2011 Third International Workshop on Security Measurements and Metrics

Quantifying the Effects of More Timely Certificate Revocation on Lightweight Mobile Devices

Sufatrio
Temasek Laboratories
National University of Singapore
5A, Engineering Drive 1, #09-02, Singapore 117411
Fax: +65 68726840
Email: tslsufa@nus.edu.sg

Roland H.C. Yap
School of Computing
National University of Singapore
13 Computing Drive, Singapore 117417
Fax: +65 67794580
Email: ryap@comp.nus.edu.sg

***Abstract*—Public Key Infrastructure (PKI) is a key infrastructure for secure communications and transactions on the Internet. We revisit the problem of timely certificate revocation and develop a performance analysis framework with more realistic assumptions of when certificates are revoked, a query model differentiating revoked and unrevoked certificates, and realistic cost factors. Our analysis is fine-grained and shows the impact of a revocation scheme on the computation, storage and bandwidth costs particularly on mobile devices as the verifiers. We apply our performance framework to analyze the following schemes: CRL, OCSP, CRS and CREV. Our analysis shows clearly the strengths and weaknesses of each scheme particularly for mobile lightweight verifiers under higher timeliness guarantees.**

I. Introduction

Public Key Infrastructure (PKI) is a critical infrastructure for securing communications and transactions over insecure public networks such as the Internet. Issued certificates, however, at times need to be revoked within their validity periods. This certificate revocation has long been recognized as a major challenge in PKI [1], [2]. In X.509 based PKI [3], [4], two standardized schemes commonly deployed are Certificate Revocation List (CRL) [4] and Online Certificate Status Protocol (OCSP) [5]. They however have their own respective drawbacks. CRL imposes a high bandwidth requirement on the *verifier* (the party relying on the certificate), and typically gives a low timeliness guarantee, i.e. *daily* CRL update (as suggested in [6]). OCSP offers a potentially real-time recency assurance, but puts a high (online) computational requirement on the Certification Authority (CA).

In this paper, we reassess the performance of revocation schemes by taking into account of two important emerging trends on the Internet. Firstly, the number of connected hosts continues to increase, of which a substantial growing proportion are mobile devices. A recent report [7] suggests that the number of smart phones shipped has already exceeded that of desktop PCs, and mobile devices are projected to become the dominant computing platform in the near future. Secondly, the growth in volume and value of Internet transactions increases the need for revocation services with *higher* timeliness guarantees (i.e. much less than a day). These trends put more demands on the scalability of revocation schemes as it can lead to unacceptable delays which may make revocation checking infeasible [8], [1]. Efficiency on power consumption, bandwidth and storage on mobile devices is also important as mobile devices have much more constraints than desktops PCs. Increased bandwidth for mobile devices may also translate into increased costs depending on the service provider plan, and therefore is important.

This paper presents an analytical framework for quantifying the performance of revocation schemes with a high timeliness guarantee, i.e. on the order of hours or minutes (rather than days as is assumed in previous work [9], [10], [11], [12]). More importantly, the framework incorporates two following observations on revocations, which we believe are important and differ from simplified assumptions taken in most previous work: (i) Most previous work [9], [10], [13], [14] assume that certificates are revoked at the midpoint of their issued lifetimes. Based on real-world CRL data, [11], [12] however found that most revocations occur during the early part of the certificate's lifetime; (ii) It is also common to assume (e.g. [10]) that the probability of a revoked certificate to be queried after its revocation remain constant. We argue however that the probability should instead be smaller because over time queries on the affected principal should be issued on the valid replacement certificate.

We show how to apply our framework to evaluate a variety of revocation schemes based on different approaches, and obtain some interesting results particularly pertinent to mobile verifiers. The size of CRL data is *substantially* higher than that derived in previous work ([9], [10], [13], [14]). This highlights that CRL-based schemes are expensive for mobile verifiers in that they require higher bandwidth and storage. We also find that the probability of a revoked certificate to be queried is quite low, e.g. less than 1% in the evaluation, which is much lower than $\sim$8% derived in [10]. This result is significant for hash-chaining based schemes

978-0-7695-4680-3/12 $26.00 © 2012 IEEE
DOI 10.1109/Metrisec.2011.17

(e.g. CRS [15]) since the mobile verifier is *almost certain* to perform repeated hash operations to prove the validity of a queried certificate. Our cost comparisons shows that CRL, OCSP and CRS reach different cost extremes in bandwidth, computation and storage costs respectively. We also show a new revocation scheme CREV [14] to be more scalable while being lightweight on the mobile verifiers.

The remainder of the paper is organized as follows. Section II surveys related work. Section III analyzes the revocation model of [11], [12] and highlights its limitations. Section IV elaborates our analysis framework. Section V evaluates several revocation schemes using our framework and discusses the results. We finally conclude in Section VI.

II. Related Work

We survey related work on cost analysis models for certificate revocation schemes, and highlight differences with our framework. A number of work propose analytical cost models for revocation schemes [9], [10], [11], [12], [13], [14]. With the exception of [11], [12], they are however not based on any empirical revocation data. Rather, they make the simplifying assumption that revocation always occurs halfway during the certificate's issued lifetime. Our framework, on the other hand, extends the realistic empirically-based revocation model proposed in [11], [12].

Zheng [9], which is widely cited, and also other work such as [11], [12], [13] derive revocation costs by fixing the revocation timeliness at one day. In contrast, we consider a revocation service with high timeliness guarantee on the order of minute(s) which allows for the evaluation of a higher revocation timeliness both on mobile verifier's costs and CA's scalability.

The work [10] proposed a more detailed framework than [9], but is still based on the midpoint revocation assumption. It derived the probabilities of whether the status of a queried certificate is valid and revoked. However it was assumed that query rates on certificates after their revocation remain constant. In contrast, we employ what is arguably a more realistic assumption – the number of queries on revoked certificates tend to decrease over time. Lastly, we incorporate standardized sizes for certificate revocation message, also referred to as Certificate Status Information (CSI), as analyzed in [16] to derive realistic message costs for revocation.

Recently, a new revocation scheme CREV was proposed in [14], which we also evaluate here using our developed more realistic framework.

III. Empirical Revocation Model of [11], [12]

Since our framework refines that proposed in [11], [12], we summarize their empirical revocation model in Sec. III-A, and analyze its limitations in Sec. III-B.

A. Overview of the Empirical Revocation Model

The aim in [11], [12] is to find CA's strategies in releasing CRLs so as to reduce its operational cost; whereas here we want to measure the tradeoffs and costs of revocation on all involved parties. [11], [12] thus derive an estimate of the number of new revocations between two successive CRL generations as well as the size of CRL on a particular day.

Based on empirical data of CRLs collected from VeriSign, they found that most certificate revocations occur during the early part of the certificate's lifetime. In addition, the percentage of revocations decreases as time elapses. This finding thus invalidates the assumption on CRL size made by most other work [9], [10], [13], [14].

The work [11], [12] model certificate revocations over time using a Probability Density Function (PDF). Suppose that there are α certificates (with uniform issued age of β days) issued at time X. From time X to $X+\beta$, on average αb of the certificates will be revoked. The basic time unit between two CRL releases (Δt) is assumed to be of one day. The function $R(t)$ is defined to represent *the revoked percentage*, i.e. the ratio of the number of revocations that occur within the time interval $[t, t+\Delta t]$, where $X<t<X+\beta$, to the total number of revocations within the time interval $[X, X+\beta]$. The PDF giving the distribution of revocation of issued certificates at a particular time is:

$$R(t) = ke^{-kt} \tag{1}$$

where t is the time (after the certificate issuance at X), and the value of $k = 0.26$ has a good fit to the actual revocation data.

Based on $R(t)$, they derive an estimate of the CRL size on a particular day. Three certification scenarios are considered, namely: certificates issued at different times with the same issued age β; certificates issued at different times with different issued ages; and a general case where the number of certificates generated follows a Poisson distribution. They derive an analytical model for the first and the second scenarios, but use simulation for the third. In our framework, we only consider the first scenario, however it is straightforward to extend the framework to analyze the second scenario.

The number of *new revocation requests* is derived as follows. Suppose that $v \in \mathbb{N}$ is any time in $(0, \beta]$. As mentioned, [11], [12] assume Δt = 1 day. Similarly, the time interval between two certificate generations (ΔX) is also assumed to be one day. Function $f(v)$ in [11] is defined as the number of *new* certificate revocations between day v and day $v+\Delta t$ from all of the valid generations:

$$\begin{aligned} f(v) = \alpha bR(v) + \alpha bR(v - \Delta t) + \alpha bR(v - 2\Delta t) \\ + \ldots + \alpha bR(v - (n-1)\Delta t) \end{aligned} \tag{2}$$

where n is the number of certificate generations within time period β days, i.e. $n=\lceil \frac{\beta}{\Delta t} \rceil$. They find that the *steady-state condition*, where certificate issuances are balanced out by

certificate expiration, is achieved when $v \in [\beta, +\infty)$ with:

$$\begin{aligned} f(v) &= \alpha b R(1) + \alpha b R(2) + \ldots + \alpha b R(\beta) \\ &= \alpha b k e^{-k} \frac{1 - e^{-\beta k}}{1 - e^{-k}} \end{aligned} \tag{3}$$

The *size of CRL data* is derived as follows. Let $F(v)$ be the valid cumulative number of revocations from time 1 to v. It is also the size of the CRL on that day. For v in $[\beta, +\infty)$:

$$\begin{aligned} F(v) = F(\beta) &= \sum_{t=1}^{\beta} f(t) \\ &= \frac{\alpha b k e^{-k}}{1 - e^{-k}} \left[\beta - \frac{e^{-k}}{1 - e^{-k}} \left(1 - e^{-\beta k}\right) \right] \end{aligned} \tag{4}$$

That is, after the system reaches the steady-state condition when $v = \beta$, the CRL data size remains constant. This is the case since the addition of revoked certificates into the CRL is balanced out by the removal of revoked entries reaching their issued lifetimes (expiration). In Sec. IV-A, we give a more accurate derivations of $f(v)$ and $F(v)$.

B. Limitations of the Revocation Model of [11], [12]

The CRL size calculation derived in [11], [12] assumes that Δt (the time interval between two successive CRL releases) = ΔX (the time interval between two certificate generations) = 1 day. As we want higher timeliness, we need to obtain an improved CRL size estimate where ΔX and Δt are in minutes, and Δt is decoupled from ΔX, i.e. $\Delta t \neq \Delta X$.

[11], [12] also approximate the PDF when defining $f(v)$ as in Eqs. 2 and 3 by their uses of summation operations. When ΔX and Δt are much less than one day, the approximation in $f(v)$ can give invalid results as the number of revoked entries becomes larger than the total revoked certificates (see Sec. V-C for an example).

IV. A Quantitative Analysis Framework for Certificate Revocation

We propose a quantitative performance-analysis framework which employs the empirical based revocation requests from [11], [12], but updated for a finer time granularity as well as incorporating other more realistic features and assumptions.

The framework is intended to measure the overheads of revocation schemes with a single CA certification system during its *steady-state condition*, i.e. when certificate expiration is balanced by the new certificates added. This steady-state assumption is commonly adopted by many papers [11], [12], [10], [9], which allows us to focus on the stable behavior of the revocation schemes and omit possible variable external factors.

Some other assumptions commonly made when analyzing revocation schemes under steady-state (also assumed in [11], [12], [10], [9]), which we will also make, are the following. The total number of principals and the corresponding valid certificates (N) is constant. Certificates have the same lifetime (β days), where b percent of N certificates are revoked within that lifetime. The time interval between two successive CRL releases determines the timeliness for the CRL scheme and is denoted by Δt (in minutes). The time interval between two successive certificate generations (ΔX) is constant (in minutes), and is assumed to be a multiple of the timeliness (i.e. Δt). Certificate issuance takes place at a constant rate (of $\frac{N \cdot \Delta X}{\beta \cdot 1440}$ per issuance). When a certificate is revoked, a new valid certificate is issued immediately to replace it. Table I summarizes the notation and parameter values selected for the evaluation in Sec. V.

A. Refining the Empirical Revocation Model of [11], [12]

We first describe our revised method for $\Delta X = \Delta t = 1$ day, before giving a finer-grained model.

1) A More Accurate Method for CRL Size Estimate: The model in [11], [12] is built around *daily* ΔX and Δt. As such, there are $\alpha = \frac{N}{\beta}$ certificates issued on day X, which are valid between day X and day $X+\beta$. Recall again that for certificates issued at time X, $R(t) = k\, e^{-kt}$, where k =0.26 and t is the time, represents the revoked percentage.

[11], [12] calculate the CRL size estimate using Eqs. 2–4. The derivation employs $f(v)$ in [11] to obtain the number of new certificate revocations between day v and day $v+\Delta t$. However, they discretize the PDF, which gives only an approximation (see Eqs. 2–3). We instead need to *integrate* the PDF because the approximation becomes invalid when ΔX and Δt is much less than one day. A subtle point is that in the steady-state calculation (Eq. 3), $f(v)$ actually measures the number of revocations for days $[1, \beta+1]$ rather than $[0, \beta]$ which we employ.

We reformulate $F(v)$ where $\Delta X = \Delta t = 1$ day by first defining $g(v)$ to replace $f(v)$. The function $g(v)$ is defined as the number of new revocations between day $v - \Delta t$ and day v from all valid generations. When $v \in [0, \beta - 1]$, we have:

$$\begin{aligned} g(v) &= \sum_{i=1}^{v} \alpha b \int_{i-1}^{i} R(t)\, dt = \alpha b \int_{0}^{v} k e^{-kt}\, dt \\ &= \alpha b \left[-e^{-kt} \right]_0^v = \alpha b \left(1 - e^{-kv}\right) \end{aligned} \tag{5}$$

When $v \in [\beta, +\infty)$, where the steady-state condition takes place:

$$g(\beta) = \alpha b \int_{0}^{\beta} k e^{-kt}\, dt = \alpha b \left(1 - e^{-k\beta}\right) \tag{6}$$

$F(v)$, representing the cumulative number of certificate revocations from day 0 to day v, with $v \in [1, \beta - 1]$ becomes:

$$F(v) = \sum_{i=1}^{v} g(i) = \alpha b \left[v - e^{-k} \frac{1 - e^{-kv}}{1 - e^{-k}} \right] \tag{7}$$

Symbol	Description	Unit	Scenario Value(s)	
Parameters for Certification System:				
N	Number of valid (non-revoked) and not-yet-expired certificates	-	100,000	
β	Issued lifetime of a certificate	days	365	
b	Percentage of certificates revoked within β	-	0.1 = 10%	
M	Number of minutes in a day	-	1,440	
ΔX	Time interval between two successive certificate generations	mins	1,440 (1 day)	
Δt	Time interval between two successive CRL releases	mins	60	10
γ	Constant factor of $\Delta X/M$	-	$\Delta X/M$	
λ	Constant factor of $\Delta t/M$	-	$\Delta t/M$	
$d = \Delta t$	Time interval for periodic hash-token release in CRS and CREV-II	mins	60	10
t_{CREV_I}, t_{CREV_II}	Session lifetime in CREV-I and CREV-II	mins	720	180
k	Parameter for realistic certificate revocation PDF	-	0.26 (Ref. [11], [12])	
V	Number of verifiers	-	30,000,000	
Q_{V_daily}	Average daily queries needed by a verifier	-	30	
$Q_{V\_issued\_daily}$	Average daily queries *issued* by a verifier to CA/CMAE	-	$min(\frac{M}{\Delta t}, Q_{V\_daily})$	
Q_{daily}	Total average daily queries received by CA/CMAE	-	$V \cdot Q_{V\_issued\_daily}$	
$q_{per\_cert\_daily}$	Average daily queries on a certificate	-	Q_{daily}/N	
Costs of Basic Cryptographic Operations (for Desktop Computer):				
C_{sign}	Cost of a digital signature (RSA-1024) generation	ms	1.48	
C_{verify}	Cost of a digital signature (RSA-1024) verification	ms	0.07	
C_{hash}	Cost of computing a hash (SHA-1)	μs	0.40	
Costs of Basic Cryptographic Operations (for Mobile Device):				
C_{sign}	Cost of a digital signature (CRT RSA-1024) generation [17]	ms	558	
C_{verify}	Cost of a digital signature (RSA-1024) verification [17]	ms	29	
C_{hash}	(Projected) cost of computing a hash (SHA-1)	μs	160	

Table I
NOTATION AND PARAMETER VALUES USED IN THE PERFORMANCE ANALYSIS.

For $v \in [\beta, +\infty)$ which is the steady-state condition, the number of entries in the CRL ($Rev\_Entries$) is:

$$Rev\_Entries = F(\beta{-}1) \tag{8}$$

Notice that in Eq. 8, we use $F(\beta{-}1)$ instead of $F(\beta)$ as in Eq. 4 because there are only $\beta{-}1$ valid certificate generations, and not β. The generation on day v has no revoked certificates yet, and the generation on day $v{-}\beta$ is expired on that day, hence leaving only $\beta{-}1$ generations from day $v{-}1$ to $v{-}\beta+1$.

In fact, we can employ a *simpler* and *more elegant* way of calculating the CRL size on day v, which is described as follows. First, we define $h(i)$ as the total number of (non-expired) revoked certificates from certificates generated in the i generation(s) *prior to* the current generation:

$$h(i) = \alpha b \int_0^i R(t)\, dt \tag{9}$$

The number of entries in CRL during the steady-state condition (with $\beta{-}1$ certificate generations counted) is:

$$\begin{aligned} Rev\_Entries &= \sum_{i=1}^{\beta-1} h(i) = \sum_{i=1}^{\beta-1} \alpha b \int_0^i R(t)\, dt \\ &= \alpha b \left[(\beta-1) - e^{-k} \frac{1 - e^{-k(\beta-1)}}{1 - e^{-k}} \right] \end{aligned} \tag{10}$$

yielding the same result as Eq. 8. In our generalization below, we will make use of $h(i)$.

2) A Finer Granularity of Certificate Generation (ΔX) and CRL Issuance (Δt) in Minutes: We want to have a finer grained revocation timeliness guarantee on the order of minutes. Hence, ΔX and Δt are now in minutes. Yet, some aspects of the certificate modeling, such as the certificate lifetime β and the defined function $R(t)$ as in [11], [12], are in days. As such, we need some adjustments to scale the model for a finer time granularity.

Let $M = 1440$ denote the number of minutes in a day. We define a constant ratio $\lambda = \frac{\Delta t}{M}$, which basically expresses Δt in day unit. Additionally, let us also define $\gamma = \frac{\Delta X}{M}$. If $\Delta t = \Delta X$, then $\lambda = \gamma$. The number of certificate issued per generation (α') is now:

$$\alpha' = \frac{N\gamma}{\beta} = \alpha\gamma \tag{11}$$

We generalize the previously defined $h(v)$ into $h'(v)$ in order to take into account the scale for finer time granularity:

$$\begin{aligned} h'(i) &= \alpha' b \int_0^i R(t)\, dt = \alpha' b \int_0^i k e^{-kt} dt \\ &= \frac{N\gamma}{\beta}\, b \left(1 - e^{-ki}\right) \end{aligned} \tag{12}$$

In a certification where $\Delta X = \Delta t$, the number of revoked

entries during its steady-state condition is:

$$\begin{aligned} Rev\_Entries' &= \sum_{i=1}^{\eta} h'(\gamma\, i) = \sum_{i=1}^{\eta} \alpha' b \int_0^{\gamma i} R(t)\, dt \\ &= \frac{N\gamma}{\beta}\, b \left[\left(\frac{\beta}{\gamma} - 1 \right) - e^{-k\gamma}\, \frac{1 - e^{-k(\beta-\gamma)}}{1 - e^{-k\gamma}} \right] \end{aligned} \tag{13}$$

where $\eta = \frac{\beta}{\gamma} - 1$ denotes the number of counted certificate generations.

When $\Delta X \neq \Delta t$, we assume ΔX to be a multiple of Δt, i.e. $\Delta X = c\,\Delta t$ for some $c > 1$, the number of entries in the CRL periodically varies depending on the how far a CRL issuance occurs from the last certificate generation. When a CRL release takes place at time $j \cdot \Delta t$, where $1 \leq j \leq c-1$, after the last certificate release, the number of entries in the CRL is:

$$\begin{aligned} Rev\_Entries'(j) &= \sum_{i=1}^{\frac{\beta}{\gamma}} h'\big((i-1)\gamma + j\lambda\big) \\ &= \frac{N\gamma}{\beta}\, b \left[\frac{\beta}{\gamma} \;-\; e^{-k\lambda j}\, \frac{1 - e^{-k\beta}}{1 - e^{-k\gamma}} \right] \end{aligned} \tag{14}$$

When a CRL release coincides with a certificate issuance, the number of entries in the CRL is:

$$Rev\_Entries'(c) = \sum_{i=1}^{\frac{\beta}{\gamma}-1} h'\,(\gamma\, i) \tag{15}$$

which is exactly the same as Eq. 13.

It can be shown from Eqs. 14 and 15 that the maximum number of entries in the CRL happens when $j = c - 1$. To obtain the average number of entries in the CRL, we simply compute:

$$\overline{Rev\_Entries'} = \frac{\sum_{j=1}^{c} Rev\_Entries'(j)}{c} \tag{16}$$

B. More Realistic Query Models for Revoked Certificates

In hash-chaining based revocation schemes, like CRS, the verifier's cost in validating a certificate may depend on whether the certificate is valid or revoked. Thus, the framework should incorporate some realistic assumptions for the probability of a revoked (but not expired) certificate to be queried by the verifiers.

We would expect that once a certificate has been revoked, the likelihood that the revoked certificate will be queried by a verifier should *decrease* over time. One reason for this would be that the revoked certificate could be replaced. For example, a new certificate will be used for SSL in a secure web server with a recently revoked certificate, or mobile code which has been previously signed by a revoked certificate is replaced by mobile code signed using a new certificate. Some verifiers may still have a stale reference to the revoked certificate, but the numbers of queries issued by those verifiers would be expected to decrease over time.

Let us consider a principal X that uses certificate $Cert_X$ for the verifiers to validate. We denote t_{issued_X} as the time when $Cert_X$ is issued. When certificate $Cert_X$ is revoked at time t_{rev_X}, it is replaced by a new certificate $Cert'_X$. The function $S(t)$ is defined as the *probability* that $Cert_X$ will still be used in a query involving X within a time interval $[(t-1)\Delta t,\ t\Delta t\,]$ (in minutes) after t_{rev_X}. As t increases, we assume a reasonable model is given by a *exponential decay* function, although other functions could be used. Thus, for $0 < t < \beta - (t_{rev\_X} - t_{issued\_X})$, we have:

$$S(t) = a\, e^{-r\,\lambda\, t} \tag{17}$$

A constant factor $0 < a \leq 1$ is employed to capture the effects that some revoked certificates are actually replacements to previously replaced revoked certificates. Hence, they were not yet fully referred to when their revocations occur.

We also take into account the fact that certificates are used for different purposes, such as for online transactions and code/document signing. As such, one should also have different *query models* on revoked certificates to suit these different purposes. For simplicity, in our evaluation, we use a common query model which is Eq. 17 parameterized by a common a =0.95 and two different values of r. The value of a is set to 0.95 based on the fact that revocation does not affect 90% of the certificates, and that the chances that some of the revoked certificates (the other 10%) represent previous certificate replacements are also not so high.

We now distinguish between:

- Model 1 (online-transaction certificates): This applies to certificates used for online transactions where a revoked certificate will quickly cease to be referred to. Here we use $S(t)$ with $r = 0.95$, which is denoted by $S_1(t)$ in later sections.
- Model 2 (code/document-signing certificates): This applies to certificates used to provide non-online services, such as the signing of documents/software that may be distributed in an off-line manner (e.g. using a physical media). The only difference is that a revoked certificate is expected to be referred to for a longer time period following its revocation. We choose a value of r=0.01 in the evaluation of revocation schemes and call this model $S_2(t)$.

C. Probabilities of Querying Revoked and Valid Certificates

Our objective here is to derive the probability a query from a verifier is issued on a valid certificate (Pr_{valid}) and conversely, a revoked one (Pr_{rev}). We derive the probabilities as follows.

First, from Table I, recall that the total number of daily queries issued by all verifiers is: $Q_{daily} = V \cdot Q_{V\_issued\_daily}$. Since we assume that queries on all the certificates in the system are uniformly distributed, we have the

number of queries per certificate in a day: $q_{per\_cert\_daily} = Q_{daily}/N$.

Now, we derive $Q_{rev\_\beta\_days}$ which is the total number of queries on *all revoked* certificates (until their expirations) *throughout* β days under the steady-state condition. In general, there are n query models on the different types of revoked certificates. For each model m, we define $Q_{rev\_\beta\_days}(m)$ as follows:

$$Q_{rev\_\beta\_days}(m) = \sum_{i=1}^{\frac{\beta}{\lambda}-1} \Bigg[\Big(N_m \cdot b \cdot \int_{\lambda(i-1)}^{\lambda i} R(t)\, dt \Big) \cdot \Big(\sum_{j=1}^{\frac{\beta}{\lambda}-i} S_m(j) \cdot q_{per\_interval} \Big) \Bigg] \tag{18}$$

where: N_m denotes the number of certificates with query model m; $\lambda = \frac{\Delta t}{M}$; and $q_{per\_interval} = \lambda \cdot q_{per\_cert\_daily}$ denotes the number of queries on a certificate within the time interval Δt minutes. This Eq. 18 basically counts, for every Δt, the number of certificates belonging to query model m which are just revoked, and subsequently the number of queries these certificates will be subject to throughout their remaining lifetime (according to the pertinent query model).

To obtain the overall $Q_{rev\_\beta\_days}$, we derive:

$$Q_{rev\_\beta\_days} = \sum_{m=1}^{n} Q_{rev\_\beta\_days}(m) \tag{19}$$

Now, the total number of queries on *all* non-expired certificates (both valid and revoked ones) for β days is $Q_{all\_\beta\_days} = Q_{daily} \cdot \beta$. Hence, the total number of queries on *all valid* certificates throughout β days is: $Q_{valid\_\beta\_days} = Q_{all\_\beta\_days} - Q_{rev\_\beta\_days}$. Finally, the probabilities of whether a status query is issued on a valid certificate (Pr_{valid}) and a revoked certificate (Pr_{rev}) are:

$$Pr_{rev} = \frac{Q_{rev\_\beta\_days}}{Q_{all\_\beta\_days}} \qquad Pr_{valid} = \frac{Q_{valid\_\beta\_days}}{Q_{all\_\beta\_days}} \tag{20}$$

D. Taking Realistic CSI Message Sizes into Account

Many previous work [9], [10] take a simplified (or minimalistic) approach with respect to the CSI message sizes of standardized revocation schemes. In practice, however, the CSI also include various auxiliary information. To achieve a realistic analysis, we follow the CSI message sizes in [16], which utilize a BER viewer for all ASN.1 standardized data structures. Furthermore, we employ SHA-1 for the hash algorithm and SHA-1 with RSA for the signature algorithm in the evaluation in Sec. V.

V. Application of the Framework on Revocation Schemes

We now apply our framework to several certificate revocation schemes to evaluate their performance with higher timeliness guarantees. Below, we refer to *Certificate Management Ancillary Entity* (CMAE) as a designated repository/directory from which a verifier may obtain revocation information issued by the CA.

A. Performance Comparison Metrics

We use the following notation to denote various costs incurred in a particular revocation scheme: Ovh_A= computation time needed by entity A (in seconds), Bw_{A-B} = network bandwidth needed from entity A to B (in MB), and $Stor_A$ = storage needed on A (in MB). The entities involved are: CA indicating the CA, Ver indicating a verifier, $CMAE$ indicating a repository, and $EVCP$ indicating a principal's server used in CREV [14].

To compare different schemes, we use the following metrics where the subscripts denote the entity:

1) Certificate creation cost: Ovh_{CA}.
2) Update costs: Ovh_{CA}, Ovh_{CMAE} (Ovh_{EVCP}) and $Bw_{CA-CMAE}$ ($Bw_{CA-EVCP}$).
3) Query costs: Ovh_{CA}, Ovh_{CMAE} (Ovh_{EVCP}), $Bw_{CMAE-Ver}$ ($Bw_{EVCP-Ver}$) and Ovh_{Ver}.
4) Storage requirement at one point in time (during the steady-state condition): $Stor_{CA}$ and $Stor_{CMAE}$ ($Stor_{EVCP}$).
5) Timeliness: the revocation latency, which also represents the *window of vulnerability* of the revocation scheme. Note that the lower the latency is, the higher the timeliness guarantee is provided.

For metrics (1) to (3), we measure the total *cost per day*. We use $D\_\langle Cost \rangle$ to denote *the daily cost* of a cost metric. In order to have a more compact notation, we abuse the notation slightly and write $\forall X$ for all instances of entity X, and $\exists X$ for a single instance of X. Thus, for example, $D\_Bw_{CMAE-\exists Ver}$ denotes the daily bandwidth cost between the CMAE and a single verifier, whereas $D\_Bw_{CMAE-\forall Ver}$ is the daily bandwidth needed by the CMAE to all the verifiers.

B. Performance Models for: CRL, OCSP, CRS, CREV

We evaluate the following revocation schemes using our analysis framework: CRL (with a CMAE), OCSP (with the CA as a Responder), CRS (with a CMAE) and CREV. When measuring network costs, we do not consider the cost due to the underlying network transfer mechanism(s), e.g. HTTP, TCP or IP. However, we use realistic message sizes (for the CSI) from [18] and L_M is used to denote the length of the message portion of M.

In our evaluation of revocation schemes, we employ a revocation timeliness of 1 hour and also 10 minutes. The number of daily queries *needed* by a verifier (Q_{V_daily}) is set at a moderate number, 30 queries (see Table I). We assume that the queries from a verifier are issued throughout the day with uniformly distributed time intervals. While this is a simplifying assumption, it is reasonable as one could

expect that the aggregate behavior across a large number of verifiers can look more uniform even if individual verifiers are less uniform. Since a verifier can cache the revocation information as long as it is still valid, the number of daily queries *actually issued* by a verifier is: $Q_{V\_issued\_daily} = min(\frac{M}{\Delta t}, Q_{V\_daily})$. The total number of daily queries issued by *all* verifiers is therefore: $Q_{daily} = V \cdot Q_{V\_issued\_daily}$. In all the schemes below, the *daily* cost of certificate creation is: $D\_Ovh_{CA} = (\frac{N}{\beta} + \frac{N \cdot b}{\beta}) \cdot C_{sign}$.

1) CRL (with a CMAE): In our framework, the average number of revoked (but non-expired) certificates where $\Delta X = c\,\Delta t$ is $\overline{Rev\_Entries'}$ (see Eq. 16). The CRL size is: $L_{CRL} = L_{CRL\_fields} + \lfloor \overline{Rev\_Entries'} \rfloor \cdot L_{CRL\_entry}$ where $L_{CRL\_fields} = 400$ bytes is the message length of the CRL header and signature, and $L_{CRL\_entry} = 39$ bytes is the length of each entry in the CRL data [16]. With $U = \frac{M}{\Delta t}$ as the total number of CRL updates in a day, the daily update costs are: $D\_Bw_{CA-CMAE} = U \cdot L_{CRL}$, $D\_Ovh_{CA} = U \cdot C_{sign}$, and $D\_Ovh_{CMAE} = U \cdot C_{verify}$.

The daily query costs of CRL are: $D\_Bw_{CMAE-\forall Ver} = Q_{daily} \cdot L_{CRL}$, $D\_Bw_{CMAE-\exists Ver} = Q_{V\_issued\_daily} \cdot L_{CRL}$, $D\_Ovh_{CA} = 0$, $D\_Ovh_{CMAE} = 0$, and $D\_Ovh_{Ver} = Q_{V\_issued\_daily} \cdot C_{verify}$

The storage requirements are: $Stor_{CA} = Stor_{CMAE} = L_{CRL}$. Finally, the revocation latency is δ minutes.

2) OCSP (with CA as Responder): We analyze the overheads of OCSP where the CA functions as the OCSP Responder, and that a nonce is used to bind an OCSP Response with the corresponding Request. There is no update cost between the CA and CMAE, since no CMAE is involved. With L_{OCSP_Resp}= 459 bytes as the length of OCSP Response [16], the daily query costs (due to status reply) are: $D\_Bw_{CA-\forall Ver} = Q_{daily} \cdot L_{OCSP\_Resp}$, $D\_Bw_{CA-\exists Ver} = Q_{V\_issued\_daily} \cdot L_{OCSP\_Resp}$, $D\_Ovh_{CA} = Q_{daily} \cdot C_{sign}$, and $D\_Ovh_{Ver} = Q_{V\_issued\_daily} \cdot C_{verify}$ The storage requirement is: $Stor_{CA} = 0$. The revocation latency of OCSP scheme can be close to zero when desired.

3) CRS (with a CMAE): We set the time interval for periodic hash-token release $d = \Delta t$ minute(s). The length of the hash chain is: $\ell = \frac{\beta M}{\Delta t} - 1$. Here, we assume that the CA stores the whole hash chain for all the valid certificates in its storage. An amortization technique such as [19] can be used to reduce its storage requirements, but at the cost of additional online processing for the CA.

We use $L_{CRS\_fields} = 161$ bytes to denote the length of the CA's timestamp and signature, and $L_{S\_No} = 7$ bytes to denote the length of a certificate's serial number [16]. The bandwidth cost for a *single* update between CA and CMAE is $Bw_{CA-CMAE} = L_{CRS\_fields} + (N + \lfloor \overline{Rev\_Entries'} \rfloor) \cdot (L_{S\_No} + L_{hash})$. With $U = \frac{M}{\Delta t}$, the total daily update costs become: $D\_Bw_{CA-CMAE} = U \cdot Bw_{CA-CMAE}$, $D\_Ovh_{CA} = U \cdot C_{sign}$, and $D\_Ovh_{CMAE} = U \cdot C_{verify}$

For queries, the overheads incurred on the verifier depends on the following conditions. If the certificate is revoked, then $Ovh_{Verifier\_rev} = C_{hash}$. If the certificate is valid, the overhead varies according to the number of hash operations required. The best case occurs when the first token is checked, or when the verifier has previously checked the last token, thus $Ovh_{Verifier\_valid\_best} = C_{hash}$. The worst case happens when the verifier needs to perform ℓ hash operations, resulting in $Ovh_{Verifier\_valid\_worst} = \ell \cdot C_{hash}$. The average number of hash computations is $\frac{\ell+1}{2}$, hence $Ovh_{Verifier\_valid\_avg} = \frac{\ell+1}{2} \cdot C_{hash}$. The verifier's average query cost which encompasses both the revoked and the valid cases is therefore $Ovh_{Verifier\_Avg} = Pr_{rev} \cdot Ovh_{Verifier\_rev} + Pr_{valid} \cdot Ovh_{Verifier\_valid\_avg}$, with Pr_{rev} and Pr_{valid} defined in Equation (20).

The daily query costs are: $D\_Bw_{CMAE-\forall Ver} = Q_{daily} \cdot L_{hash}$, $D\_Bw_{CMAE-\exists Ver} = Q_{V\_issued\_daily} \cdot L_{hash}$, $D\_Ovh_{CA} = 0$, $D\_Ovh_{CMAE} = 0$, $D\_Ovh_{Ver\_Avg} = Q_{V\_issued\_daily} \cdot Ovh_{Ver\_Avg}$.

For the storage costs, note that the CA can remove the sub-chains it has released. Thus, the storage requirements are:

$$Stor_{CA} = \sum_{i=1}^{\ell} \frac{N\lambda}{\beta} \cdot (i+1) \cdot L_{hash} + \lfloor \overline{Rev\_Entries'} \rfloor \cdot L_{hash}$$
$$= \frac{N\lambda}{\beta} \cdot \frac{\ell^2 + 3\ell}{2} \cdot L_{hash} + \lfloor \overline{Rev\_Entries'} \rfloor \cdot L_{hash}$$

and $Stor_{CMAE} = Bw_{CA-CMAE}$. Finally, the revocation latency of CRS is $d = \Delta t$ minutes.

4) CREV-I: In CREV scheme, each principal (EVCP) has its own server to host/cache the relevant CSIs issued by the CA. The verifiers validating a principal obtains the CSIs from the EVCP. This may significantly relieve the burden of the centralized CA/CMAE since, as can be seen in Table I, it is generally assumed that the number of verifiers (V) is much greater than the number of principals (N).

The CA sends an OCSP Response message every $d_{CA} = \Delta t$ minutes. In a day, each EVCP thus performs $S = \frac{M}{t_{CREV\_I}}$ session establishments, and receives $U = \frac{M}{\Delta t}$ OCSP Response messages. We use $L_{CREV\_I\_Msgs} = 1{,}577$ bytes to denote the length of all messages in a session establishment, L_{OCSP_Resp} to denote the length of an OCSP Response message, and L_T to denote the length of the timestamp or a CA's nonce.

The total daily update costs of CREV-I are: $D\_Bw_{CA-\forall EVCP} = N \cdot (S \cdot L_{CREV\_I\_Msgs} + U \cdot L_{OCSP\_Resp})$, $D\_Bw_{CA-\exists EVCP} = S \cdot L_{CREV\_I\_Msgs} + U \cdot L_{OCSP\_Resp}$, $D\_Ovh_{CA} = N \cdot S \cdot (2 \cdot C_{verify} + C_{sign}) + N \cdot U \cdot C_{sign}$, and $D\_Ovh_{EVCP} = S \cdot (2 \cdot C_{sign} + C_{verify}) + U \cdot C_{verify}$.

The total daily query costs are: $D\_Bw_{EVCP-\forall Ver} = Q_{per\_cert\_daily} \cdot L_{OCSP\_Resp}$, $D\_Bw_{EVCP-\exists Ver} = Q_{V\_issued\_daily} \cdot L_{OCSP\_Resp}$, with $D\_Ovh_{CA} = 0$, $D\_Ovh_{EVCP} = 0$, and $D\_Ovh_{Ver} = Q_{V\_issued\_daily} \cdot$

C_{verify}.

The storage requirements are: $Stor_{CA} = 0$ and $Stor_{EVCP} = L_T + L_{OCSP\_Resp}$.

5) CREV-II: We set the hash-chain update interval in CREV-II (d) to be Δt minutes. Each EVCP thus performs $S = \frac{M}{t_{CREV\_II}}$ session establishments daily, and receives $U = S \cdot \ell_{CREV\_II} = \frac{M}{\Delta t} - S$ hash-token updates daily. We use $L_{CREV\_II\_Reply} = 605$ bytes to denote the length of $Session\,Reply$ message, and $L_{CREV\_II\_Msgs} = 1{,}615$ bytes to denote the length of all messages in a session establishment of CREV-II.

The total daily update costs are: $D\_Bw_{CA-\forall EVCP} = N \cdot (S \cdot L_{CREV\_II\_Msgs} + U \cdot L_{hash})$, $D\_Bw_{CA-\exists EVCP} = S \cdot L_{CREV\_II\_Msgs} + U \cdot L_{hash}$, $D\_Ovh_{CA} = N \cdot S \cdot (2 \cdot C_{verify} + C_{sign})$, and $D\_Ovh_{EVCP} = S \cdot (2 \cdot C_{sign} + C_{verify}) + U \cdot C_{hash}$.

The verifier's average query cost which encompasses both the revoked and the valid cases is: $Ovh_{Ver\_Avg} = Pr_{rev} \cdot Ovh_{Ver\_rev} + Pr_{valid} \cdot Ovh_{Ver\_valid\_avg}$, with $Ovh_{Ver\_rev} = C_{hash}$ and $Ovh_{Ver\_valid\_avg} = \frac{\ell_{CREV\_II}+1}{2} \cdot C_{hash} + C_{verify}$.

The total daily costs are: $D\_Bw_{EVCP-\forall Ver} = Q_{per\_cert\_daily} \cdot (L_{hash} + L_{CREV\_II\_Reply})$, $D\_Bw_{EVCP-\exists Ver} = Q_{V\_issued\_daily} \cdot (L_{hash} + L_{CREV\_II\_Reply})$, $D\_Ovh_{CA} = 0$, $D\_Ovh_{EVCP} = 0$, and $D\_Ovh_{Ver} = Q_{V\_issued\_daily} \cdot Ovh_{Ver\_Avg}$.

The storage requirements are: $Stor_{CA} = \frac{N}{\ell_{CREV\_II}} \cdot \frac{\ell^2_{CREV\_II}+3\ell_{CREV\_II}}{2} \cdot L_{hash}$, and $Stor_{EVCP} = L_T + L_{CREV\_II\_Reply} + L_{hash}$.

C. Evaluation Scenarios and Results

We conducted our evaluation on a certification system with 100,000 certificates, and a 10% revocation rate (as suggested in [20]). The other parameter values for two evaluation scenarios (with various timeliness guarantees) are given in Table I. Our main objective is to have a quantitative comparison of the costs incurred by the various schemes under the same evaluation scenarios. For the calculation of Pr_{rev} and Pr_{valid}, we assume that 80% of the certificates follow $S_1(t)$ query model on revoked online-transaction certificates and the other 20% observe $S_2(t)$ for code/document certificates.

We choose SHA-1 for the hash function and signatures are created with RSA with a 1024-bit modulus. The overheads of basic cryptographic operations for desktop PCs are shown in Table I based on the Crypto++ 5.6.0 benchmarks (http://www.cryptopp.com/benchmarks.html) on an Intel Core-2 PC with 1.83 GHz CPU running Windows Vista. As for the cryptographic costs on the mobile devices, we use benchmarks from [17], which reported the costs for RSA-1024 encryption and decryption using CRT-RSA on a HTC Touch Dual from 2007 with Qualcomm MSM 7200 400 MHz processor and 128-MB of RAM. These results indicates that the particular mobile phone is about 400 times slower than the PC used in Crypto++ benchmarks. As they did not report the timings for computing SHA-1, we assume the costs to be also the same ratio compared to the Crypto++ benchmarks.

Performance Factors	Revocation Latency		
	1 day	1 hour	10 mins
$\overline{Rev\_Entries'}$	9880.33	9894.05	9894.53
$CRL\,Size$ (KB)	376.68	377.21	377.21
Pr_{rev}	0.005170	0.005279	0.005283
Pr_{valid}	0.994830	0.994721	0.994717

Table II
VALUES OF SOME PERFORMANCE FACTORS UNDER VARIOUS REVOCATION LATENCIES.

The expected values of $\overline{Rev\_Entries'}$, CRL data size, Pr_{rev} and Pr_{valid}) from the analytical model in steady state under different timeliness guarantees are shown in Table II. We remark that the values only have some small variation with different timeliness due to averaging behavior. What is more significant is the estimates differ significantly from other analytical models as discussed below.

As shown in Table II, under three different revocation latencies (with Δt from 1 day to 10 mins), there are 9,880–9,894 entries in the CRL which represent $\sim 98\%$ of the total revoked certificates. The number of entries is significantly higher than using prior simplified model which were based on an assumption that revoked entries are kept in the CRL for $\frac{\beta}{2}$ days [9], [10] (only 5,000 entries in the CRL). So, even with N=100,000 and $\Delta t = 1$ day, the *difference* in CRL data size calculation is 190,720 bytes, which increases the burden of bandwidth-limited mobile verifiers. We remark that simply applying function $f(v)$ in [11], [12] gives $Rev\_Entries \approx$ 5.7M (with Δt=ΔX=1 hour) and $\approx$ 200M (with Δt=ΔX=10 minutes) on 100,000 certificates and 10% revocation, which shows that the approximation can have significant error since the maximum number of entries in the CRL can only be 10,000.

Under the timeliness guarantee of 1 day, we have $Pr_{rev} = 0.005170$ and $Pr_{valid} = 0.994830$. Hence, the probability of a verifier's query to be reported as valid is higher than $\sim$92% reported in [10], where it is simply assumed that Pr_{valid}=$\frac{N}{N+Rev\_Entries}$. The higher percentage is due to our use of $S(t)$, which models the "decaying" effect of queries on the revoked certificates over time which we argue is more realistic. This result is particularly relevant for hash-chaining based schemes since now the mobile verifier is *almost certain* to perform the repeated hash operations to prove the validity (instead of the revocation) of a queried certificate.

We summarize the results in Table III and Table IV, which compare the performance factors of several revocation schemes under revocation latency of 1 hour and 10 minutes respectively.

The evaluation shows clearly the differences and trade-offs between different revocation schemes. It highlights the

Entity	Daily Costs (U=Update, Q=Query)	Unit (/day)	CRL	OCSP	CRS	CREV-I	CREV-II
CA	D_Ovh_{CA} (U+Q)	sec	0.036	$\mathbf{1.07\times10^{6}}$	3.39×10^{-5}	3876	324
	D_Ovh_{CA} (Cert Creation)	sec	0.41	0.41	0.41	0.41	0.41
	$Stor_{CA}$	MB	0.37	0	$\mathbf{8.36\times10^{3}}$	0	13.35
	$D\_Bw_{CA-CMAE}$ (U)	MB	8.84	-	67.92	-	-
	$D\_Bw_{CA-\forall EVCP}$ (U)	MB	-	-	-	1351.36	350.00
	$D\_Bw_{CA-\forall Ver}$ (Q)	MB	-	$\mathbf{3.15\times10^{5}}$	-		
CMAE	D_Ovh_{CMAE} (U+Q)	sec	0.0017	-	0.0017	-	-
	$Stor_{CMAE}$	MB	0.37	-	2.83	-	-
	$D\_Bw_{CA-CMAE}$ (U)	MB	8.84	-	67.92	-	-
	$D\_Bw_{CMAE-\forall Ver}$ (Q)	MB	$\mathbf{2.65\times10^{8}}$	-	1.38×10^{4}	-	-
EVCP	D_Ovh_{EVCP} (U+Q)	sec	-	-	-	0.0077	0.0061
	$Stor_{EVCP}$	MB	-	-	-	4.5×10^{-4}	6.1×10^{-4}
	$D\_Bw_{CA-\exists EVCP}$ (U)	MB	-	-	-	0.014	0.0035
	$D\_Bw_{EVCP-\forall Ver}$ (Q)	MB	-	-	-	3.15	4.29
Verifier	D_Ovh_{Ver} (Q)	sec	0.70	0.70	**16.73**	0.70	0.72
	$D\_Bw_{CA-\exists Ver}$ (Q)	MB	-	0.01	-	-	-
	$D\_Bw_{CMAE-\exists Ver}$ (Q)	MB	**8.84**	-	4.58×10^{-4}	-	-
	$D\_Bw_{EVCP-\exists Ver}$ (Q)	MB	-	-	-	0.011	0.014
Revocation Latency		mins	60	$\approx$0	60	60	60

Table III
COST COMPARISON WITH Δt=1 HOUR. THE COMPUTATION OVERHEADS OF VERIFIER (D_Ovh_{Ver}) ARE CALCULATED BASED ON THE BENCHMARKED CRYPTOGRAPHIC COSTS FOR MOBILE PHONE.

drawback of **CRL**, namely high bandwidth between the CMAE and verifiers ($D\_Bw_{CMAE-Ver}$), and the associated large bandwidth for the individual verifiers. We remark that the costs in our analysis do not cover, apart from the cryptographic operation and bandwidth costs, other costs such as storing and processing the CRL. These additional costs can be significant with mobile devices especially if the CRL size increases further or if the devices issue more queries.

OCSP, on the other hand, has a high CA processing cost (D_Ovh_{CA}) placing a high burden on a centralized CA. Significant bandwidth is also needed (D_Bw_{CA-Ver}) to deliver the OCSP response messages to all verifiers.

CRS has much smaller bandwidth requirements than CRL and OCSP. It however comes with an increased storage cost at the CA for the hash chains ($Stor_{CA}$). In addition, the required repeated hash operations on the verifier is also significant for mobile devices. This is because the total computation costs increases linearly with the timeliness guarantee and the number of queries from the device.

Based on the evaluation, the **CREV** schemes seem to give the best balance of the incurred costs, in the sense that the costs are all moderate and they do not reach the worst-case costs in the other schemes.

VI. CONCLUSION

We have proposed an analytical framework for evaluating certificate revocation schemes. Our framework enhances the revocation model in [11], [12] which was primarily used for analyzing certificate revocation release policies with a granularity in day(s). In addition, our model incorporates more realistic features than existing work such as [9], [10], [13], [14]. We show how to apply the analysis framework to evaluate the two most commonly used revocation mechanisms at present, CRL and OCSP. We also evaluate CRS and CREV. The evaluation of CRL, OCSP and CRS clearly highlights where the tradeoffs in these schemes are. It also shows that CRS might be less desirable on mobile devices since the computation costs are considerably higher, which can be substantial, once high timeliness guarantee is needed. Our evaluation also shows that CREV, a new scheme, is more scalable than OCSP and provides a better cost tradeoff than the other three well-known schemes.

REFERENCES

[1] J. Lopez, R. Oppliger, and G. Pernul, "Why have Public Key Infrastructures failed so far?" *Internet Research*, vol. 15, no. 5, 2005.

[2] P. Gutmann, "PKI: It's not dead, just resting," *Computer*, vol. 35, no. 8, 2002.

[3] ITU-T Recommendation X.509, "IT - OSI - The Directory: Public-key and attribute certificate frameworks," 2000.

[4] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk, *Internet X.509 PKI certificate and CRL profile*, RFC 5280, 2008.

[5] M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams, *X.509 Internet PKI Online Certificate Status Protocol-OCSP*, RFC 2560, 1999.

Entity	Daily Costs (U=Update, Q=Query)	Unit (/day)	CRL	OCSP	CRS	CREV-I	CREV-II
CA	D_Ovh_{CA} (U+Q)	sec	0.21	$\mathbf{1.33\times10^{6}}$	2.03×10^{-4}	22608	1296
	D_Ovh_{CA} (Cert Creation)	sec	0.41	0.41	0.41	0.41	0.41
	$Stor_{CA}$	MB	0.37	0	$\mathbf{5.01\times10^{4}}$	0	19.07
	$D\_Bw_{CA-CMAE}$ (U)	MB	53.05	-	407.50	-	-
	$D\_Bw_{CA-\forall EVCP}$ (U)	MB	-	-	-	7506.56	1491.55
	$D\_Bw_{CA-\forall Ver}$ (Q)	MB	-	$\mathbf{3.94\times10^{5}}$	-	-	-
CMAE	D_Ovh_{CMAE} (U+Q)	sec	0.01	-	0.01	-	-
	$Stor_{CMAE}$	MB	0.37	-	2.83	-	-
	$D\_Bw_{CA-CMAE}$ (U)	MB	53.05	-	407.50	-	-
	$D\_Bw_{CMAE-\forall Ver}$ (Q)	MB	$\mathbf{3.32\times10^{8}}$	-	1.72×10^{4}	-	-
EVCP	D_Ovh_{EVCP} (U+Q)	sec	-	-	-	0.03	0.02
	$Stor_{EVCP}$	MB	-	-	-	4.53×10^{-4}	6.11×10^{-4}
	$D\_Bw_{CA-\exists EVCP}$ (U)	MB	-	-	-	0.080	0.015
	$D\_Bw_{EVCP-\forall Ver}$ (Q)	MB	-	-	-	3.94	5.36
Verifier	D_Ovh_{Ver} (Q)	sec	0.87	0.87	**125.48**	0.87	0.91
	$D\_Bw_{CA-\exists Ver}$ (Q)	MB	-	0.013	-	-	-
	$D\_Bw_{CMAE-\exists Ver}$ (Q)	MB	**11.05**	-	5.72×10^{-4}	-	-
	$D\_Bw_{EVCP-\exists Ver}$ (Q)	MB	-	-	-	0.013	0.018
Revocation Latency		mins	10	≈0	10	10	10

Table IV
COST COMPARISON WITH Δ=10 MINUTES. THE COMPUTATION OVERHEADS OF VERIFIER (D_Ovh_{Ver}) ARE CALCULATED BASED ON THE BENCHMARKED CRYPTOGRAPHIC COSTS FOR MOBILE PHONE.

[6] VeriSign, Inc., "Verisign certification practice statement ver. 3.8.1," 2009.

[7] S. Weintraub, "The numbers don't lie: Mobile devices overtaking PCs, Fortune," http://tech.fortune.cnn.com/2010/08/11/the-great-game-mobile-devices-overtaking-pcs/, 2010.

[8] K. Scheibelhofer, "PKI without revocation checking," in *PKI R&D Workshop*, 2005.

[9] P. Zheng, "Tradeoffs in certificate revocation schemes," *ACM Computer Communication Review*, vol. 33, no. 2, 2003.

[10] T.-L. Lim and A. Lakshminarayanan, "On the performance of certificate validation schemes based on pre-computed responses," in *GLOBECOM'07*, 2007.

[11] C. Ma, N. Hu, and Y. Li, "On the release of CRLs in Public Key Infrastructure," in *USENIX Sec'06*, 2006.

[12] N. Hu, G. K. Tayi, C. Ma, and Y. Li, "Certificate revocation release policies," *Journal of Computer Security*, vol. 17, no. 2, 2009.

[13] V. Goyal, "Certificate revocation using fine grained certificate space partitioning," in *Financial Cryptography and Data Security*, 2007.

[14] Sufatrio and R. H. C. Yap, "Trusted principal-hosted certificate revocation," in *5th International Conference on Trust Management (IFIPTM 2011), Springer*, 2011.

[15] S. Micali, "Efficient certificate revocation," MIT, Tech. Rep., 1996.

[16] T. Perlines Hormann, K. Wrona, and S. Holtmanns, "Evaluation of certificate validation mechanisms," *Computer Comm.*, vol. 29, no. 3, 2006.

[17] K. Hansen, T. Larsen, and K. Olsen, "On the efficiency of fast RSA variants in modern mobile phones," *International Journal of Computer Science and Information Security*, vol. 6, no. 3, 2009.

[18] J. Iliadis, S. Gritzalis, D. Spinellis, D. de Cock, B. Preneel, and D. Gritzalis, "Towards a framework for evaluating certificate status information mechanisms," *Computer Comm.*, vol. 26, no. 16, 2003.

[19] M. Jakobsson, "Fractal hash sequence representation and traversal," in *ISIT '02*, 2002.

[20] S. Berkovits, S. Chokhani, J. A. Furlong, J. A. Geiter, and J. C. Guild, *Public Key Infrastructure Study: Final Report*, MITRE Corporation, 1994.