked certificates that were actively used during scan time. Additionally, we analyzed the stapled OCSP responses TLS servers appended to the TLS handshakes. Clients can request the validity status of a certificate from a TLS server through a Status Request extension. This mechanism is only defined for OCSP [14] at the moment; hence, it can only request stapled OCSP responses.

Table 6 gives an overview of the servers we scanned. We interpret a scanned combination of IP address and domain as target. The table shows that $70.2\,\%$ of the connections we performed included a valid certificate. While all of these certificates supported OCSP, and we requested a stapled response from all targets, only $43.5\,\%$ made use of this feature to inform us about the revocation status of their leaf certificate. However, not all of these responses were usable: 149 could not be parsed with our library, $4.9\,\mathrm{k}$ were OCSP responses with an error (*unauthorized* or *try later*) originating from the CA, $15.7\,\mathrm{k}$ targets provided a response for the wrong certificate, and for $23.1\,\%$ of the stapled OCSP responses we could not verify the signature with one of the peer certificate provided in the TLS handshake or embedded in the stapled response.

TABLE 7. OBSERVATIONS OF REVOKED CERTIFICATES DURING SCAN-TIME. THE COVERAGE WHERE WE KNOW THE REVOCATION STATUS IS DEPENDING ON THE REVOCATION SOURCE.

Revocation Source	Valid Targets		Valid Certificates	
	Covered	Revoked	Covered	Revoked
OCSP	222.0M	35.2k	67.1M	23.8k
Stapled OCSP resp.	188.7M	4.8k	51.7M	1.8k
CRLs	251.2M	102.8k	58.1M	37.7k
CCADB CRLs	513.6M	147.7k	156.1M	69.5k
Total	515.1M	147.7k	156.9M	71.1k

When conducting the Internet scan we saw thousands of valid but revoked certificates, as shown in Table 7. These certificates were sometimes used by multiple targets; hence, the number of targets with a revoked certificate is even higher. Interestingly, we received stapled OCSP responses that informed us about the revoked certificate the server was using. We only considered a stapled response if it contained a valid signature and the serial numbers matched.

In conclusion, in our latest Internet-wide IPv4 and IPv6 scan, $0.05\,\%$ of the valid certificates were revoked already at scan time. The amount would likely increase even further over time because $0.37\,\%$ of the certificates we tracked through the last year were revoked at some point. Sometimes, revocations were even forwarded through stapled OCSP responses.

5. Related Work

Researchers have repeatedly investigated the TLS ecosystem and certificate revocations. Both are constantly adapting, and new insights can be gained as time passes.

5.1. Active Scanning for Certificate Analyses

Multiple works analyzed the TLS ecosystem via active scans. Farhan and Chung [15] conducted an Internet-wide study exploring the evolution of TLS certificates from 2013 to 2021. They collected $359\,\mathrm{M}$ certificates with active scans of the IPv4 address space over eight years. However, they did not use SNIs nor IPv6, explaining the lower number of certificates. $34\,\%$ were valid certificates. Chung *et al.* [8] investigated invalid certificates from 2013 to 2015 through active measurements of the IPv4 address space. Back then, the rate of invalid certificates was $88\,\%$ out of the total $80\,\mathrm{M}$. We performed our Internet-wide measurements with SNIs over two years, revealing a valid rate of $95\,\%$ out of the total $1.2\,\mathrm{G}$ certificates.

Kumar *et al.* [24] investigated certificate misissuance on the Internet and developed the Zlint tool. They utilized the CCADB to find the owner of issuing certificates in case the issuer's name was misleading. Interestingly, they observed different top issuers; *e.g.*, we could not see Symantec anymore, which likely relates to the major browsers distrusting them [31] in 2018.

5.2. Certificate Revocations

Several works investigated certificate revocations. In 2014, Zhang *et al.* [42] analyzed the Heartbleed event and whether CAs revoked all vulnerable certificates properly. Their analysis is based only on CRLs, and they used IPv4 scans filtered to the Alexa Top 1 Million domains. According to their results, 73 % of the vulnerable certificates were not reissued, and more than 87 % were not revoked three weeks after the vulnerability was disclosed. One year later, Liu *et al.* [28] investigated certificate revocations over CRLs, OCSP requests, and OCSP stapling. However, they only used the OCSP in case no CRL was present in a certificate. Back then, 99.9 % of certificates offered CRLs, and 95.0 % included an OCSP endpoint. We observed CRLs for just 22 % of the certificates, and all our valid certificates included OCSP endpoints. Their data is based on weekly IPv4 HTTPS scans from 2013 to 2015, and they observed 39 M certificates and 5 M valid leaf certificates. Interestingly, they observed a very high rate of revoked certificates due to the Heartbleed [42] vulnerability: 8 % compared to the 0.37 % we identified. 2.8 % of the servers they analyzed provided stapled OCSP responses; we observed 46 %. Through repeated probing of a small random subset, they revealed that 18 % supported OCSP stapling in theory; however, the servers did not always forward the stapled responses. This aligns with the work of Chung *et al.* [9] in 2018, where they uncovered that the Internet is not ready for OCSP Must-Staple because the presence of the stapled OCSP responses cannot be guaranteed. Their analysis is based on 490 M certificates obtained through IPv4 scans and CT logs, and they identified 112 M valid certificates. They observed outages in OCSP responders and sometimes received stapled responses that were unparseable, had unmatching serial numbers, or had an incorrect signature. This aligns with our observations. They showed a steady rise of OCSP-enabled certificates, which we can now confirm to have reached 100 %. A more recent view on certificate revocations was provided by Korzhitskii and Carlsson [23] in 2021. They analyzed all 48 M valid Censys certificates expiring between March 2 and April 1, 2020. They identified 1 M revoked certificates. Their revocation rate of 2.18 % is higher than ours; however, their data covered a mass-revocation event of 773 k revocations from Let's Encrypt. They tracked the revocation status with CRLs and the OCSP shortly before and up to 100 days after expiry. Their results revealed that most statuses disappear a few days after expiry (*e.g.*, for Let's Encrypt), but this varies on the issuing CA. The distribution of revocation reasons included in the CRLs is similar to ours. Only the rate of revocations announced as *Cessation of Operation* was a bit higher in our data set. Interestingly, they observed 589 certificates that changed their status from revoked back to good. Halim *et al.* [21] investigated such cases in more detail and found at least two additional cases. They discussed that the effect could be cache-related. Recently, Cerenius *et al.* [6] investigated certificate revocations in context of replacement practices. They collected revocations via CRLs and OCSP but did not use the CRLs published in the CCADB. Their data set is compromised of IPv4 address space scans without SNIs and CT log data from 2020; however, they only analyzed 1.2 M certificates.

An interesting work has been proposed by Trevor *et al.* [37], where they proposed a novel and scalable format to construct CRLs. They used a bit vector to compress the revocation information for each CA highly. They show the feasibility with a data set of 84 M valid certificates, where 1 M million were revoked.

In conclusion, several related works investigated the topic of X.509 certificate revocations. It is challenging to acquire a global view of the TLS ecosystem, and every work is based on different time frames and uses different subsets of certificates that are used on the Internet as basis of their analyses. However, to the best of our knowledge, no work has yet analyzed the impact the CCADB CRLs have on certificate revocation research yet.

6. Discussion

Analyzing vast numbers of certificates has several challenges and can give fascinating insights into the Internet. We want to discuss some aspects of our work in the following paragraphs.

6.1. Declining Popularity of CRLs

When comparing our observations on certificates supporting CRLs to related work, we see an interesting decline. For example, Liu *et al.* [28] observed 99.9 % of their certificates containing CRLs in 2015. In our work, we observed CRLs for 22 % of the certificates. Moreover, certificates offering OCSP endpoints increased from 95 % to 100 %. The comparison indicates a declining popularity of CRLs and that CAs focus on offering OCSP endpoints. CRLs can get very large and it can become inefficient to download the full list to check a single certificate. The development makes sense if the main use case are end devices that want to check the status of a single or few certificates a time. However, the CCADB CRLs rapidly changed the picture in 2022 and we observed a coverage of 100 % of these new CRLs. In contrast to OCSP, the CCADB CRLs enable large-scale analyses and to realize novel revocation distribution methods (*cf.*, section 2.3).

6.2. Revocations for Invalid Certificates

In our analyses, we focused on valid leaf certificates. However, some of the invalid certificates contained OCSP endpoints or CRL distribution points as well. In some of these cases, our scanner identified certificates as invalid because they were already expired at scan time or the SNI did not match. In other cases, we saw indicators of a private PKI. The largest was from Cloudflare, supporting both OCSP and CRLs, where we saw 263 k certificates. These certificates indicate that we scanned origin servers from Cloudflare customers, which should only be accessed through the CDN.

6.3. OCSP Stapling for Revoked Certificates

Interestingly, we received several stapled OCSP responses from servers informing us that their certificate was actually revoked (see section 4.4). This means that these web servers did not make the presence of the stapled

OCSP responses dependent on the included status. While it would make sense for a malicious actor to drop such responses in the hope a client will treat its absence the same as a good response, we think this is exactly how web servers should handle revoked certificates, or else it would defeat the whole purpose of the stapled responses and this way administrators might notice the problem. Most of these revocations had either none or the *Superseeded* reason. We checked some of the websites, and Chrome, Safari, and Firefox noticed the revoked certificate. 58 of the certificates included a common name listed on the Tranco [27] top list. When manually checking these sites, we could not see the revoked certificate anymore; apparently, the issue had been fixed.

6.4. Considering Revocations in Active Measurement Papers

Certificate revocations happen all the time. Related work revealed that the rate can be as high as $8\,\%$ [28] and $2.18\,\%$ [23], resulting in $405\,\mathrm{k}$ and $1\,\mathrm{M}$ total revocations, respectively. While we observed just $0.37\,\%$ revoked certificates, this still resulted in around $4.5\,\mathrm{M}$ revocations in the last two year due to our larger data set. This is an amount that should not be neglected in studies focusing on valid certificates. However, many research papers do not consider revocations. Previously, collecting a massive amount of revocation statuses from every analyzed certificate was either incomplete when considering CRLs or impracticable with the OCSP. However, the CRLs from the CCADB offer a great new possibility for researchers to to incorporate revocations in their research:

i) it is a central repository with links to all relevant revocation databases, and downloading them is straightforward;
ii) all certificates should be covered according to the root store policies of Mozilla [33] and Apple [2]; and
iii) our results revealed that they provided a comprehensive view on global certificate revocations.

While results might not be influenced significantly in normal circumstances if the revocation rate is low (*e.g.*, observed in this paper), in the face of mass-revocation events, it can be much higher and significantly impact the results if neglected.

6.5. Is our Certificate Data Set Complete?

We could see more than $42.0\,\mathrm{M}$ revoked serial numbers listed on the CCADB CRLs after October 1, 2022. However, we were only able to identify around $4.5\,\mathrm{M}$ revoked certificates. This means our methodology could not capture a large portion of the revoked certificates from which some might have never been used on the Internet. Still, we collected more certificates than related work scanning the IPv4 address space (*e.g.*, $359\,\mathrm{M}$ valid certificates over 8 years [15] compared to the $1.1\,\mathrm{G}$ valid certificates we collected in two years) because we scanned with SNIs. However, we do not know about every possible (sub-) domain; hence, the data set is likely incomplete. On the other side, we think it is sufficient to answer our research questions.

7. Conclusion

This work provides an Internet-wide view of certificate revocations in the TLS ecosystem. It proposes a data collection pipeline that collects revocation information of X.509 certificates used on the Internet and compares different methods of announcing a revocation: the OCSP, CRLs, CCADB CRLs, and stapled OCSP responses.

Our results showed that the CCADB CRLs provided the most comprehensive view of global certificate revocations: they covered $100\,\%$ of the $1.1\,\mathrm{G}$ valid leaf certificates we collected, revealed $44\,\%$ more revocations, and provided almost a complete superset of the revocations announced by other means. In addition, conventional CRLs covered only $22\,\%$ of valid leaf certificates, the OCSP was too costly to provide a constant up-to-date database, and we received stapled OCSP responses for $44\,\%$ of scanned targets with a valid certificate.

This paper investigated and confirmed the effectiveness of a new possibility to collect and analyze certificate revocations: the CRLs published in the CCADB. Initiated by root store policy changes from Mozilla [33] and Apple [2]: as of October 1, 2022, CAs must provide these CRLs for all intermediate certificates capable of issuing new certificates.

The ability to quickly collect the full set of relevant certificate revocations on the Internet should allow researchers to better understand and monitor the certificate ecosystem, improve measurement studies that are based on valid certificates, and help develop and maintain novel approaches for distributing revocation information on the Internet.

Acknowledgment

This work was supported in part by the German Federal Ministry of Education and Research (BMBF) under the projects PRIMEnet (16KIS1370), 6G-life (16KISK002) and 6G-ANNA (16KISK107), by the German Research Foundation (HyperNIC, DFG grant no. CA595/13-1, and by the European Union's Horizon 2020 research and innovation program (grant agreement no. SLICES-SC 101008468 and SLICES-PP 101079774).

References

[1] Josh Aas. Why ninety-day lifetimes for certificates?, 2015. (accessed Feb. 28, 2024). URL: https://letsencrypt.org/2015/11/09/why-90-days.html.
[2] Apple. Apple Root Certificate Program. (accessed Mar. 25, 2024). URL: https://www.apple.com/certificateauthority/ca_program.html.
[3] Andrew Ayer. The SSL certificate issuer field is a lie, 2023. (accessed Feb. 28, 2024). URL: https://blog.apnic.net/2023/03/08/the-ssl-certificate-issuer-field-is-a-lie/.
[4] Sharon Boeyen, Stefan Santesson, Tim Polk, Russ Housley, Stephen Farrell, and David Cooper. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. RFC 5280, 2008. doi:10.17487/RFC5280.
[5] CA/Browser Forum. Baseline requirements for the issuance and management of publicly-trusted certificates, version 1.8.6, 2022. (accessed Feb. 28, 2024). URL: https://cabforum.org/wp-content/uploads/CA-Browser-Forum-BR-1.8.6.pdf.
[6] David Cerenius, Martin Kaller, Carl Magnus Bruhner, Martin Arlitt, and Niklas Carlsson. Trust Issue(r)s: Certificate Revocation

and Replacement Practices in the Wild. In *Proc. Passive and Active Measurement (PAM)*, 2024. doi:10.1007/978-3-031-56252-5_14.

[7] Chromium. HSTS Preload List. (accessed May 3, 2024). URL: https://hstspreload.org/.

[8] Taejoong Chung, Yabing Liu, David Choffnes, Dave Levin, Bruce MacDowell Maggs, Alan Mislove, and Christo Wilson. Measuring and Applying Invalid SSL Certificates: The Silent Majority. In *Proc. ACM Int. Measurement Conference (IMC)*, 2016. doi:10.1145/2987443.2987454.

[9] Taejoong Chung, Jay Lok, Balakrishnan Chandrasekaran, David Choffnes, Dave Levin, Bruce M. Maggs, Alan Mislove, John Rula, Nick Sullivan, and Christo Wilson. Is the Web Ready for OCSP Must-Staple? In *Proc. ACM Int. Measurement Conference (IMC)*, 2018. doi:10.1145/3278532.3278543.

[10] Cisco. Umbrella Popularity List, 2024. (accessed Feb. 28, 2024). URL: https://s3-us-west-1.amazonaws.com/umbrella-static/index.html.

[11] Cloudflare. Cloudflare Radar: Domain Rankings. (accessed May 3, 2024). URL: https://radar.cloudflare.com/domains.

[12] David Dittrich, Erin Kenneally, et al. The Menlo Report: Ethical principles guiding information and communication technology research. *US Department of Homeland Security*, 2012.

[13] Zakir Durumeric, Eric Wustrow, and J. Alex Halderman. ZMap: Fast internet-wide scanning and its security applications. In *Proc. USENIX Security Symposium*, 2013. doi:10.5555/2534766.2534818.

[14] Donald E. Eastlake. Transport Layer Security (TLS) Extensions: Extension Definitions. RFC 6066, 2011. doi:10.17487/RFC6066.

[15] Syed Muhammad Farhan and Taejoong Chung. Exploring the Evolution of TLS Certificates. In *Proc. Passive and Active Measurement (PAM)*, 2023. doi:10.1007/978-3-031-28486-1_4.

[16] Aaron Gable. A New Life for Certificate Revocation Lists, 2022. (accessed Feb. 28, 2024). URL: https://letsencrypt.org/2022/09/07/new-life-for-crls.html.

[17] Oliver Gasser, Quirin Scheitle, Pawel Foremski, Qasim Lone, Maciej Korczynski, Stephen D. Strowes, Luuk Hendriks, and Georg Carle. Clusters in the Expanse: Understanding and Unbiasing IPv6 Hitlists. In *Proc. ACM Int. Measurement Conference (IMC)*, 2018. doi:10.1145/3278532.3278564.

[18] Oliver Gasser, Markus Sosnowski, Patrick Sattler, and Johannes Zirngibl. TUM goscanner, 2023. (accessed May 3, 2024). URL: https://github.com/tumi8/goscanner.

[19] Mark Goodwin. Revoking Intermediate Certificates: Introducing OneCRL, 2015. (accessed Feb. 28, 2024). URL: https://blog.mozilla.org/security/2015/03/03/revoking-intermediate-certificates-introducing-onecrl/.

[20] Google. Chrome User Experience Report. (accessed May 3, 2024). URL: https://developer.chrome.com/docs/crux.

[21] Adam Halim, Max Danielsson, Martin Arlitt, and Niklas Carlsson. Temporal Analysis of X.509 Revocations and their Statuses. In *Proc. IEEE European Symposium on Security and Privacy (EuroS&P)*, 2022. doi:10.1109/EuroSPW55150.2022.00032.

[22] J.C. Jones. Introducing CRLite: All of the Web PKI's revocations, compressed , 2020. (accessed Feb. 28, 2024). URL: https://blog.mozilla.org/security/2020/01/09/crlite-part-1-all-web-pki-revocations-compressed/.

[23] Nikita Korzhitskii and Niklas Carlsson. Revocation Statuses on the Internet. In *Proc. Passive and Active Measurement (PAM)*, 2021. doi:10.1007/978-3-030-72582-2_11.

[24] Deepak Kumar, Zhengping Wang, Matthew Hyder, Joseph Dickinson, Gabrielle Beck, David Adrian, Joshua Mason, Zakir Durumeric, J. Alex Halderman, and Michael Bailey. Tracking Certificate Misissuance in the Wild. In *Proc. IEEE Symposium on Security and Privacy (S&P)*, 2018. doi:10.1109/SP.2018.00015.

[25] Craig Labovitz. Internet traffic 2009-2019. In *Proc. Asia Pacific Regional Internet Conf. Operational Technologies*, 2019.

[26] James Larisch, David Choffnes, Dave Levin, Bruce M. Maggs, Alan Mislove, and Christo Wilson. CRLite: A Scalable System for Pushing All TLS Revocations to All Browsers. In *Proc. IEEE Symposium on Security and Privacy (S&P)*, 2017. doi:10.1109/SP.2017.17.

[27] Victor Le Pochat, Tom Van Goethem, Samaneh Tajalizadehkhoob, Maciej Korczyński, and Wouter Joosen. Tranco: A research-oriented top sites ranking hardened against manipulation. In *Network and Distributed System Security*, 2019. doi:10.14722/ndss.2019.23386.

[28] Yabing Liu, Will Tome, Liang Zhang, David Choffnes, Dave Levin, Bruce Maggs, Alan Mislove, Aaron Schulman, and Christo Wilson. An End-to-End Measurement of Certificate Revocation in the Web's PKI. In *Proc. ACM Int. Measurement Conference (IMC)*, 2015. doi:10.1145/2815675.2815685.

[29] Zane Ma, Joshua Mason, Manos Antonakakis, Zakir Durumeric, and Michael Bailey. What's in a name? Exploring CA certificate control. In *Proc. USENIX Security Symposium*, 2021.

[30] Majestic. The Majestic Million. (accessed Feb. 28, 2024). URL: https://majestic.com/reports/majestic-million/.

[31] Mozilla. CA/Symantec Issues. (accessed Feb. 28, 2024). URL: https://wiki.mozilla.org/CA/Symantec_Issues.

[32] Mozilla. Common CA Database. (accessed Feb. 28, 2024). URL: https://www.ccadb.org/.

[33] Mozilla. Mozilla Root Store Policy, Version 2.8. (accessed Mar. 25, 2024). URL: https://github.com/mozilla/pkipolicy/blob/2.8/rootstore/policy.md.

[34] Craig Partridge and Mark Allman. Addressing Ethical Considerations in Network Measurement Papers. In *Proceedings of the 2015 ACM SIGCOMM Workshop on Ethics in Networked Systems Research*, 2016. doi:10.1145/2793013.2793014.

[35] Eric Rescorla. The Transport Layer Security (TLS) Protocol Version 1.3. RFC 8446, 2018. doi:10.17487/RFC8446.

[36] Stefan Santesson, Michael Myers, Rich Ankney, Ambarish Malpani, Slava Galperin, and Dr. Carlisle Adams. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP. RFC 6960, 2013. doi:10.17487/RFC6960.

[37] Trevor Smith, Luke Dickinson, and Kent Seamons. Let's Revoke: Scalable Global Certificate Revocation. In *Network and Distributed System Security*, 2020. doi:10.14722/ndss.2020.24084.

[38] Markus Sosnowski, Johannes Zirngibl, Patrick Sattler, Juliane Aulbach, Jonas Lang, and Georg Carle. An Internet-wide View on HTTPS Certificate Revocations: Observing the Revival of CRLs via Active TLS Scans. In *Proc. IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*, July 2024. doi:10.1109/EuroSPW61312.2024.00038.

[39] Markus Sosnowski, Johannes Zirngibl, Patrick Sattler, Georg Carle, Claas Grohnfeldt, Michele Russo, and Daniele Sgandurra. Active TLS Stack Fingerprinting: Characterizing TLS Server Deployments at Scale. In *Proc. Network Traffic Measurement and Analysis Conference (TMA)*, 2022.

[40] Markus Sosnowski, Johannes Zirngibl, Patrick Sattler, Georg Carle, Claas Grohnfeldt, Michele Russo, and Daniele Sgandurra. EFACTLS: Effective Active TLS Fingerprinting for Large-scale Server Deployment Characterization. *IEEE Transactions on Network and Service Management*, 2024. doi:10.1109/TNSM.2024.3364526.

[41] The Chromium Projects. CRLSets. (accessed Feb. 28, 2024). URL: https://www.chromium.org/Home/chromium-security/crlsets/.

[42] Liang Zhang, David Choffnes, Dave Levin, Tudor Dumitraş, Alan Mislove, Aaron Schulman, and Christo Wilson. Analysis of SSL Certificate Reissues and Revocations in the Wake of Heartbleed. In *Proc. ACM Int. Measurement Conference (IMC)*, 2014. doi:10.1145/3176244.

[43] Johannes Zirngibl, Lion Steger, Patrick Sattler, Oliver Gasser, and Georg Carle. Rusty Clusters? Dusting an IPv6 Research Foundation. In *Proc. ACM Int. Measurement Conference (IMC)*, 2022. doi:10.1145/3517745.3561440.