ISSN: 0975-5665, Nov, 2013 www.ijrct.org

Certificate Revocation using Online Certificate Status Protocol (OCSP)

G.ShivaJyothi,Shivajyothi.goli@yahoo.com, A.P, INDIA

Abstract:

With the increasing acceptance of digital certificates, there has been a gaining impetus for methods to nullify the compromised digital certificates and enable the end user to receive this information before he trusts a revoked certificate. The problem of certificate revocation is getting more and more crucial with the development of wide spread PKIs. In this paper we present how Certificate Revocation Lists (CRL) used for certificate revocation with different techniques. We explained to overcome disadvantages of CRL by the Online Certificate Status Protocol (OCSP) for revoking digital certificates using Requester-Responder approach.

1. Introduction

The certificate revocation list (CRL) is currently the most widespread implementation mechanism for certificate revocation within a public-key infrastructure (PKI). It is simplistic in design, but problematic for operation and maintenance in a large scale PKI. A CRL is a list of revoked certificates that have been issued and subsequently revoked by a given Certification Authority. Certificates may be revoked for a number of reasons including failure or compromise of a device that is using a given cert, compromise of the key pair used by a certificate, or errors within an issued certificate, such as an incorrect identity or the need to accommodate a name change. The mechanism used for certificate revocation depends on the Certification Authority. Most Certification Authorities support cert revocation from the management interface. The Revoked certificates are represented in the CRL by their serial numbers.

A CRL is a comprehensive list of digital certificates suspended or revoked prior to their expiration date by the root certification authority (root CA) or subordinate CA that generated them. The comprehensiveness of the CRL is directly proportional to the date and time the CRL is published and the information/requests the CA receives for certificate suspension or revocation. If a network device is attempting to verify the validity of a certificate, it will download and scan the current CRL for the serial number of the presented cert. The CRL is signed by the Certification Authority to ensure the authenticity of the document and may be distributed through a variety of protocols, such as http, ldap, tftp, or other services. CRLs are generally published on a periodic interval, or Certification Authorities may publish a new CRL any time a certificate they are responsible for is revoked. Like most documents created by a PKI, the CRL has an expiration time, date, and all components of a PKI that will verify that certificates should download a new copy of the CRL, when the old CRL expires. The CRL may eventually grow to a cumbersome size in very large PKIs. If a PKI has revoked so many certificates that the CRL exceeds a cumbersome size, it is worthwhile to look into breaking the CRL into multiple files. This will save bandwidth and time when cryptography peers

download a new copy of the CRL and will ensure that a router will have sufficient buffer space to hold and scan the CRL for revoked certificates. The specific of dividing the CRL into a number of more manageable files is outside of this document's scope; however, PKI documentation should offer design guidance for deploying the optimal CRL distribution scheme. CRLs are practical for most PKI applications, but may not be appropriate for some uses. Some instances where CRLs are not adequate include:

• Large numbers of revoked certificates or multiple CRLs. CRLs in cache on devices can consume a large quantity of memory. Downloading large CRLs over low-speed links may use excessive bandwidth, which causes network congestion. Frequent CRL expiration. If CRLs expire frequently, the Certificate Distribution Point (CDP) will be heavily loaded, and frequent CRL download will burden network devices and bandwidth with non-production traffic.

• Immediate notification of cert revocation is required. Some high-security applications require more immediate notification of cert revocation. If CRL has a two day expiration interval, it may be up to 48 hours before a router downloads a new CRL. This leaves a long period of time before a router is notified that a certificate is no longer valid. There are two main methodologies for disseminating CRL information, *pulling CRL* and *pushing CRL* [9]. However, these two methods have resulted in a variety of CRL implementations, e.g. delta-CRL, over-issued CRL, and distribution point (DP) CRL or segmented CRL ([6],[9]). A general model for a CRL with specific details for each implementation follows.

1.1.1 Delta-CRL

The implementers of a PKI must be concerned about the potential size of a CRL. When the CRL grows, so does the time it takes to push it through the network, since the entire CRL needs to be transmitted. This can result in latency to a user's query about a certificate's status, or even network congestion. The problem is exacerbated when a particular CA has issued a large number of certificates with long validity periods. To counter this, the delta-CRL model was developed.

The basic concept of delta-CRLs is that the users and/or PKI-enabled applications cache CRLs locally. The first time the CA issues a CRL, it is termed the base-CRL. Every additional CRL is in the form of a delta-CRL, outlining the additions and subtractions to the base-CRL and/or any previously received delta-CRLs. The user and/or PKI-enabled application stores the base-CRL and all delta-CRLs locally. This process may not be a simple as it sounds, and more research is needed to validate the protocol. When a user needs to verify a certificate he contacts the directory and queries for the latest delta-CRL. If he/she already has it, the verification can take place locally. If not, all missing delta-CRLs must be downloaded and the locally cached CRLs must be updated. Only after the update is complete can the validation occur. [5].

1.1.2 Distribution point (DP) CRL (segmented CRL)

Whereas the delta-CRL model attempted to address the problem of CRL size by issuing a base-CRL followed by delta-CRLs, the DP-CRL model addresses the same problem, but via segmenting the CRL. The segmenting is done along some previously determined lines and the segments are either kept in separate files on

the central directory or passed to distribution points on other directories. All of this can be done due to the distribution point extension field in the X.509v3 certificate and the "Issuing Distribution Point" extension field in the X.509v2 CRL. When a user or PKI-enabled application wants to verify a certificate it looks up the "Distribution Point" extension field in the X.509v3 certificate and uses the information in the field to go to the issuing distribution point.

1.1.3 Sliding Window Delta CRL

A problem with Delta CRL's is the rate of request for full CRL's in relation to the rate of request for Delta CRL's. With a traditional Delta CRL, the expiration date for the retrieved CRL is governed by the NextUpdate field in the CRL. The end user will request Delta CRL's until the NextUpdate time in the cached CRL is reached. After the newest full CRL is published, the users seeking to verify certificates will seek to obtain the full CRL, and statistically the majority of users in a system will seek a full CRL within a relatively short period centered around the arrival of the *NextUpdate* publish time. This period of time represents a spike in system and resource utilization which can adversely affect the timeliness of revocation checking should be minimized. The graph in Figure 1 demonstrates this concept (This graph indicates the probability of request for a full CRL for 30,000 end users requesting 10 certificates per day with a full CRL issued only at time zero).

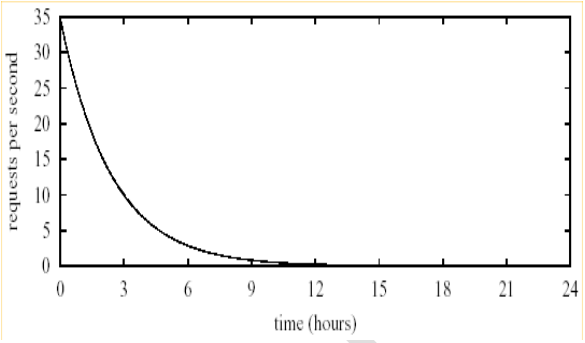

Figure 1. Un-Segmented CRL16.

1.1.4 Over-issued CRL

One of the deficiencies with the basic CRL model is the premise that there will be periods when the CRL repository/directory is more heavily utilized than others, and that this heavy usage will affect overall PKI performance and security. The reason stems from the hypothesis that users and PKI-enabled applications, which cache CRLs, will uniformly try to access the new CRL as soon as it is issued, i.e. when its predecessor CRL expires. To solve this problem, over-issued CRLS were proposed as an alternative, to reduce peak directory usage [9]. It places an increased burden on the CAs, by having them issue CRLs more frequently, i.e. daily or hourly. This results in numerous valid/not expired CRLs which contain overlapping validity periods. As an example, CA 1 issues CRL 1 on 1 Jan. It expires on 1 Feb. CA 1 then continues to issue CRLs every day, i.e. CRL 2 (expires 2 Feb),CRL 3 (expires 3 Feb), etc. If user 1 first needs to validate a certificate on 1 Jan he will download CRL 1. If the next time he needs to validate a certificate is 14 Jan, he will discard CRL 1 and cache CRL 14. This procedure results in more of a random-like distribution, although CRLs issued on Monday – Friday will probably be more heavily utilized than those issued during the weekend. The net result is that there are

ISSN: 0975-5665, Nov, 2013 www.ijrct.org

many more than one CRL current at any time and clearly not all users and PKI enabled applications will have the same most recently downloaded CRL cached.

1.1.5 Hybrid CRL options

In addition to the delta-CRL, over-issued CRL, and DP or segmented CRL models, these models can be mixed. One could mix the mechanisms above to get the over-issued delta-CRL, over-issued DP-CRL, and segmented (DP) delta-CRL models. These combine, to varying degrees of effectiveness, to take advantage of each model' s advantages. These variants will not be explored further in this paper, but the interested reader is referenced to Årnes [2] for an overview of the options.

These are circumstances where CRL is an inadequate mechanism for cert revocation notification. In cases where CRLs are inappropriate for checking certificate status OCSP offers a better choice.

2. Online Certificate Status Protocol (OCSP)

OCSP addresses are some of the shortcomings of CRLs. They offer a real-time mechanism for certificate status checking. An end host can query the OCSP server when a cert is presented to find out if the certificate has been revoked. This resolves many of the issues that arise from the use of CRLs, but some other problems may appear from the use of OCSP. Some OCSP servers still use the CRL published by a Certification Authority to advise clients on the revocation status of a digital certificate, whereas other OCSP servers integrate tightly enough with the PKI to be able to query the certificate database directly for certificate revocation status. When crypto peers need to check the revocation status of certificates they transmit a query to the OCSP server with the serial number of the certificate in question. The OCSP server examines its copy or copies of the CRL to determine if the Certification Authority has listed the certificate as being revoked and replies with a message to the crypto peer that the certificate's status is "revoked", "good", or "unknown". This dialogue between the crypto peer and the OCSP server will consume less bandwidth than all, but the smallest of CRL downloads. It also consumes no memory on the crypto peer, as it will not have to cache the CRLs. In cases where an OCSP server relies on the CRL, the Certification Authority must only publish the CRL for the OCSP server's use. This will allow CRL to be updated on a more frequent interval and to offer a more "real-time" certificate revocation status, without consuming large quantities of network bandwidth with frequent, large CRL downloads, to all the cryptographic peers in a network. If the OCSP server integrates directly with the PKI to have immediate access to certificate revocation information, cryptographic peers will receive an immediate response to certificate revocation status any time they query the OCSP server.

ISSN: 0975-5665, Nov, 2013 www.ijrct.org

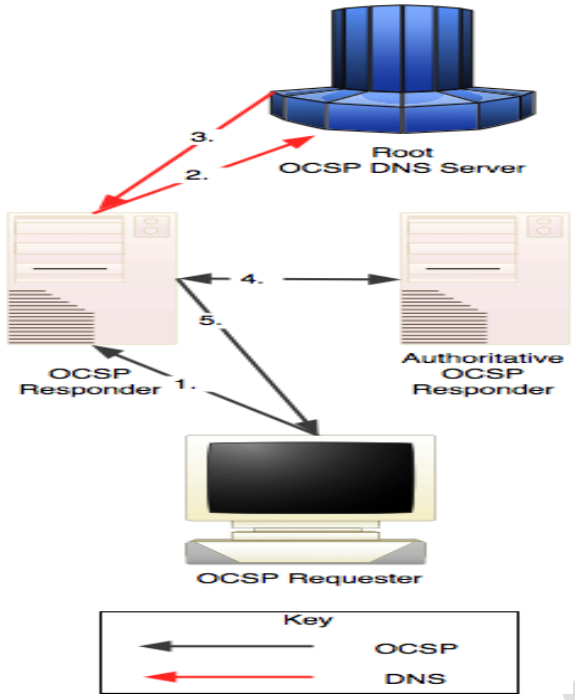

Figure.2. OCSP Requester-Responder with Root DSN Server

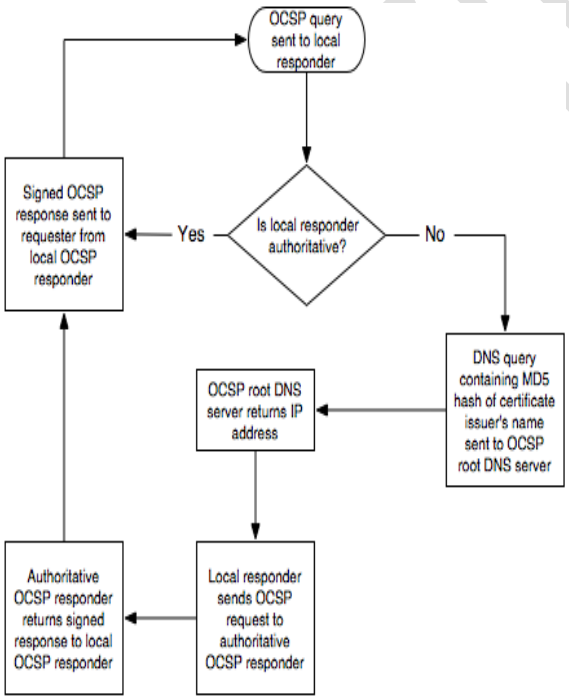

Figure.3. OCSP Requester-Responder Paradigm

OCSP is an IETF standard that enables real-time and low overhead certificate validation. The Online Certificate Status Protocol (OCSP) supplements CRL validation, and enables high-performance validation of certificate status. Further, an OCSP server can retrieve the CRLs from all CAs in an organization. Upon implementation, an organization can use an OCSP server as a single point of contact for revocation validation. The protocol is provided by an OCSP responder (a server) in which real-time revocation information is available through a request/response mechanism. This enables client applications to obtain timely information on the revocation status of a certificate. The OCSP responder maintains the status of certificates either by direct interaction with the CA or by caching information in its repository. If a responder does not hold the status information for a requested certificate, OCSP provides alternative methods to resolve the query. Responders can be configured to query the CA upon request for on-line information, or to query other responders holding cached certificate status information until the status of the requested certificate is found. The requester suspends acceptance of the certificate until a status is reported. An OCSP server works using a Responder-Repeater configuration.

Typically, an enterprise would configure a single OCSP Responder and multiple OCSP Repeaters. In this way, OCSP enables an end entity to request certificate status information about one or more certificates through a single OCSP request. The request contains the OCSP version, the service request type, and one or more certificate identifiers (since bulk

ISSN: 0975-5665, Nov, 2013 www.ijrct.org

requests are supported [9]). An OCSP responder checks the revocation status information for the requested certificate(s) against the information currently stored in its repository, then issues the end entity a digitally signed response. The response contains the certificate identifier, the certificate status, and a validity period for the response [10]. The OCSP response is digitally signed to authenticate the responder as a trusted entity and ensure the integrity of the response. Additionally, OCSP supports an option for digitally signing the request. The protocol is described in RFC 2560 [11]. Various implementations exist in form of add-ons for CAs and general verification servers (targeting at VAs).

3.Analysis

This is a typical PULL-model allowing the client to control when and how much information it gets. Its performance is fine as long as the amount of requests is not too high for one single server. A simple check without extended information will only involve two messages and provide exactly the desired information. An OCSP-server may soon become a bottleneck if it is the only access point for a CA. Thus, multiple OCSP-server should be able to fulfill requests for certificates from a certain CA which will require replication. Note that the RFC 2560 states nothing about the mechanism revocation information gets from the CA to the OCSP-server. This has to be done using another approach and incorporating other protocols. Most recent implementations support CRLs or CRL-derived methods for doing this. OCSP seems the right approach into the direction of an infrastructure providing access points who can be queried about certificate revocation information. Here, the protocol would fit perfectly.

4.Alternate Method

On-line methods of revocation notification may be applicable in some environments as an alternative to the X.509 CRL. On-line revocation checking may significantly reduce the latency between a revocation report and the distribution of the information to relying parties. Once the CA accepts the report as authentic and valid, any query to the on-line service will correctly reflect the certificate validation impacts of the revocation. However, these methods impose new security requirements; the certificate validator shall trust the on-line validation service while the repository does not need to be trusted. One of the online method that is gaining popularity is the Online Certificate Status Protocol (OCSP)[5]. It specifies a protocol used to determine the current validity status of a certificate online. OCSP is designed for X.509 certificates but may also work with other kind of certificates. The protocol can be used instead of or even together with CRLs if more timely information about the status is required. Information about the way to obtain a certificates status can be included within the extension fields of a X.509-certificate.

The protocol is applied between a client (OCSP requester, acting for the user) and a server (OCSP responder, representing a directory). The client generates a so called OCSP request that primary contains one ore even more identifiers of certificates queried, i.e. their serial number together with other data. Then, the (optionally signed) request is send to the server. The server receiving the OCSP request creates an OCSP response: Since all syntactical and content checks

succeed, the response mainly includes a timestamp representing the time when the actual request is generated, furthermore, the identifiers and status values of the requested certificates together with a validity interval. A certificate status value is either set to good, revoked or unknown. Be aware that ”good” implies three meanings: firstly, the certificate is not revoked, but secondly, it may also not be issued yet or even thirdly, the time at which the response is produced is not within the validity of the certificate. Status ”revoked” stands for a revocation or onhold of the certificate. If the answer is ”unknown” the server has no information available about the required certificate. The validity interval specifies the time at which the status being indicated is known to be correct and optional the time at or before newer information will be available about the status of the certificate. The OCSP response should be digitally signed either by the server or by the CA. In case of any error the OCSP response contains an error message. The OCSP response is send to the requesting client of the user who then analyzes the data. Formats of request and response are due to the transmission protocol e.g. HTTP or LDAP. Depending on proper defined time schedules, OCSP provides more timely status information than any other method. A preproducing of signed responses is currently optional. OCSP is especially appropriated for attribute certificates where status information always need to be up-to-date. In the practice, the caching of HTTP-browsers must be handled carefully.

5.Conclusion

As we have, there exist many ways to achieve certificate revocation, but none of them is perfect in all respects. There are some vulnerabilities to all schemes, and considerations for implementing a PKI with revocation capabilities. For our proposed scheme, we found that OCSP was the best choice , mainly because of the low bandwidth consumption and the availability of implemented products. However, OCSP has its disadvantages in that it is vulnerable to replay and denial-of-service attacks, and needs a large overhead at the responder’s part due to the many signature generations. We believe that a hybrid solution may be wise, in order to mitigate some of the vulnerabilities.

6.References

[1] Richard C. Ankney. Certificate Revocation Mechanisms. White paper from CertCo Inc. Available from from: http://www.certco.com/pdf/revoc.pdf.

[2] Alfred W. Arsenault and Sean Turner. PKIX Roadmap. Internet Draft, available at http://www.ietf.org/internet-drafts/draft-ietf-pkix-roadmap-06.txt. Expires in May, 2001.

[3] Paul Ashley and Mark Vandenwauver. Practical Intranet Security – Overview of the State of the Art and Available Technologies. Kluwer Academic Publishers, 1999.

[4] Mihir Bellare, Oded Goldreich and Shafi Goldwasser. Incremental Cryptography and Application to Virus Protection. Proceedings of the 27th Annual ACM Symposium on the Theory of Computing, pp 45-56, 1995.

[5] Marc Branchaud. A Survey of Public Key Infrastructures. Thesis, McGill University, Montreal, Canada, 1997.

[6] David A. Cooper. A Model of Certificate Revocation. Proceedings of the Fifteenth Annual Computer Security Applications Conference, pp 256-264, 1999.

[7] DISA/NSA. Public Key Infrastructure Roadmap for the Department of Defense, June 14, 1999.

[8] Carl Ellison and Bruce Schneier. Ten Risks of PKI: What You're not Being Told about Public Key Infrastructure. Computer Security Journal, vol. XVI, no. 1, pp. 1-7, 2000.

[9] Jalal Feghhi et al. Digital Certificate: Applied Internet Security. Addison Wesley Longman, Inc. Reading, MA, 1999.

[10] Michael Myers et al. X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP. The Internet Society,1999. Available from http://www.ietf.org/rfc/rfc2560.txt?number=2560.

[11] Ronald L. Rivest. Can We Eliminate Certificate Revocation Lists? Proceedings of Financial Cryptography: Third International Conference, pp 178-183, 1998.

