A More Efficient Use of Delta-CRLs*

David A. Cooper
Computer Security Division
National Institute of Standards and Technology
Gaithersburg, MD 20899-8930
david.cooper@nist.gov

Abstract

Delta-certificate revocation lists (delta-CRLs) were designed to provide a more efficient way to distribute certificate status information. However, as this paper shows, in some environments the benefits of using delta-CRLs will be minimal if delta-CRLs are used as was originally intended. This paper provides an analysis of delta-CRLs that demonstrates the problems associated with issuing delta-CRLs in the "traditional" manner. A new, more efficient technique for issuing delta-CRLs, sliding window delta-CRLs, is presented.

1. Introduction

In a 1994 report, the MITRE Corporation suggested that the distribution of revocation information has the potential to be the most costly aspect of running a large scale public key infrastructure (PKI) [1]. Since the MITRE report was published, several alternative revocation distribution mechanisms have been proposed. This paper provides an analysis of one of these alternative distribution mechanisms, delta-CRLs.

A certificate revocation list (CRL) is a list containing the serial numbers of all certificates issued by a given certification authority (CA) that have been revoked and have not yet expired. A client (relying party) wishing to make use of the information in a certificate (e.g., to verify a signature) must first validate the certificate. In order to validate a certificate, the relying party must, among other things, acquire a recently issued CRL in order to determine whether the certificate has been revoked. Once the client has obtained a CRL, that CRL may be cached for use in future validations. However, after a certain point, a newer CRL must be obtained in order to ensure that validations are based on up-to-date certificate status information.

A delta-CRL is a CRL that only provides information about certificates whose statuses have changed since the issuance of a specific, previously issued CRL. A client in need of more up-to-date certificate status information, that has already obtained a copy of the previously issued CRL, can download the latest delta-CRL instead of downloading the latest full CRL. Since delta-CRLs tend to be significantly smaller than full CRLs, this will tend to reduce the load on the repository and improve the response time for the client.

The use of delta-CRLs in a system will not reduce the request rate for revocation information from the repository. For every request that would have occurred in a system containing only full CRLs, there will be at least one request for a delta-CRL and there may be a second request for a full CRL. The advantage of a system in which delta-CRLs are available is that most of the requests for full CRLs will be replaced by requests for delta-CRLs, which may, in general, be serviced more quickly.

By replacing most of the requests for full CRLs with requests for delta-CRLs, the average request rate for full CRLs can be substantially reduced. However, in order for the full benefits of delta-CRLs to be realized, it it important that the peak request rate for full CRLs be substantially reduced as well. As will be shown in this paper, this may not be the case if delta-CRLs are implemented as was originally intended.

This paper begins with a brief summary of some previous work that has been done to model various methods of certificate revocation [2]. The techniques that were developed in this previous modeling effort are then applied to delta-CRLs as they were originally designed and the resulting model is used to show the problems with the "traditional" method of issuing delta-CRLs. Finally, a new way of implementing delta-CRLs, sliding window delta-CRLs, is presented.

2. Background

In a previous paper, we presented a model for the "traditional" method of certificate revocation, in which a CA

*This paper appears in the *Proceedings of the 2000 IEEE Symposium on Security and Privacy*, pages 190-202, May 2000.

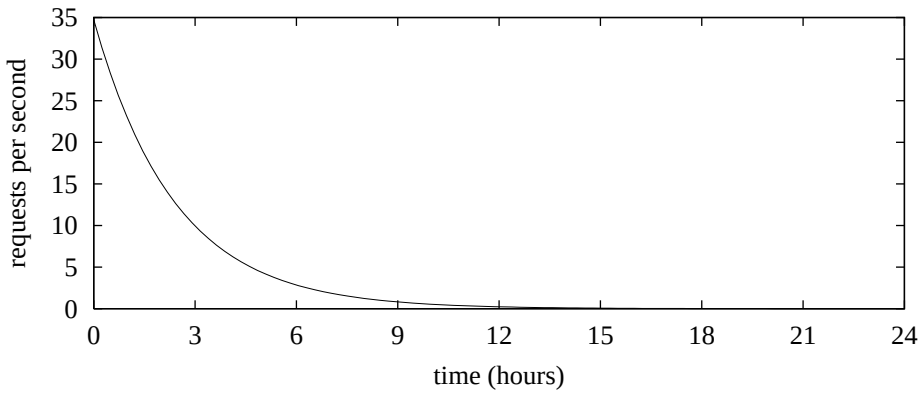

Figure 1. Unsegmented CRL

periodically issues a single CRL listing all unexpired, revoked certificates. In the model, it was assumed that certificate validations by relying parties follow an exponential interarrival probability density [4], that the timings of the validations of different relying parties are independent of each other, that relying parties do not request revocation information from the repository until it is needed to perform a validation, and that they cache any downloaded CRLs until they expire.

In the traditional method of certificate revocation, each CRL includes a **nextUpdate** field that specifies the time at which the next CRL will be issued. Thus, once a relying party has obtained a CRL in order to perform a validation, it will not need to request any further information from the repository to perform future validations until the time specified in the **nextUpdate** field of the CRL in its cache has been reached. So, during the period of time in which a CRL is valid (i.e., the most current), each relying party will make at most one request to the repository for revocation information. This request will be made the first time after the current CRL is issued that the relying party performs a validation.

From the above argument, it can be seen that if a new CRL is issued at time 0, a relying party will request a CRL from the repository in the interval $[t \ldots t+dt]$ if and only if it performs a validation in the interval $[t \ldots t+dt]$ but performed no validations in the interval $[0 \ldots t)$. Since each relying party's validations follow an exponential interarrival probability density, the probability that a relying party will perform no validations in the interval $[0 \ldots t)$ is e^{-vt}, where v is the relying party's validation rate. The probability that a relying party will perform a validation in the interval $[t \ldots t+dt]$, in the limit $dt \to 0$, is $ve^{-v\,dt}dt = v\,dt$. So, the probability that a relying party will request a CRL in the interval $[t \ldots t+dt]$ is $ve^{-vt}dt$. This can be multiplied by the number of relying parties, N, to obtain the expected number of requests for CRLs during the interval $[t \ldots t+dt]$: $N_{req}(t) = Nve^{-vt}dt$. Dividing this equation by dt results in the request rate for CRLs from the repository at time t:

$$R(t) = \frac{N_{req}(t)}{dt} = \frac{Nve^{-vt}dt}{dt} = Nve^{-vt} \qquad (1)$$

Figure 1 shows the request rate for a CRL, issued using the traditional method, over the course of 24 hours. The graph in figure 1 was drawn assuming that a CRL was issued at time 0 and that no other CRLs were issued during the period of time shown in the graph. It was also assumed that there are 300,000 relying parties each validating an average of 10 certificates per day.

2.1. Over-issued CRLs

As can be seen in figure 1, the problem with the traditional method of issuing CRLs is that requests for revocation information are not evenly distributed across time. When a new CRL is issued, the request rate is initially the same as the validation rate, Nv. The request rate then drops off exponentially as an increasing number of relying parties perform validations using CRLs in their caches that were obtained to perform previous validations.

One method described in [2] to spread out requests for revocation information is over-issuing. With over-issuing, a CA issues new CRLs more often than necessary. For example, a CA may issue a new CRL once every 6 hours even though each CRL is valid for 24 hours (see figure 2). The result will be that the CRLs in relying parties' caches will expire at different times and so requests to the repository for new CRLs will be more spread out. Figure 3 shows the request rate for CRLs over the course of 24 hours for a case in which CRLs are valid for 24 hours but are issued every

cRLNumber = 1 **thisUpdate** = Mon. 00:00 **nextUpdate** = Tues. 00:00	**cRLNumber** = 2 **thisUpdate** = Mon. 06:00 **nextUpdate** = Tues. 06:00	**cRLNumber** = 3 **thisUpdate** = Mon. 12:00 **nextUpdate** = Tues. 12:00
cRLNumber = 4 **thisUpdate** = Mon. 18:00 **nextUpdate** = Tues. 18:00	**cRLNumber** = 5 **thisUpdate** = Tues. 00:00 **nextUpdate** = Wed. 00:00	**cRLNumber** = 6 **thisUpdate** = Tues. 06:00 **nextUpdate** = Wed. 06:00

Figure 2. Over-issued CRLs

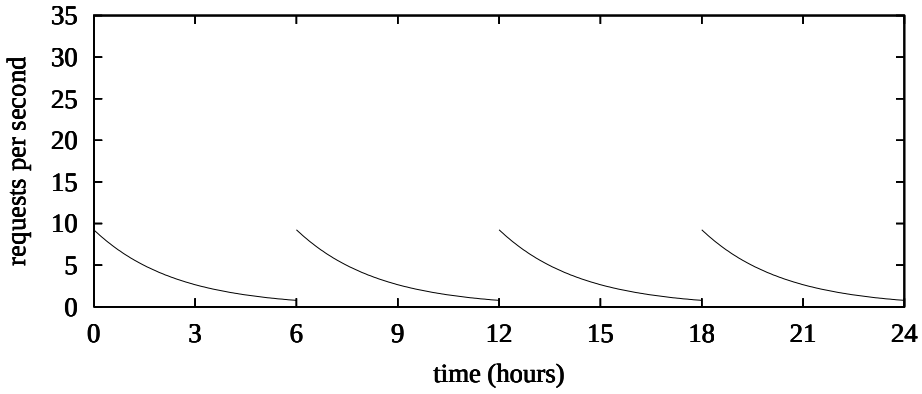

Figure 3. Request rate for over-issued CRLs

6 hours. As in figure 1, it is assumed that there are 300,000 relying parties each validating an average of 10 certificates per day. As can be seen, the result of over-issuing is to spread out the requests for CRLs. The result in this case is a drop in the peak request rate from 34.722 requests/s to 9.250 requests/s.

In order to compute the request rate for over-issued CRLs for the general case, one must first compute the probability that a relying party will request a CRL from the repository in any given interval (where an interval is the period of time between the issuance of two CRLs). A relying party will only request a CRL from the repository in an interval if it performs a validation in that interval and does not already have an unexpired CRL in its cache. If O represents the number of CRLs that are valid at any given time ($O = 4$ in figure 2) and P_{val} is the probability that a relying party will perform a validation in any given interval, then the probability that a relying party will request a CRL in interval n is P_{val} times the probability that the relying party did not request a CRL in any of the previous $O - 1$ intervals:

$$P_{I,n} = P_{val}\left[1 - \sum_{j=n-O+1}^{n-1} P_{I,j}\right] \tag{2}$$

Once the system has reached a steady state, the probability that a relying party will request a CRL in an interval will be the same in each successive interval (i.e., $P_I = P_{I,n} = P_{I,n-1} = \ldots$). So, in the steady state:

$$P_I = P_{val}[1 - (O-1)P_I] \tag{3}$$

Equation (3) can be solved for P_I:

$$P_I = \frac{P_{val}}{(O-1)P_{val} + 1} \tag{4}$$

Dividing equation (4) by P_{val} results in the probability that a relying party that performs a validation in an interval will request a CRL from the repository in that interval. As before, if a relying party requests a CRL in an interval, it will perform the request at the time that it performs its first validation of the interval. Thus, if an interval begins at time 0, the probability that a relying party will request a CRL from the repository between times t and $t + dt$ (in the limit $dt \to 0$) is

$$\frac{ve^{-vt}dt}{(O-1)P_{val} + 1} \tag{5}$$

Equation (5) can be multiplied by the number of relying parties to obtain the expected number of requests for CRLs between times t and $t + dt$. Dividing the result by dt yields the request rate for over-issued CRLs:

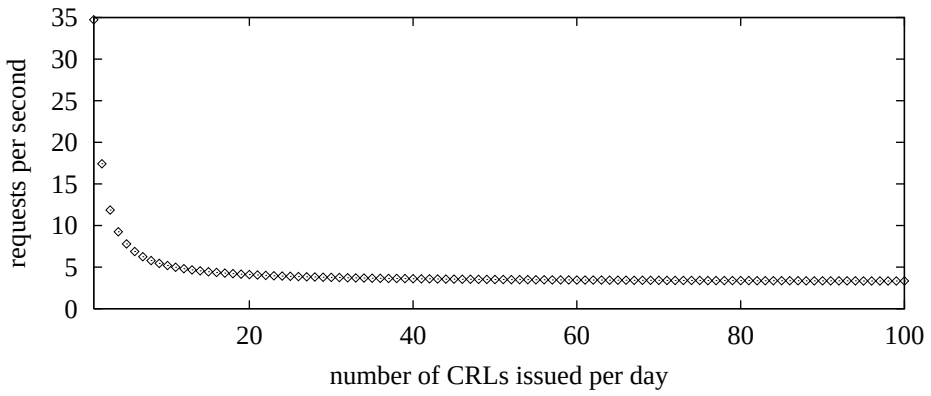

Figure 4. Peak request rate as a function of number of CRLs issued per day

$$R_O(t) = \frac{Nve^{-vt}}{(O-1)P_{val}+1} \quad (6)$$

where t is the amount of time since the last CRL was issued.

Since validations follow an exponential interarrival probability distribution, the probability that a relying party will perform no validations during any given interval is $e^{-vl/O}$ where l is the length of time that a CRL is valid (i.e., an interval is of length l/O). Therefore, $P_{val} = 1 - e^{-vl/O}$ and

$$R_O(t) = \frac{Nve^{-vt}}{(O-1)(1-e^{-vl/O})+1} \quad (7)$$

In general, the more CRLs are over-issued, the more requests for CRLs will be spread out and the lower the peak request rate will be. Figure 4 shows the peak request rate as a function of O, for the case in which $N = 300{,}000$, $v = 10$ validations/day, and $l = 24$ hours. As can be seen in the figure, the peak request rate drops quickly at first, and then levels off as it approaches the theoretical minimum of 3.157 requests/s. The theoretical minimum peak request rate occurs when CRLs are issued continuously and so the theoretical minimum peak request rate is

$$R_O = \lim_{O\to\infty}\left[\frac{Nv}{(O-1)(1-e^{-vl/O})+1}\right] = \frac{Nv}{vl+1} \quad (8)$$

3. Traditional delta-CRLs

With the traditional method for issuing delta-CRLs, a base (or full) CRL is issued periodically and each delta-CRL lists all of the certificates whose statuses have changed since the last base CRL was issued. Whenever a base CRL is issued, a final delta-CRL referencing the previously issued base CRL is also issued. Figure 5 shows an example of delta-CRLs issued in the traditional manner. In this example, relying parties download base CRLs at most once every 4 hours. Delta-CRLs are then obtained to ensure that validations are based on certificate status information that is at most 10 minutes old.

Each validation will require access to a delta-CRL and its corresponding base CRL (either downloaded from the repository or generated locally from a delta-CRL and a previous base CRL). So, the request rate for delta-CRLs will be the same as the request rate for full CRLs in a system that does not use delta-CRLs (see equation (1)).

Base CRLs, on the other hand, will be downloaded less frequently. Using figure 5 as an example, a relying party will only need to download base CRL number 25 if it performed no validations between 00:00 and 04:00 (and so never obtained base CRL number 1) or it if did obtain CRL number 1 but did not perform a validation between 04:00 and 04:10 (in which case it did not download delta-CRL number 25 and so could not generate base CRL number 25 locally).

In general, if a relying party needs to download a given base CRL, it will request that base CRL from the repository the first time that it performs a validation after the base CRL was issued. The period of time between the issuances of two base CRLs can be divided into intervals, where each interval represents the period of time between the issuances of two delta-CRLs. Each interval begins with the issuance of a delta-CRL and ends just before the next delta-CRL is issued. Therefore, each interval will correspond to a single delta-CRL, the delta-CRL issued at the beginning of the interval. Below, the request rate for base CRLs will be determined separately for two types of intervals: intervals corresponding to delta-CRLs issued at the same time as a new base CRL (a "synch" interval) and intervals during

cRLNumber	base CRL	delta-CRL
1	**thisUpdate** = 00:00 **nextUpdate** = 04:00	**thisUpdate** = 00:00 **nextUpdate** = 00:10 **BaseCRLNumber** = 1
2		**thisUpdate** = 00:10 **nextUpdate** = 00:20 **BaseCRLNumber** = 1
⋮	⋮	⋮
24		**thisUpdate** = 03:50 **nextUpdate** = 04:00 **BaseCRLNumber** = 1
25	**thisUpdate** = 04:00 **nextUpdate** = 08:00	**thisUpdate** = 04:00 **nextUpdate** = 04:10 **BaseCRLNumber** = 1
26		**thisUpdate** = 04:10 **nextUpdate** = 04:20 **BaseCRLNumber** = 25
⋮	⋮	⋮

Figure 5. Traditional delta-CRLs

which no base CRL is issued (a "non-synch" interval).

If time t is in a "synch" interval, then a relying party will request a base CRL from the repository at time t if and only if it performed no validations during the time period in which the previous base CRL was the most recent and it is performing its first validation of the current interval at time t. If base CRLs are issued L time units apart ($L = 4$ hours in figure 5) then the probability that a relying party will perform no validations during the period of time in which a base CRL is the most current is e^{-vL}. Similarly, if the current interval began at time 0, the probability that the relying party performed no validations from time 0 to time t is e^{-vt} and the probability that the relying party performed a validation between times t and $t + dt$ (in the limit $dt \rightarrow 0$) is $ve^{-v\,dt}dt = v\,dt$. So, the probability that the relying party will request a base CRL between times t and $t + dt$ is $ve^{-v(t+L)}dt$. If this equation is multiplied by the number of relying parties and divided by dt, the result is the request rate for base CRLs during a "synch" interval:

$$R_s(t) = Nve^{-v(t+L)} \tag{9}$$

If time t is in a "non-synch" interval, then a relying party will request a base CRL from the repository at time t if and only if at time t it is performing its first validation since the most recent base CRL was issued. Thus the request rate for base CRLs during "non-synch" intervals is the same as the request rate for CRLs issued in the traditional manner:

$$R_{ns}(t) = Nve^{-vt} \tag{10}$$

where t is the amount of time since the most recent base CRL was issued.

The problem with the traditional method of issuing delta-CRLs can be seen in equation (10). The request rate for base CRLs, except during the first interval after a base CRL is issued, is the same as if delta-CRLs were not used at all. While the peak request rate is reduced by a factor of e^{-vl} (where l is the length of time that a delta-CRL is valid) as a result of the reduced request rate during the "synch" interval, it may not be significantly reduced if intervals are made short in order to provide relying parties with very fresh certificate status information. Figure 6 shows the request rates for base CRLs and delta-CRLs over the period of time during which one base CRL is valid. In the figure it is assumed that base CRLs and delta-CRLs are issued as in figure 5. As before, it is assumed that there are 300,000 relying parties each validating an average of 10 certificates per day. As can be seen, the peak request rate for base CRLs is reduced from 34.722 requests/s to 32.393 requests/s. However, the 2.329 requests/s for base CRLs saved as a result of using delta-CRLs are replaced by 34.722 requests/s for delta-CRLs.

One way to compare the relative performance of two different types of certificate status distribution mechanisms is to compare the peak bandwidth that is generated by each mechanism. In [1], it is estimated that the size of a CRL is 51 bytes plus 9 bytes for each certificate included on the CRL. If an average of r certificates are revoked each day, certificates are valid for L_c days, and a certificate, at the time of revocation, has an average of $L_c/2$ days until it

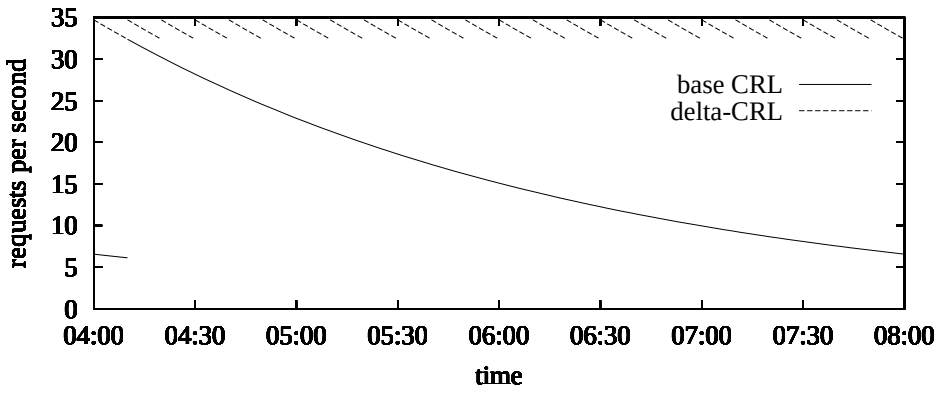

Figure 6. Request rate for base and delta-CRLs in figure 5

expires, then the average size of a full or base CRL will be

$$S_f = 51 + 4.5\, rL_c \tag{11}$$

If a delta-CRL is issued that provides information about status changes over the course of the last w days, then the average size of a delta-CRL will be

$$S_\Delta = 51 + 9\, rw \tag{12}$$

Taking the example from above, if an average of 1000 certificates are revoked each day and certificates are valid for 365 days then, in a system that issues CRLs in the traditional manner, the peak bandwidth will be $34.722 \times (51 + 1642500) = 55696$ Kbytes/s. For the example in figure 6, the peak bandwidth will occur 10 minutes after each base CRL is issued and will be $32.393 \times (51 + 1642500) + 34.722 \times (51 + 62.5) = 51964$ Kbytes/s. Thus, in this example, issuing delta-CRLs in the traditional manner only reduces the peak bandwidth by 6.7% over the traditional method of issuing CRLs.

4. Sliding window delta-CRLs

In section 2, it was shown that requests for CRLs can be spread out by over-issuing the CRLs. This section will present a new method of issuing delta-CRLs, sliding window delta-CRLs, that provides the benefits of over-issuing in a system that uses delta-CRLs.

The idea behind sliding window delta-CRLs can be seen by looking at figure 5. Each delta-CRL in figure 5 provides information about any certificate whose status has changed between the time the base CRL referenced by **BaseCRLNumber** was issued and the time the delta-CRL was issued. In other words, the delta-CRL provides information about all status changes that occurred during a certain "window" of time. The problem with the traditional method of issuing delta-CRLs is that the "window" sizes of the delta-CRLs vary. In the example in figure 5, the window sizes for the delta-CRLs vary between 10 minutes and 4 hours. As can be seen in figure 6, the request rate for base CRLs drops as the window size increases and then jumps up again after "synch" intervals when the window size is reduced to 10 minutes.

In general, if a relying party last obtained fresh certificate status information at time t and obtains a delta-CRL that references a base CRL that was issued at time $t' \leq t$ then the relying party can use the delta-CRL to update its local cache without obtaining a new base CRL. So, the larger the window sizes of the delta-CRLs, the lower the request rate will be for base CRLs. The idea behind sliding window delta-CRLs, then, is for each delta-CRL to have the same, large window size instead of using variable size windows as with the traditional method.

Figure 7 shows an example of sliding window delta-CRLs. In this figure, each delta-CRL is valid for 10 minutes but has a window size of 4 hours. Since, in this example, each **cRLNumber** is referenced as a **BaseCRLNumber**, a base CRL is issued (over-issued) each time a delta-CRL is issued. This is a result of a requirement of the X.509 **deltaCRLIndicator** extension which specifies that the CRL referenced in the **BaseCRLNumber** of a delta-CRL must have been issued as a base CRL [3]. If delta-CRLs are instead indicated as being delta-CRLs using the **crlScope** extension, then it is only necessary to issue a new base CRL at least once every 4 hours.

4.1. The performance of sliding window delta-CRLs

As with the traditional method of issuing delta-CRLs, the request rate for delta-CRLs in a system that uses sliding window delta-CRLs is the same as the request rate for

cRLNumber	base CRL	delta-CRL
⋮	⋮	⋮
25	**thisUpdate** = 04:00 **nextUpdate** = 08:00	**thisUpdate** = 04:00 **nextUpdate** = 04:10 **BaseCRLNumber** = 1
26	**thisUpdate** = 04:10 **nextUpdate** = 08:10	**thisUpdate** = 04:10 **nextUpdate** = 04:20 **BaseCRLNumber** = 2
⋮	⋮	⋮
48	**thisUpdate** = 07:50 **nextUpdate** = 11:50	**thisUpdate** = 07:50 **nextUpdate** = 08:00 **BaseCRLNumber** = 24
49	**thisUpdate** = 08:00 **nextUpdate** = 12:00	**thisUpdate** = 08:00 **nextUpdate** = 08:10 **BaseCRLNumber** = 25
⋮	⋮	⋮

Figure 7. Sliding window delta-CRLs

full CRLs in a system that issues CRLs in the traditional manner. Therefore, in order to determine the performance of sliding window delta-CRLs, it is only necessary to determine the request rate for base CRLs.

In performing a validation, a relying party will only need to obtain a base CRL if the last time it performed a validation was before the time that the base CRL referenced by the current delta-CRL was issued. If w represents the window size of the current delta-CRL and t is the amount of time since the current delta-CRL was issued, then the probability that a relying party will perform no validations between the time the base CRL referenced by the current delta-CRL was issued and time t is $e^{-v(t+w)}$. The probability that a relying party will perform a validation between times t and $t + dt$, in the limit $dt \to 0$, is $ve^{-v\,dt}dt = v\,dt$. Multiplying the number of relying parties, N, by the product of these two equations and dividing by dt results in the request rate for base CRLs at time t:

$$R_{s\Delta}(t) = Nve^{-v(t+w)} \tag{13}$$

Figure 8 shows the request rate for base CRLs and delta-CRLs over same period of time as shown in figure 6 assuming that base CRLs and delta-CRLs are issued as shown in figure 7. As in the previous graphs, it is assumed that there are 300,000 relying parties each validating an average of 10 certificates per day. As can be seen, the peak request rate for base CRLs is now only 6.558 requests/s. This is the same as the request rate for base CRLs at 4:00 in figure 6, the time at which the window size is at its largest in the traditional delta-CRL example.

As can be seen in equation (13), the peak request rate for base CRLs can be made arbitrarily small by increasing the window size. At the same time, the request rate for delta-CRLs is not affected by the window size. However, as the window size increases, so does the size of each delta-CRL (see equation (12)). Continuing with the example from section 3, if delta-CRLs are issued as in figure 7, the peak bandwidth will be $6.558 \times (51 + 1642500) + 34.722 \times (51 + 1500) = 10572$ Kbytes/s, a savings of 79.7% over the traditional method of issuing delta-CRLs. The peak bandwidth can be reduced even further by using a window size of 18 hours. This results in a peak bandwidth of $1.920 \times 10^{-2} \times (51 + 1642500) + 34.722 \times (51 + 6750) = 261.4$ Kbytes/s, a savings of 99.5% over the traditional method of issuing delta-CRLs.

5. Over-issuing delta-CRLs

If relying parties are required to obtain fresh certificate status information very frequently, such as every 10 minutes, then it may not be possible to significantly reduce the peak request rate for delta-CRLs. However, if the validity periods of delta-CRLs are long enough, then it may be possible to significantly reduce the peak request rate for delta-CRLs in addition to the peak request rate for base CRLs by over-issuing the delta-CRLs.

As before, the request rate for over-issued delta-CRLs will be the same as the request rate for over-issued CRLs in a system that does not use delta-CRLs:

$$R_{\Delta}(t) = \frac{Nve^{-vt}}{(O-1)(1-e^{-vl/O})+1} \tag{14}$$

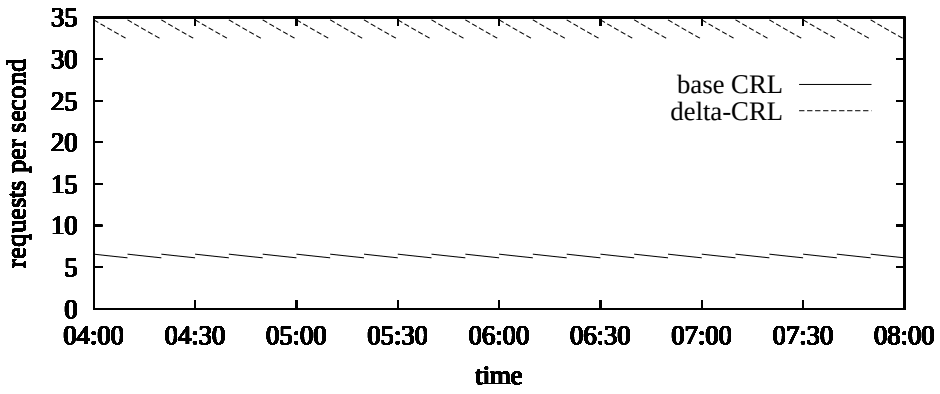

Figure 8. Request rate for base and delta-CRLs in figure 7

where l is the length of time that a delta-CRL is valid, O is the number of delta-CRLs that are valid at any given time, and t is the amount of time since the most recent delta-CRL was issued.

The request rate for base CRLs can be determined by first determining the probability that a relying party will request a base CRL in any given interval, where an interval is the period of time between the issuance of a delta-CRL and the issuance of the next delta-CRL. A relying party that performs a validation in a given interval will request a base CRL in that interval if and only if the amount of time since it last received updated certificate status information exceeds the window size of the delta-CRLs[1]. This can happen in one of two ways. One possibility is that the relying party did not perform any validations during the period (window) covered by the delta-CRL. The other possibility is that the relying party performed one or more validations, but all of the validations were performed using a delta-CRL that was retrieved before the beginning of the period covered by the current delta-CRL. So, if P_{val} is the probability that a relying party will perform a validation during the course of an interval and P_Δ is the probability that a relying party will request a delta-CRL during the course of an interval, then the probability that a relying party will request a base CRL during the course of an interval is[2]

$$P_b = P_{val}\left\{(1-P_{val})^{wO/l} + P_\Delta \sum_{i=1}^{O-1}\left[1-(1-P_{val})^i\right](1-P_{val})^{wO/l-i}\right\} \quad (15)$$

Since validations follow an exponential interarrival probability density, the probability that a relying party will not perform any validations during the course of an interval is $e^{-vl/O}$ and so $P_{val} = 1 - e^{-vl/O}$. The probability that a relying party will request a delta-CRL in any given interval can be computed by integrating equation (14) over the course of an interval (using $N = 1$):

$$\begin{aligned} P_\Delta &= \int_0^{l/O} \frac{ve^{-vt}dt}{(O-1)(1-e^{-vl/O})+1} \\ &= \frac{1-e^{-vl/O}}{(O-1)(1-e^{-vl/O})+1} \end{aligned} \quad (16)$$

Using these equations for P_{val} and P_Δ, equation (15) can be simplified to

$$\begin{aligned} P_b &= \frac{(1-e^{-vl/O})\,e^{-(w+l/O-l)v}}{(O-1)(1-e^{-vl/O})+1} \\ &= P_\Delta\, e^{-(w+l/O-l)v} \end{aligned} \quad (17)$$

Equation (17) can now be used to compute the request rate for base CRLs. If a relying party requests a base CRL during the course of an interval, then it will request the base CRL at the time that it performs its first validation of the interval. If the interval begins at time 0, then the probability that the relying party will perform its first validation of the interval between times t and $t + dt$ (in the limit $dt \to 0$) is $ve^{-vt}dt$. The probability that a relying party that performs a validation during an interval will need to request a base CRL during that interval can be computed by dividing equation (17) by P_{val}:

$$\frac{e^{-(w+l/O-l)v}}{(O-1)(1-e^{-vl/O})+1} \quad (18)$$

Multiplying the number of relying parties, N, by the product of these two equations and dividing by dt results in the request rate for base CRLs over the course of an interval:

[1] It is assumed that the window size of a delta-CRL is at least as large as the validity period of the delta-CRLs.

[2] See appendix A.1 for a more detailed explanation of the derivation of P_b.

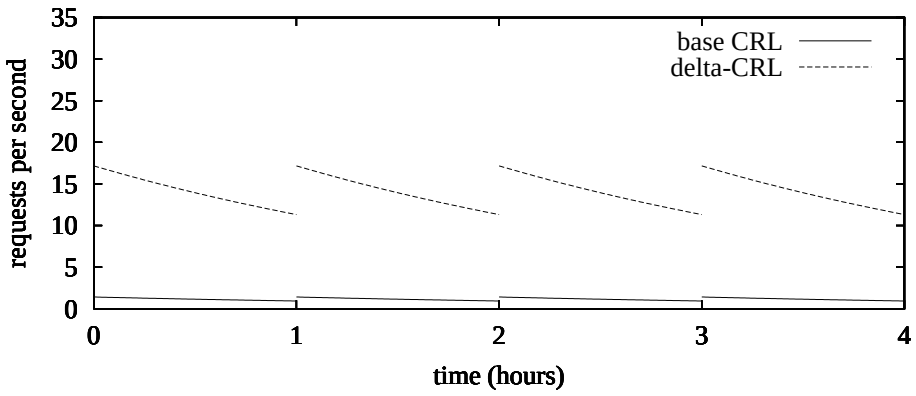

Figure 9. Request rate for base CRLs and over-issued delta-CRLs

$$R_b(t) = \frac{Nve^{-(t+w+l/O-l)v}}{(O-1)(1-e^{-vl/O})+1} = R_\Delta(t)\, e^{-(w+l/O-l)v} \quad (19)$$

Figure 9 shows the request rate for base CRLs and delta-CRLs for a scenario in which delta-CRLs are issued once an hour, are valid for 4 hours, and have a window size of 9 hours. As before, it is assumed that there are 300,000 relying parties each validating an average of 10 certificates per day.

If it is assumed that an average of 1000 certificates are revoked each day and that certificates are valid for 365 days then, using equations (11) and (12) for the sizes of base CRLs and delta-CRLs respectively, the peak bandwidth for the scenario shown in figure 9 is 2318 Kbytes/s. If the optimal window size of 21 hours were used, the peak bandwidth would drop to 148.1 Kbytes/s. This compares to a minimum peak bandwidth (using a window size of 20 hours) of 269.4 Kbytes/s if sliding window delta-CRLs were used, but the delta-CRLs were not over-issued. If traditional delta-CRLs were used, the peak bandwidth would be 10572 Kbytes/s.

6. Choosing an optimal window size

This section will show how one can choose the optimal window size for a given environment. It will be assumed in this section that the CRL lifetime, l, is fixed as is the number of relying parties, N, and the validation rate, v. As was shown earlier, the peak request rate for delta-CRLs will drop as the amount of over-issuing increases. The value for O, then, needs to be chosen by determining the point at which the cost of increasing the issuance frequency for delta-CRLs exceeds the benefit derived from the decreased peak request rate for delta-CRLs.

Once the amount of over-issuing has been chosen, one can select a window size. The selection of a window size also involves a trade-off. As the window size increases, the request rate for base CRLs decreases. However, increasing the window size also increases the size of the delta-CRLs. While there may be many factors that need to be taken into account in choosing an optimal window size, this section, as a simple example, will show how to determine the window size that will minimize the peak bandwidth.

The peak bandwidth for a sliding window delta-CRL system can be computed as $B = S_f R_b(0) + S_\Delta R_\Delta(0)$ where S_f is the size of a base CRL (equation (11)), S_Δ is the size of a delta-CRL (equation (12)), $R_b(0)$ is the peak request rate for base CRLs (equation (19)), and $R_\Delta(0)$ is the peak request rate for delta-CRLs (equation (14)). The optimal window size can be computed by solving the equation $\frac{dB}{dw} = 0$ for w. The result is that peak bandwidth is minimized when[3]

$$w = l - \frac{l}{O} + \left(\frac{1}{v}\right) \lg\left(\frac{(51+4.5rL_c)v}{9r}\right) \quad (20)$$

In general, if a CRL header is S_H bytes and a CRL entry is S_E bytes then the optimal window size is

$$w = l - \frac{l}{O} + \left(\frac{1}{v}\right) \lg\left(\frac{(S_H+0.5S_E\, rL_c)v}{S_E\, r}\right) \quad (21)$$

Table 1 demonstrates the advantages of over-issuing when using sliding window delta-CRLs. This table shows the request rates and peak bandwidth for a system in which there are 300,000 relying parties each validating an average of 10 certificates/day. It is assumed that an average of 1000 certificates are revoked each day, that certificates

[3] See appendix A.2 for a more complete derivation of equations (20) and (21).

Table 1. Peak request rates and bandwidth with sliding window delta-CRLs

O	w	$R_\Delta(0)$	$R_b(0)$	Bandwidth (Kbytes/s)
1	20	34.72	8.35×10^{-3}	269
2	20	22.18	1.23×10^{-2}	183
4	21	17.17	9.50×10^{-3}	148
10	21.6	14.58	8.06×10^{-3}	129
100	22	13.17	7.16×10^{-3}	118
∞	22.02	13.02	7.14×10^{-3}	117

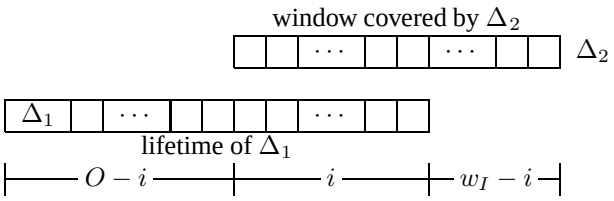

Figure 10. Sample delta-CRL request scenario

are valid for 365 days, and that each delta-CRL is valid for 4 hours. In each entry in the table, the optimal window size has been chosen using equation (20), but with the constraints that the window size must be at least as large as the delta-CRL validity period (i.e., $w \geq l$) and that the window size must be a integral multiple of l/O.

As a general rule, as the validity period for delta-CRLs, l, increases, the peak bandwidth decreases. However, if delta-CRLs are not over-issued, this may not always hold. If delta-CRLs are not over-issued, then the optimal window size is $w = \left(\frac{1}{v}\right) \lg \left(\frac{(S_H + 0.5 S_E\, r L_c) v}{S_E\, r}\right)$, a value that does not depend on l. For example, if $S_H = 51$, $S_E = 9$, $r = 1000$, $v = 10$ validations/day, and $L_c = 365$ days, the optimal window size for $O = 1$ is 0.75. However, the window size must be at least as large as the valid lifetime of a delta-CRL. So, if $l = 1$ day, the optimal window size is 1 day and the minimum peak bandwidth is 309 Kbytes/s. However, if l were reduced to 0.75, then the peak bandwidth would drop to 261 Kbytes/s.

7. Conclusions

This paper has presented a model for the traditional method of issuing delta-CRLs. As was shown, delta-CRLs, when issued in the traditional manner, may fail to provide the performance advantages for which they were designed. This is particularly the case when relying parties must always perform validations based on very fresh certificate status information. A new technique for issuing delta-CRLs, sliding window delta-CRLs, that overcomes the problems that are encountered when delta-CRLs are issued in the traditional manner was presented. As was shown, issuing delta-CRLs in the new way provides the performance benefits for which delta-CRLs were originally designed.

References

[1] S. Berkovits, S. Chokhani, J. A. Furlong, J. A. Geiter, and J. C. Guild. *Public Key Infrastructure Study: Final Report*. Produced by the MITRE Corporation for NIST, Apr. 1994.

[2] D. A. Cooper. A model of certificate revocation. In *Proceedings of the Fifteenth Annual Computer Security Applications Conference*, pages 256–264, Dec. 1999.

[3] ISO/IEC JTC1/SC06/WG7. Draft amendment on certificate extensions, Oct. 1999.

[4] A. S. Tanenbaum. *Computer Networks*. Prentice-Hall, Inc., second edition, 1989.

A. Explanations for equations and derivations

This appendix provides more detailed information on the derivations of some of the equations in sections 5 and 6.

A.1. Equations (15) and (17)

This section will explain how equation (15) was derived and will show, step-by-step, how equation (17) was derived from equation (15).

As was stated in section 5, a relying party performing a validation in an interval will request a base CRL in that interval if the relying party did not perform any validations

$$
\begin{aligned}
P_b &= P_{val}\left\{(1-P_{val})^{wO/l} + P_\Delta \sum_{i=1}^{O-1}\left\{\left[1-(1-P_{val})^i\right](1-P_{val})^{wO/l-i}\right\}\right\} \\
&= P_{val}\left\{e^{-wv} + P_\Delta \sum_{i=1}^{O-1}\left\{\left[1-e^{-ivl/O}\right]e^{-(w-il/O)v}\right\}\right\} \\
&= P_{val}\left\{e^{-wv} + P_\Delta \sum_{i=1}^{O-1}\left\{e^{-wv}e^{ivl/O} - e^{-wv}\right\}\right\} \\
&= P_{val}\, e^{-wv}\left\{1 + P_\Delta \sum_{i=1}^{O-1}\left\{e^{ivl/O} - 1\right\}\right\} \\
&= P_{val}\, e^{-wv}\left\{1 + P_\Delta \left[-(O-1) + \frac{1-e^{vl}}{1-e^{vl/O}} - 1\right]\right\} \\
&= P_{val}\, e^{-wv}\left\{1 + P_\Delta \left[\frac{1-e^{vl}}{1-e^{vl/O}} - O\right]\right\} \\
&= P_{val}\, e^{-wv}\left\{\frac{(O-1)P_{val}+1}{(O-1)P_{val}+1} + \left[\frac{1-e^{-vl/O}}{(O-1)P_{val}+1}\right]\left[\frac{1-e^{vl}}{1-e^{vl/O}}\right] - \frac{OP_{val}}{(O-1)P_{val}+1}\right\} \\
&= P_{val}\, e^{-wv}\left[\frac{1-P_{val}}{(O-1)P_{val}+1} + \frac{e^{-vl/O}(1-e^{-vl/O})(1-e^{vl})}{e^{-vl/O}[(O-1)P_{val}+1](1-e^{vl/O})}\right] \\
&= P_{val}\, e^{-wv}\left[\frac{e^{-vl/O}}{(O-1)P_{val}+1} + \frac{-e^{-vl/O}(e^{-vl/O}-1)(1-e^{vl})}{(e^{-vl/O}-1)[(O-1)P_{val}+1]}\right] \\
&= P_{val}\, e^{-wv}\left[\frac{e^{-vl/O}}{(O-1)P_{val}+1} - \frac{e^{-vl/O}(1-e^{vl})}{(O-1)P_{val}+1}\right] \\
&= \frac{P_{val}\, e^{-wv}e^{-vl/O}}{(O-1)P_{val}+1}\left[1-(1-e^{vl})\right] \\
&= \frac{P_{val}\, e^{-(w+l/O-l)v}}{(O-1)P_{val}+1} \\
&= \frac{(1-e^{-vl/O})\, e^{-(w+l/O-l)v}}{(O-1)(1-e^{-vl/O})+1}
\end{aligned}
$$

Figure 11. Simplification of P_b

during the period covered by the current delta-CRL or if all of the validations performed by the relying party during the period covered by the current delta-CRL made use of a previously downloaded delta-CRL.

P_{val} was defined as the probability that a relying party will perform a validation during the course of an interval. So, the probability that a relying party will perform no validations during the course of an interval is $1 - P_{val}$. Since there are wO/l intervals in the period covered by a delta-CRL, the probability that a relying party will perform no validations during the period covered by a delta-CRL is

$$(1 - P_{val})^{wO/l} \tag{22}$$

In order for a relying party to perform a validation during the period covered by the current delta-CRL using a previously downloaded delta-CRL, the lifetime of the previous delta-CRL must overlap with the window of the current delta-CRL, as is shown in figure 10. In figure 10, the current delta-CRL, Δ_2, was issued at the beginning of interval p and has a window size of w_I. Thus, Δ_2 covers intervals $p - w_I$ through $p - 1$. The last delta-CRL downloaded by the relying party, Δ_1, was issued at the beginning of interval $p - w_I - O + i$ and is valid for O intervals. Thus, the lifetime of Δ_1 and the window covered by Δ_2 overlap by i intervals.

In order for the scenario in figure 10 to occur, the relying party must download a delta-CRL during interval $p - w_I - O + i$, perform one or more validations during

$$
\begin{aligned}
B(w) &= S_f R_b(0) + S_\Delta R_\Delta(0) \\
&= \left(S_H + \frac{S_E\, r L_c}{2}\right)\left[\frac{Nve^{-(w+l/O-l)v}}{(O-1)(1-e^{-vl/O})+1}\right] + (S_H + S_E\, rw)\left[\frac{Nv}{(O-1)(1-e^{-vl/O})+1}\right]
\end{aligned}
$$

$$
\begin{aligned}
\frac{dB(w)}{dw} &= \left(S_H + \frac{S_E\, r L_c}{2}\right)\left[\frac{-Nv^2e^{-(w+l/O-l)v}}{(O-1)(1-e^{-vl/O})+1}\right] + (S_E\, r)\left[\frac{Nv}{(O-1)(1-e^{-vl/O})+1}\right] \\
0 &= \left(S_H + \frac{S_E\, r L_c}{2}\right)\left[\frac{-Nv^2e^{-(w+l/O-l)v}}{(O-1)(1-e^{-vl/O})+1}\right] + (S_E\, r)\left[\frac{Nv}{(O-1)(1-e^{-vl/O})+1}\right] \\
0 &= \left\{\left(S_H + \frac{S_E\, r L_c}{2}\right)\left[-ve^{-(w+l/O-l)v}\right] + (S_E\, r)\right\}\left[\frac{Nv}{(O-1)(1-e^{-vl/O})+1}\right] \\
0 &= \left(S_H + \frac{S_E\, r L_c}{2}\right)\left[-ve^{-(w+l/O-l)v}\right] + (S_E\, r) \\
S_E\, r &= \left(S_H + \frac{S_E\, r L_c}{2}\right)ve^{-(w+l/O-l)v} \\
e^{(w+l/O-l)v} &= \frac{(S_H + 0.5 S_E\, r L_c)v}{S_E\, r} \\
\left(w + \frac{l}{O} - l\right)v &= \lg\left(\frac{(S_H + 0.5 S_E\, r L_c)v}{S_E\, r}\right) \\
w &= l - \frac{l}{O} + \left(\frac{1}{v}\right)\lg\left(\frac{(S_H + 0.5 S_E\, r L_c)v}{S_E\, r}\right)
\end{aligned}
$$

Figure 12. Derivation of optimal window size

intervals $p - w_I$ through $p - w_I + i - 1$, and then perform no validations during intervals $p - w_I + i$ through $p - 1$. The probability that the relying party will download a delta-CRL during interval $p - w_I - O + i$ is P_Δ (see equation (16)). The probability that the relying party will perform no validations during intervals $p - w_I$ through $p - w_I + i - 1$ is $(1 - P_{val})^i$. So, the probability that the relying party will perform one or more validations during intervals $p - w_I$ through $p - w_I + i - 1$ is $1 - (1 - P_{val})^i$. Finally, the probability that the relying party will perform no validations during intervals $p - w_I + i$ through $p - 1$ is $(1 - P_{val})^{w_I - i} = (1 - P_{val})^{wO/l - i}$. Therefore, the probability that a relying party that performs a validation during interval p will request delta-CRLs under the circumstances depicted in figure 10 is

$$P_\Delta[1 - (1 - P_{val})^i](1 - P_{val})^{wO/l - i} \qquad (23)$$

The value of i in figure 10 can range between 1 and $O - 1$ and so equation (23) must be summed over all values of i between 1 and $O - 1$. This can be combined with equation (22) to obtain the probability that a relying party that performs a validation in an interval will request a base CRL in that interval:

$$(1 - P_{val})^{wO/l} + P_\Delta \sum_{i=1}^{O-1} \left[1 - (1 - P_{val})^i\right](1 - P_{val})^{wO/l - i} \qquad (24)$$

In order to obtain the probability that a relying party will request a base CRL in an interval, equation (24) must be multiplied by the probability that a relying party will perform a validation during the course an interval:

$$P_b = P_{val}\left\{(1 - P_{val})^{wO/l} + P_\Delta \sum_{i=1}^{O-1} \left[1 - (1 - P_{val})^i\right](1 - P_{val})^{wO/l - i}\right\} \qquad (25)$$

Equation (25) can then be simplified as shown in figure 11.

A.2. Equations (20) and (21)

The peak bandwidth for a sliding window delta-CRL system that may over-issue delta-CRLs can be computed

by adding the peak bandwidth resulting from requests for base CRLs to the peak bandwidth resulting from requests for delta-CRLs. The peak bandwidth for base CRLs can be obtained by multiplying the peak request rate for base CRLs (equation (19) where $t = 0$) by the size of a base CRL (equation (11)). The peak bandwidth for delta-CRLs can be obtained by multiplying the peak request rate for delta-CRLs (equation (14) where $t = 0$) by the size of a delta-CRL (equation (12)). The resulting equation for the peak bandwidth, $B(w)$, is shown in figure 12.

Since $B(w)$ is convex, the optimal window size can be computed by solving the equation $\frac{dB(w)}{dw} = 0$ for w as is shown in figure 12. Equation (20) can be derived from equation (21) by substituting $S_H = 51$ and $S_E = 9$.