On the Certificate Revocation Problem in the Maritime Sector

Guillaume Bour(✉), Karin Bernsmed, Ravishankar Borgaonkar, and Per Håkon Meland

SINTEF Digital, Trondheim, Norway
{Guillaume.Bour,Karin.Bernsmed,Ravi.Borgaonkar, Per.H.Meland}@sintef.no

Abstract. Maritime shipping is currently undergoing rapid digitalization, but with increasing exposure to cyber threats, there is a need to improve the security of the ship communication technology used during operations across international waters, as well as close to local shores and in ports. To this aid, there are ongoing standardization efforts for an international maritime Public Key Infrastructure, but the inherent properties of limited connectivity and bandwidth make certificate revocation a problematic affair compared to traditional Internet systems. The main contribution of this paper is an analysis of certificate revocation techniques based on how they fulfil fundamental maritime requirements and simulated usage over time. Our results identify CRLs (with Delta CRLs) and CRLite as the two most promising candidates. Finally, we outline the pros and cons with these two different solutions.

Keywords: Cyber security · Public key infrastructure · Certificate revocation · Maritime · Shipping

1 Introduction

Maritime shipping is currently undergoing rapid digitalization. The introduction of new communication technologies onboard ships, such as the upcoming VHF Data Exchange System (VDES) [21], enables a wide variety of new digital services, such as digital ship reporting, electronic port clearance, search and rescue communications, vessel traffic services and broadcast of maritime safety information. These services will all require information security, and a prevalent solution to establish and deploy a Public Key Infrastructure (PKI) for distributing digital certificates and securing the integrity and confidentiality of the information exchange. Several different research groups have in parallel worked to define, implement and test the characteristics of a PKI for the maritime domain [7,12,13,23], and the concept has now been acknowledged by IMO[1] and brought into the ongoing standardization by IALA[2] [5]. However, there is

[1] International Maritime Organization (IMO). http://www.imo.org.
[2] International Association of Marine Aids to Navigation and Lighthouse Authorities (IALA), https://www.iala-aism.org/.

M. Asplund and S. Nadjm-Tehrani (Eds.): NordSec 2020, LNCS 12556, pp. 142–157, 2021.
https://doi.org/10.1007/978-3-030-70852-8_9

a significant challenge related to certificate revocation yet to be solved. This is a crucial part of any PKI, and unlike typical Internet applications that can be constantly online, ships tend to be offline for long periods of time or sailing in the open sea where connectivity and bandwidth can be both poor and expensive. The consequences could be delayed awareness of revocations and less trust in the PKI itself. Even though some previous work has evaluated and compared different revocation mechanisms [15,27,28,30], there are no previous studies that address the specific maritime challenges.

The main contribution of this paper is an analysis of certificate revocation techniques based on requirements fulfilment and simulated use in a maritime setting over time. The paper is organised as follows. Section 2 provides the background to our work, including a description of the envisioned maritime PKI as well as an overview of existing solutions for certificate revocation. Section 3 presents the fundamental requirements and Sect. 4 gives an analysis of and simulation benchmarks for the solution candidates. Section 5 discusses these results and Sect. 6 concludes the paper.

2 Background

2.1 The Maritime PKI

A normal PKI depends on a hierarchy of trust, e.g. as depicted in Fig. 1. There are three layers in this model:

1. A Trusted International Root Certificate Authority (CA) that issues certificates to its subordinates. The root CA is envisioned to be operated by IMO, since they are a trusted entity by the majority of maritime stakeholders around the world.
2. A number of (intermediate) Issuing National CAs, that would typically be the Flag State administrations associated with each country.[3]
3. End entities, which are the ships, ports and coastal services that need to communicate securely.

In addition, an entity called "Revocation issuer" responsible for issuing information about invalid certificates, will be needed. This role could also be handled by the root CA.

2.2 Existing Revocation Solutions

Revocation of certificates in a PKI ecosystem can happen for a number of reasons, such as change of ship name, change of association between the end entity and the issuing CA, or compromise of the corresponding private key [10]. Affected entities should be informed as fast as possible after a revocation, and we have described existing revocation mechanisms that we have considered for the maritime sector.

[3] Every merchant ship has to be registered under a jurisdiction, called the flag state, which has the responsibility to enforce regulations over vessels registered under its flag.

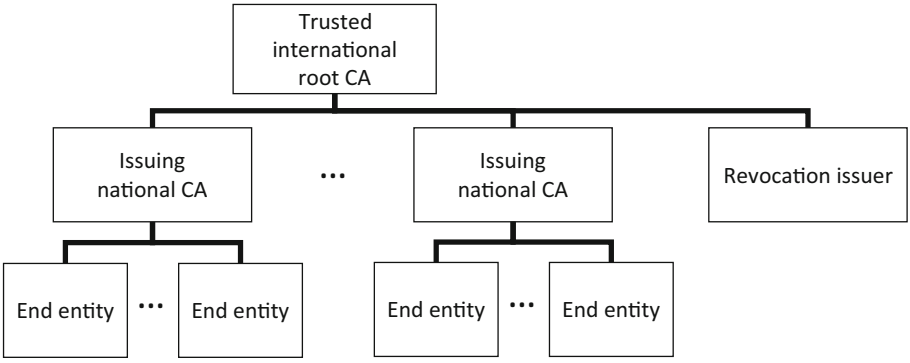

Fig. 1. The PKI trust hierarchy.

Certificate Revocation List (CRL). (RFC 5280 [10]) are issued at regular time intervals. When a user wants to check the validity of a certificate, he needs to have a local copy of the CRL installed. This is usually be achieved by pulling the CRL from the CA's CRL distribution endpoint. While this method works well for PKIs with relatively few end entities, the solution does not scale well, since a full and complete CRL will list all (unexpired) certificates that have been revoked. To counter this problem, delta-CRLs can be used, which only include certificates whose revocation status has changed since last update. A drawback with CRLs (and delta-CRLs) is that the validity check is done "offline" and there is a risk that end entities accept certificates that have been revoked.

The use of CRLs to revoke certificates in the Maritime PKI has been proposed by Froystad et al. [13] and the Maritime Cloud Development Forum [12]. Figure 2 presents the principle: a CRL issuer collects CRLs from all entities entitled to issue such lists, and creates a joint CRL that is distributed to all the end entities in the PKI. However, neither [13] nor [12] specify the use of delta-CRLs or discuss the risk of using obsolete CRLs.

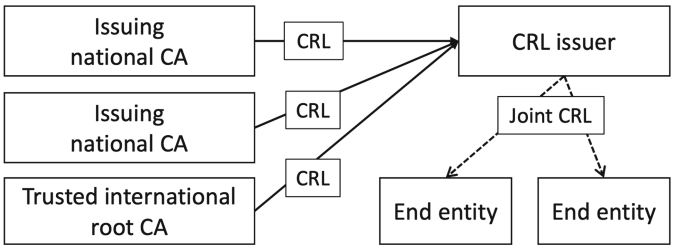

Fig. 2. CRLs from multiple sources are collected and distributed to end entities through a CRL issuer [13].

Online Certificate Status Protocol (OCSP). (RFC 6960 [26]), was designed to provide more timely revocation information compared to CRLs. The protocol is "online"; to verify a certificate's validity, the user sends an OCSP request to the CA, asking for the status of one or more given certificate(s). The CA will respond with the certificate(s)'s status (good/revoked/unknown), along with a response validity interval. However, the solution comes with some drawbacks:

- If the CA is unavailable, the validity of certificates cannot be confirmed.
- Increased latency since the OCSP request needs to be confirmed before the certificate can be used. If the CA is unreachable, it can take several seconds before the request times out,[4] which will be a no-go for time critical applications.
- While the OCSP response is signed, error messages are not. This may open up to interception attacks [18].
- The identity of the user is revealed to the CA each time the user sends an OCSP request, possibly creating privacy issues.

To counter OCSP bandwidth issues, RFC 5019 [11] defines a lightweight profile that minimises the communication bandwidth and client-side processing.

OCSP Stapling/OCSP Must Staple. To solve some of OCSP's problems, RFC 6961 [24] and RFC 7633 [25] (commonly referred to as "OCSP Stapling" and "OCSP Must Staple", respectively) define a method where a server makes a request to the OCSP service and get a signed message with its certificate status that it can then "staple" to its certificate during the Transport Layer Security (TLS) handshake. This method is more efficient than the original OCSP because the server can cache OCSP messages, thus reducing the latency. This also partially solves the privacy issue with OCSP, as the user does not need to reveal its identity each time. Still, OCSP Stapling suffers from being an "online" solution, which is not suitable for all maritime applications.

CRLSet [3]**/OneCRL.** [1] are currently used by Google and Firefox to revoke CA certificates stored in their web browsers. Revocation lists are built internally by the respective companies and pushed to the clients daily. The lists only include a subset of the most critical revoked certificates. The main benefit is that bandwidth is minimised. In contrast to OCSP, these solutions do not reveal communication patterns, as all end entities receive the same lists. The common downside of CRLSet and OneCRL is that end entities must be online on a daily basis.

CRLite. allows to push all TLS certificate revocations to all browsers. Initially described in a research paper by Larisch et al. [19], it has since 2019 been tested by Mozilla Firefox as its new certificates revocation method [8]. CRLite relies

[4] Telemetry from Internet browsing with Firefox shows that OCSP requests "time out about 15% of the time, and take in average 350 ms even when they succeed" [4].

on cascading *bloom filters* to efficiently push out the revocation information, which is "a simple space-efficient randomized data structure for representing set to support membership queries" [22]. The answer to a membership query is probabilistic; if the answer is negative, we know the element is not in the set, but if the answer is positive then chances are that this is a false positive. To remove false positives, CRLite takes advantage of Certificate Transparency (CT), which aims to fix several flaws in the SSL certificate system by "providing an open framework for monitoring and auditing SSL certificates in nearly real time" [2]. In practice, certificate logs are built by collecting all certificates issued by trusted CA [2,20]. In the context of CRLite, a first bloom filter is b.uilt using the revoked certificates data and then, using the CT log, a new bloom filter (smaller than the first one) containing all false positives of the first bloom filter is built. This operation can be repeated until there is no more false positive, thus the name of cascading bloom filters.

Larisch et al. [19] emphasize the following advantages of CRLite over CRLs and OCSP:

- Small Size: About 10 MB is needed to represent the status of *all Web certificates*, and 560 kB in average for daily updates. While OCSP also covers all certificates, it is online. CRL on the other hand is not efficient to handle so many certificates.
- Update Frequency: An OCSP response is valid for 4 days in average for the Web, and a CRL has usually a lifetime of 7 days. CRLite is believed to be more up-to-date because it involves daily update.
- Failure Mode: CRLite covers all certificates and allows end entities to operate in a hard fail mode.
- Privacy: The privacy issues caused by OCSP are avoided as CRLite caches the information locally.
- Deployment and Auditing: CRLite is easily deployed and can be audited.
- Speed: Telemetric data shows that CRLite is faster than OCSP in 99% of the cases [16].

There is also some criticism towards the Bloom filter cascades. Holzhauser [14] shows that CRLite will suffer a scalability problem when the number of certificates increase. She also shows faults in the CRLite design that can lead to higher than expected number of false positives.

Short-Lived Certificates. Which are valid from a few hours up to 2–3 days, represent an alternative strategy. Firefox for instance, does not check such short-lived certificate for revocation if they are valid for less than 10 days [8]. This will efficiently remove the need to revoke certificates, given an acceptable risk that illegitimate keys will probably not be used before the certificates expire. The obvious downside is the need for frequent certificate renewal and distribution, which does not go hand in hand with maritime operations.

Table 1. Data capacity, cost and availability of different data bearers

Communication link	Shared capacity	Cost	Availability
VDES	153.6 kbps	Free	Near shore, between nearby ships
GSM/LTE	100 Mbps	0.006 USD/MB	Near shore
Low Frequency SATCOM	100–500 kbps	5–10 USD/MB	Globally
High Frequency SATCOM	100 kbps–8 Mbps	1–2 USD/MB	Globally, dependent on service provider
WiMAX/Wi-Fi	10–100 Mbps	Free	In port

3 Fundamental Requirements for Revocation

We perform an initial filtering of suitable revocation mechanisms by identifying the fundamental requirements imposed by the maritime sector. These are presented below.

While some ships call at port on a regular basis, others might be out at sea for several weeks or even months [13]. They usually rely on different technologies to get connectivity depending on their position: VDES, SatCom, GSM/LTE or even Wi-Fi when at shore. Table 1 developed by Frøystad et al. [13] gives an overview of their properties. Some will in many cases be too expensive or not available at all while at sea. It should therefore be possible to use a cache of revocation information, e.g. when vessels encounter each other on open sea.

Requirement 1 - Offline support: The revocation mechanism should be able to operate when the vessel has no internet connectivity.

Chrome and Firefox currently push incomplete CRLs with only high value certificates in them, but this is not acceptable in the maritime context. The solution needs to be complete, meaning that revocations must be shared between intermediate CAs and eventually known to all end entities. Of course, with a solution that operates offline from time to time, there will be some delay before a revocation information is available to the end entities. The update frequency of the revocation information is left for discussion.

Requirement 2 - Completeness: The revocation mechanism should inform about all revoked vessels in a timely manner.

While the revocation mechanism should not be dependent on the communication link, its bandwidth usage should be as low as possible to ensure acceptance affordability in the wider maritime community. In practice, this means within the capacities given by Table 1.

Requirement 3 - Bandwidth: The bandwidth usage of the revocation mechanism should be within the capacity of the available communication link.

4 Analysis of the Revocation Candidates

4.1 Requirements-Based Selection of the Revocation Mechanism

We now use the requirements identified in Sect. 3 to do an initial filtering of the candidate solutions.

R1 - Offline support: Amongst the previously presented revocation mechanisms, only CRL, DeltaCRL, CRLSet/OneCRL and CRLite are truly offline mechanisms, meaning that the ships can stay off-line for a long period of time and the mechanism will still work, even though it will be on outdated data. On the contrary, OCSP Stapling, OCSP Must Staple and Short-Lived Certificate all rely on periodic connection to update their staple or certificate in order to work. They are therefore not considered viable solutions for the maritime sector.

R2 - Completeness: All revocation mechanisms but CRLSet and OneCRL are complete or can be. CRLSet and OneCRL by definition only include high value revocation information to allow a quick reaction to critical events like of a CA compromise.

R3 - Bandwidth: This parameter is difficult to evaluate. A known problem with CRL is their growing size when the number of certificates in the system growths. However, the deltaCRL is less dependent on this parameter, and more on the system revocation ratio. CRLite was conceived with low-bandwidth usage in mind, but it has only been applied to the web by Mozilla, and the web has very different parameters than the maritime sector.

Table 2 shows which mechanisms fit our requirements the best. As can be seen, CRL/DeltaCRL and CRLite meet al.l three requirements, but there is an uncertainty on their respective bandwidth usage, which we analyse further below.

Table 2. Requirements vs revocation mechanisms

	CRL/Delta-CRL	OCSP	OCSP Stapling	OCSP Must Staple	CRLSet/OneCRL	CRLite	Short-Lived Certificate
R1 Offline support	✓		∼	∼	✓	✓	∼
R2 Completeness	✓	✓	✓	✓		✓	✓
R3 Bandwidth	?					?	

Legend: ✓: OK ∼: Partially OK
?: Unknown Nothing: Not OK

4.2 Bandwidth Analysis for CRL/DeltaCRL and CRLite

In order to estimate bandwidth requirements for the different revocation solutions, we need to estimate some parameters for the PKI. This includes the expected number of certificates in the system and the expected revocation frequency.

The number of merchant ships in the world is varying, but as of January 2019, there were 96,295 registered ships in the total fleet world wide, whereof 51,684 were commercial ships of 1000 gt and above [6]. In addition there will be shore users communicating with the ships (ports, VTS, applications human users etc.), but this number is lower than 1000. To simplify, we approximate that **the total number of end entities that will be enrolled in the maritime PKI will be around 100 000.** Based on the simulation, we observed that the result remains the same with more entities in the PKI, which covers the case of several certificates per ship.

Revocation of the digital certificates from the end entities in a PKI can happen for a number of reasons, but in the maritime domain, change of flag is expected to be the main driver for revocation. We foresee that the frequency of other reasons are negligible in comparison. All commercial ships must be registered with a country, which is known as its Flag State. Ships normally change their flags in connection with sale and purchase transactions. However, ship owners may also do this to avoid the stricter marine regulations imposed by their own countries. In practice, many ships are therefore registered under a flag that does not match the nationality of the vessel owner ("flag of convenience"). The Flag State with the largest number of registered ships is Panama (6465 ship as of January 2019), followed by China (4039 ships), Liberia (3456) and the Marshall Island (3454) [6]. Even though the total number of Flag States is fairly large (117 as of January 2019), we do not foresee that all of these will operate their own Intermediate CA. However, a ship will still need to obtain a new digital certificate when it changes its flag. A study from 2008 [9] provides an indication of the frequency of flag changes. The study uses data collected from 35,261 port state control inspections on 7,547 vessels, carried out between 2002 and 2008. The data shows that 25.3% of all the inspected ships have had at least one change of flag during this time period. Further, 9.5% of all the ships have had at least one change in flag since their previous inspection, where the average number of inspections per ship in this time period was 4.05. Unfortunately, the paper does not include information on how many times the ships have re-flagged, when they have changed their flags "more than once," but we can use an approximation of the **yearly revocation ratio of around 5%.**

We also need to know how long the certificates will be valid. This will impact our analysis, because when revoked certificates expire, they will not be included in the transmitted revocation information anymore. The validity period of end entity certificates in a maritime PKI was studied by several independent research groups [12,13] who proposed to set it to 3 years. We thus chose to fix this parameters to **3 years** for our simulation.

Theoretical Approach[5]

Size of the CRL: In order to evaluate the size of the CRL, we will consider the scenario in which the CRL contains the most certificates. Given that the certificates have a 3-year validity period, they are removed from the CRL once they are not valid anymore. Thus the maximum number of certificates in the CRL is 15 000 ($3 \times 100000 \times 0.05$). After doing some tests, we calculated that in average the size (in bytes) of a CRL is given by $S(n) = 277 + 50 \times n$, where n is the number of certificates to be included in the CRL (with a reason code), 277 the size of an empty CRL in bytes and 50 the average size added by the addition of a certificate to the CRL (empirically determined). So, for the maritime sector we end up with a maximum CRL size of 750 kB.

Size of the DeltaCRL. We need to have an idea of the number of certificates that will be included in it, both newly revoked certificates and those that need to be removed from the CRL. If we make the assumption that the CRL and DeltaCRL are issued on a weekly basis, then there is around 100 newly revoked certificates for each DeltaCRL. In addition, we will assume that there is about the same number of certificates removed from the CRL. Following the same formula as above, the size of the DeltaCRL should be around 10 kB.

Size of the CRLite Filter: We need to find an estimation of the size of the filter that needs to be downloaded by the end entities in order to check for the revocation status of a certificate. In their original paper on CRLite [19], the authors present a way to set the parameters of the different filters to have the smallest possible size of the overall bloom filter cascade. We followed that methodology and chose the filters' parameters to minimise the overall size of the bloom filter cascade. For those condition, the overall size of the filter is given by: $S_{bfc} = 4.92 * |R|$, where R is the set of revoked certificates. In our case, the result yields 73 800 bits, or 9.2 kB.

Size of the Delta CRLite Filter: There is no easy way to theoretically estimate the size of the delta filter for CRLite. This needs to be determined in an empirical manner.

As it can be observed from the estimated sizes of the different "payload" for each mechanism, these sizes are smaller than what is normally found and used on the Internet. However, there is still a 75-factor between the size of the CRL and the size of the optimised CRLite filter. The DeltaCRL is about the size of the optimised CRLite filter.

Empirical Approach

In order to get a better idea of the sizes for the different revocation mechanisms, we developed a PKI simulator, consisting of a Root CA, Intermediate CAs, End entities and a CRL issuer. The parameters of the simulator are its duration in time, the revocation ratio and frequency, along with the PKI (number of

[5] All the calculations below are based on x509 certificates in DER format.

intermediate CA and end entities). The simulator can also determine the growth of the PKI, as we can assume that not all ships will be part of it from day one, and thus get a better idea of what will happen when a real PKI is be deployed. The following steps are taken for each iteration:

1. Renew certificates that are about to expire.
2. Revoke random certificates based on the revocation ratio parameter.
3. Generate revocation data:
   (a) Generate new CRL.
   (b) Generate DeltaCRL.
   (c) Generate Optimised (minimum) CRLite filter.
   (d) Update CRLite filter.
   (e) Generate Delta Filter (using the updated filter).
4. Enrol new entities (if any, when growth enabled).

Our implementation of the CRLite bloom filter cascade follows Mozilla's implementation available on Github [17] and the implementation of the delta filter follows what is described in the CRLite paper [19].

For the parameters, we used 100 000 end entities and a revocation ratio of 5% as above. The revocation frequency was set to 7 days, which is the common revocation frequency for a CRL. We also estimated that the PKI will start with 1000 entities and then grow to 100 000 over a 5 year period. However, this was only to get an idea of the system evolution when integrating new components. What we really care about is the system in its "steady" state, which is why ran the simulation over a period of 20 years. Figures 3 and 4 along with Table 3 present the results of the simulation.

CRLite Vs CRL: The results presented in Fig. 3 show that the size of the CRL is much bigger than the size of any other mechanism, with an average size of 356 kB for the simulation. Even if this is much smaller than CRLs from the Internet world, this is still too big to be downloaded over a low-speed network. As presented in Table 3, even if the size of the DeltaCRL remains small (with an average of 8.6 kB), the delta filter is even smaller with an average size of 2 kB. It is also interesting to note that the size of the delta filter is almost constant once the system has reached its equilibrium (no more ship being added), and is not much influenced by big changes in the end entities certificates. Indeed, the certificates having a 3-years validity, large amount of already revoked certificates expire every 3-years, leading to substantial changes in CRL (and thus deltaCRL as well). The "waves" pattern that can be observed is the direct consequence of the initial certificates' expiration. The size of the filter is in average 39.5 kB, but like the CRL, it is based as a reference to get the delta filter, and is not sent over low-speed communication channels. The size of the optimised filter, calculated every day for comparison, is close to the size of the DeltaCRL, with an average size of 10.1 kB. Based on the analysis of the simulation, it seems like CRLite is indeed well suited for low-bandwidth usage, not only for the Internet world, but also for the maritime sector, with a payload being five times smaller than the DeltaCRL.

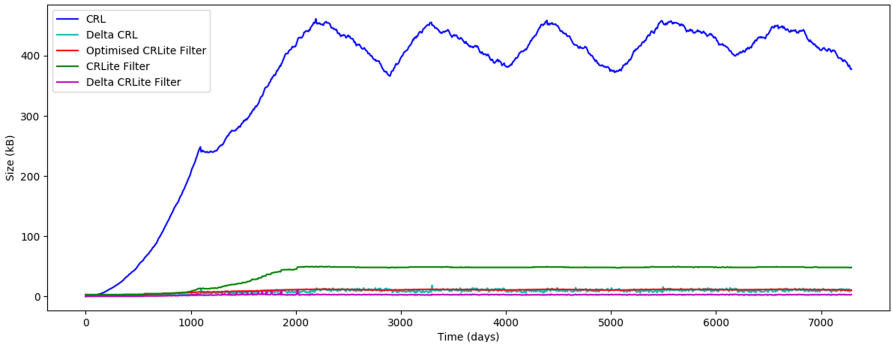

Fig. 3. Results of the PKI revocation simulation, comparing the size of the payload for CRL/DeltaCRL and CRLite filter/Delta filter, over a period of 20 years.

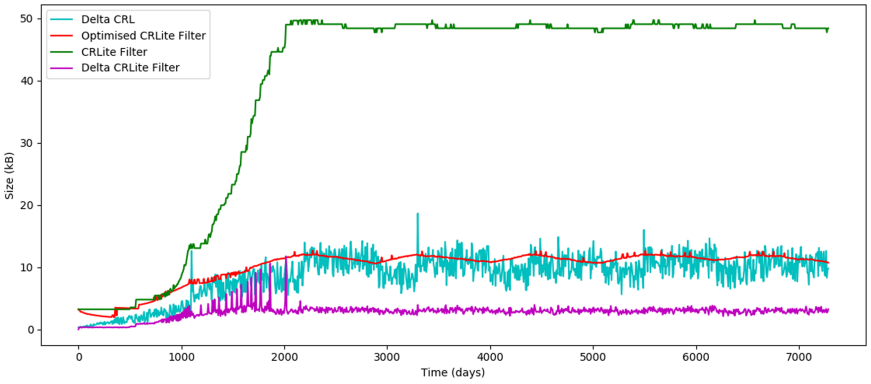

Fig. 4. Zoom on the results of the PKI revocation simulation, comparing the size of the payload for CRL/DeltaCRL and CRLite filter/Delta filter, over a period of 20 years.

Table 3. Size of the payload for different revocation mechanisms

	Theory (Max)	Simulation (Avg.)
CRL Size	750 kB	356 kB
DeltaCRL Size	10 kB	8.6 kB
Optimised filter Size	9.2 kB	10.1 kB
Filter Size	?	39.5 kB
Delta Filter Size	?	2.74 kB

5 Discussion

5.1 CRL and DeltaCRL

Using solely CRL as the revocation mechanism for the maritime PKI is not possible as shown by the simulation: the size of the PKI grows fast and does not

meet the low-bandwidth usage requirements. It must be coupled with the use of DeltaCRL.

The main advantage of using CRL/DeltaCRL is that it is a well-known and standardised revocation mechanism. It is also already implemented in commercial PKI solution and is thus more easily acceptable. However, ships can stay at sea for long periods of time, and might have to download several DeltaCRL (or the full CRL) to catch up. Moreover, it is known that CRL/DeltaCRL does not scale well for the Internet. This is true as well for the maritime sector where the constraints are even more strict regarding the internet access and the bandwidth usage. Finally, the CRLs must be collected from all CAs to create a joint CRL by the CRL issuer. This CRL is then transmitted to the end entities. That means that different states might have to trust not only the Root CA but also the CRL issuer.

5.2 CRLite for the Maritime Sector

The second solution is to adapt CRLite to the maritime sector. To the best of our knowledge, this has not been proposed before. A web browser and a ship are very different in nature, and while the main concept can remain the same, the update frequency along with the push/pull model might have to be adapted to fit the need of the maritime sector.

The first advantage of CRLite over CRL is the smaller size of the payload that needs to be delivered to the end-entities. Based on our simulation, the size of the CRLite payload is five times smaller than the equivalent payload for the DeltaCRL. As explained in the background section, CRLite relies on having both the revocation information and the valid certificates information in order to create the filter. To achieve that, CRLite relies on Certificate Transparency, which, even if it is out of the scope for this paper, harden the security of the PKI as a whole. Finally, the authors of the original CRLite paper proposed a way to create the filter in a distributive manner and not involving only the issuer. This is an interesting property for the maritime sector where different states have to collaborate but do not necessarily trust each other.

On the other hand, CRLite is a recent technology (at least compared to CRL), and is neither standardised nor field-tested, which can be an issue to be accepted by the maritime organisations. There is also a lack of formal security analysis and research done. A bachelor thesis from ETH Zurich analysed CRLite and more specifically the usage of Bloom Filter Cascade and concluded that the mechanism presents some weaknesses [14]. In particular, Bloom Filter Cascades do not adapt and scale well with the market growth. This argument does not hold for the maritime sector however, as the amount of certificates is much lower than in the Internet world, where CRLite has been proven to work with around 36M certificates (which is more certificates than the maritime PKI will ever have). Finally, another negative point compared to CRL is that the end entities will get no information on the revoked certificate. When checking for the status of a certificate with CRLite, the answer is either “valid” or “revoked.” Depending on how applications plan to handle revocation information, this can be a problem.

5.3 Common Topics to both Solutions

No matter which solution is chosen, the frequency of the updates needs to be determined. CRL and DeltaCRL are usually issued on a weekly basis. For CRLite it is on a daily basis. In our simulation however, we used a weekly basis for both methods as a mean for comparison. Choosing the update frequency comes down to answering the question: "How long do we want the ships to accept revoked certificates." The answer to this question might vary between different ship owners and flag states.

Related to this, the importance of the revocation information may vary depending on the reason why a certificate is revoked. For instance, a certificate being revoked because the ship has changed its flag state is not a security threat by itself, but a certificate revoked because CA compromise is. Different priority could thus be given to different revocation information. The notion of "scopes" is described in RFC 5280 [10], and CRLs (and DeltaCRLs) can be issued with a scope. For instance, it is possible to have a CRL for certificate revoked with the reason code "keys compromised" and another one with all the remaining reasons. It is also possible to implement different frequencies for the different scopes, thus allowing reducing the bandwidth costs as well. Splitting the revocation information in scopes is also feasible in CRLite, but as there are very few cases of key compromises compared to other reasons, creating a filter for those might not be justifiable.

How applications handle the revocation information is also another important issue. Currently, in the Internet world, browsers tend to apply a "soft-fail" techniques, meaning that if it can't verify the certificate validity, it will consider it valid, creating a feeling of false security for the user. In the maritime world, it will be important to think about the failure scenarios, how the information is communicated to the user and what are the process to respond to those failures.

5.4 Looking Elsewhere

Something that has been out of our research scope is to analyse solutions that are still on a conceptual level. For instance, a blockchain-based certificate transparency and revocation mechanism for the web has been proposed by Wang et al. [29] The idea is to remove the trust from the CA, and to transfer it to the end-entities which are in this case the browsers. Servers can then publish their certificates to a public blockchain, and a browser will accept the certificate received during the SSL/TLS negotiations if and only if it matches the ones in the public blockchain and if it is not revoked. It is very much likely that this and similar solutions will require a degree of connectivity that could be difficult to obtain in a maritime setting.

6 Conclusion

In this paper we have identified two potential candidate solutions for revocation of certificates in the maritime sector: 1) CRLs (with DeltaCRLs) and 2) CRLite

(with Delta filters). Both these solutions can operate when vessels are offline and they both inform about all revoked vessels in a timely manner. However, our results from simulating the behaviour of these two different solutions over time show that will be significant changes in terms of required bandwidth. While the size of the CRLs itself will have an average size of 365 kB, the size of the DeltaCRLs is expected to be relative small (8.6 kB). Still, CRLite is a much better solution in this respect, with 39.5 kB filter size and 2.74 kB Delta filter size. However, as explained in the discussion, there are pros and cons with both solutions and the final choice will be a trade-off between selecting a more well-known and mature technology (CRL), or going for a potentially more efficient, but less tested, solution (CRLite).

Acknowledgements. This work has been supported by the Research Council of Norway through the "Cyber Security in Merchant Shipping - Service Evolution" project with contract number 295969.

References

1. CA:RevocationPlan. https://wiki.mozilla.org/CA:RevocationPlan#OneCRL. Accessed on 08 Jun 2020
2. Certificate transparency. http://www.certificate-transparency.org/. Accessed on 08 Jun 2020
3. CRL Sets. https://dev.chromium.org/Home/chromium-security/crlsets. Accessed on 08 Jun 2020
4. Improving revocation: OCSP must-staple and short-lived certificates. https://blog.mozilla.org/security/2015/11/23/improving-revocation-ocsp-must-staple-and-short-lived-certificates/. Accessed on 08 Jun 2020
5. The technical specification of VDES. IALA Guideline G1139, Edition 3.0, June 2019
6. UNCTAD Handbook of Statistics 2019 - Merchant Fleet. https://stats.unctad.org/handbook/MaritimeTransport/MerchantFleet.html
7. CySiMS Deliverable D2.2 Using digital signatures in the maritime domain (2017)
8. Revocation Checking in Firefox (2019). https://wiki.mozilla.org/CA/Revocation_Checking_in_Firefox. Accessed on 08 Jun 2020
9. Cariou, P., Wolff, F.C.: Do port state control inspections influence flag- and class-hopping phenomena in shipping? Working Papers hal-00455155, HAL, February 2010. https://ideas.repec.org/p/hal/wpaper/hal-00455155.html
10. 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 5280, May 2008. https://tools.ietf.org/html/rfc5280
11. Deacon, A., Hurst, R.: The lightweight online certificate status protocol (OCSP) profile for high-volume environments. RFC 5019 September 2007. https://tools.ietf.org/html/rfc5019
12. Forum, M.C.D.: Identity Management and Cyber Security. IALA Input paper: ENAV19-n.n.n
13. Frøystad, C., Bernsmed, K., Meland, P.H.: Protecting future maritime communication. In: Proceedings of the 12th International Conference on Availability, Reliability and Security, pp. 1–10 (2017)

14. Holzhauser, K.: An Analysis of Bloom Filter Cascades-CRLite (2020)
15. Jain, G.: Certificate revocation: a survey (2000). http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.17.587&rep=rep1&type=pdf
16. Jones, J.: CRLite: speeding up secure browsing (2020). https://blog.mozilla.org/security/2020/01/21/crlite-part-3-speeding-up-secure-browsing/. Accessed on 08 Jun 2020
17. Jones, J.: filter-cascade (2020). https://github.com/mozilla/filter-cascade/blob/master/filtercascade/. Accessed on 08 Jun 2020
18. Langley, A.: Revocation checking and Chrome's CRL (2012). https://www.imperialviolet.org/2012/02/05/crlsets.html. Accessed on 08 Jun 2020
19. Larisch, J., Choffnes, D., Levin, D., Maggs, B.M., Mislove, A., Wilson, C.: Crlite: a scalable system for pushing all TLS revocations to all browsers. In: 2017 IEEE Symposium on Security and Privacy (SP), pp. 539–556. IEEE (2017)
20. Laurie, B.: Certificate transparency. Commun. ACM **57**(10), 40–46 (2014)
21. Lázaro, F., Raulefs, R., Wang, W., Clazzer, F., Plass, S.: VHF data exchange system (VDES): an enabling technology for maritime communications. CEAS Space **11**, 55–63 (2019). https://doi.org/10.1007/s12567-018-0214-8
22. Mitzenmacher, M.: Compressed bloom filters. IEEE/ACM Trans. Netw. **10**(5), 604–612 (2002)
23. Peiponen, H., Kukkonen, A.: Integrity monitoring and authentication for VDES pre-distributed public keys. IALA Committee Working Document. Input paper: ENAV18-11.10
24. Pettersen, Y.: The transport layer security (TLS) multiple certificate status request extension. RFC 6961, June 2013. https://www.ietf.org/rfc/rfc6961.txt
25. Pettersen, Y.: X.509v3 Transport layer security (TLS) feature extension. RFC 7633, October 2015. https://tools.ietf.org/html/rfc7633
26. Santesson, S., Myers, M., Ankney, R., Malpani, A., Galperin, S., Adams, C.: X.509 internet public key infrastructure online certificate status protocol - OCSP. RFC 6960, June 2013. https://tools.ietf.org/html/rfc6960
27. Smith, T., Dickinson, L., Seamons, K.: Let's revoke: scalable global certificate revocation. In: Proceedings 2020 Network and Distributed System Security Symposium, Internet Society, San Diego, CA (2020)
28. Wang, Q., Gao, D., Chen, D.: Certificate revocation schemes in vehicular networks: a survey. IEEE Access **8**, 26223–26234 (2020)
29. Wang, Z., Lin, J., Cai, Q., Wang, Q., Zha, D., Jing, J.: Blockchain-based certificate transparency and revocation transparency. IEEE Trans. Dependable Secure Comput. 1 (2020)
30. Wohlmacher, P.: Digital certificates: a survey of revocation methods. In: Proceedings of the 2000 ACM workshops on Multimedia, pp. 111–114 (2000)