Efficient Certificate Revocation : A P2P Approach

Chu Yee Liau *Stéphane Bressan* *Kian-Lee Tan*
Department of Computer Science
National University of Singapore
{liaucy,steph,tankl}@comp.nus.edu.sg

***Abstract*— Certificate revocation is one of the many challenges faced by Public Key Infrastructure (PKI). Certificate revocation is the action of declaring a certificate, which has not expired, is no longer valid due to various reasons ranging from change of relationship between certificate issuer and the public key owner to compromised private keys of the associated certificate to change of information contained in the certificate. All the revoked certificates by the certificate issuer must be made available to all the end-entities, which need to verify a certificate. Many schemes have been proposed for certificate revocation; each with its own strenghts and weaknesses. Some of these schemes, although straightforward and easy to implement, suffer when faced with the challenge of efficient distribution of certificate revocation information. In this paper we look into the use of Peer-to-Peer (P2P) technology to effectively and efficiently distribute the revoked information. P2P is an emerging paradigm that is now viewed as a potential technology that could re-formulate well known distributed architectures (e.g., the Internet). It is a network architecture in which all participating computers (or nodes), in most cases, have equivalent capabilities and responsibilities. Certificate revocation schemes such as Certification Revocation Lists, which has the potential to distribute very large list, will definitely benefit from the P2P implementation.**

I. Introduction

Internet has brought tremendous business advantages to the business world. However, being a public network, it is hard to control how a piece of information transmits from one computer to its desired destination over the Internet. As such it is insecure to transmit sensitive information through Internet.

In order to have a secure network, cryptography techniques such as symmetric key encryption and public key cryptography are being employed. While symmetric key encryption is well developed and efficient, it has problem in key distribution. Public key cryptography, being computational intensive in nature, is less efficient in encrypting large messages. Nevertheless, public key cryptography manages to solve the key distribution issue face by symmetric key encryption. The combination of the two has provided a medium for secure communication over a public network such as Internet. This secure medium provides confidentiality, authenticity, integrity and non-repudiation through encryption and digital signature. Confidentiality prevents information to be known to unintended party while transmitting over the Internet. Integrity makes sure that the information transferred is not tampered by others. Authenticity refers to the need for the receiver to be assured that the data truly came from the alleged sender. Non-repudiation provides undeniable evidence of an interaction between two parties.

However, with the public key cryptography alone is not enough as there is no trust. Therefore, technology and standards such as Public Key Infrastructure (PKI) has emerged as the foundation for trusted secure transaction over the networks. The primary role of PKI is to establish identities that can be trusted. This is done through the use of public key certificate in the form of digital certificate. A public key certificate associates a public key its owner and it is signed by a trustworthy party [1]. In addition to public key, the certificate will contain other information of the certificate owner such as name, address, etc. Besides, some information about the certificate, such as version, serial number, validity period, etc., is also included in the certificate.

Certificate revocation is one of the challenges faced by PKI. Certificate revocation is the action of declaring a certificate, which has not expired, is no longer valid due to some reasons. A certificate can be revoked due to reasons such as; change of relationship between certificate issuer and the public key owner, the private key of the associated certificate is compromised or simply change of information contain in the certificate. All the revoked certificates by the certificate issuer must be made available to all the end-entities, which need to verify a certificate. There are many schemes proposed for certificate revocation. Each of these schemes has their strengths and weaknesses.

Some of these schemes, although straightforward and easy to implement, faces the challenge of efficient distribution. In this paper we will look into using peer-to-peer architecture to effectively distribute the revoked information. Peer-to-peer (P2P) technology [2], [3], [4], also called peer computing, is an emerging paradigm that is now viewed as a potential technology that could re-architect distributed architectures (e.g., the Internet). It is a network architecture in which all participating computers (or nodes) have equivalent capabilities and responsibilities. Certificate revocation scheme such as Certification Revocation Lists, which has the potential of having to distribute very large list, will definitely benefits from P2P implementation.

II. Certificate Revocation Schemes

Before we start discussing about some of the certificate revocation schemes, it is good to look at some of PKI components and their definitions. The PKI model that we will be looking at is the PKIX model which makes use of X.509 standards [5]. The X.509 standard defined certificate formats and fields, and procedures for distribution of public keys as well as certificate revocation.

- Certification Authority (CA) - CA is responsible for creating and issuing end-entities certificates. CA is also responsible for management of all aspects of the life cycle of the certificate after its issuance. This includes tracking of certificate status and issuing certification revocation notices for its revoked certificates. CA is a trusted entity.
- Registration Authority (RA) - is an optional component within a PKI. If RA is not exist, its role will be taken care by CA. The main function of RA is the administrative tasks associated with registering the end-entity (i.e. the subject of the certificate issued by CA).
- Repository - it is sometimes being referred to as Directory. It provides storage for certificates and revocation notices issued by CA. Depending on revocation schemes, a repository can be a trusted or non-trusted entity.
- End-entity - is the user of PKI certificates and/or end user system that is the subject of a certificate.

In the PKIX model, a CA is required to maintain information about the status of a certificate. This includes the support of certificate revocation. As describe in previous section, certificate revocation is the process where a CA declares the non-expired certificate as invalid due to certain reasons. In this section, we shall describe some of the schemes proposed and implemented for the purpose.

A. Certificate Revocation List (CRL)

In this method, a list of revoked certificate, known as certificate revocation list (CRL), which is within its validity period is periodically generated by CA for its domain. The main problem of this method is when there is a large domain being involved, the list can grow to huge size because the number of revoked certificate is usually proportional to the size of the domain. When the size of the CRL becomes larger more network load will be put on the network and server when the end-entities download the list.

It is a practice for the end-entities to cache the CRL for as long as the CRL remains valid. However, the frequency of list updates is limited in such a way that the list obtained may not always be fresh.

B. Delta CRL

The traditional CRL scheme is being criticized for not being efficient in size and it might not be practical to set the refresh period of the CRL to a small value. However, small refresh period is needed in order to get the freshness of the CRL. In other words, the shorter a validity period of a CRL, the more up-to-date it reflects. In order to overcome this limitation, delta CRL is introduced.

Delta CRL works as an extension to CRL. As the name suggests, it is a list of incremental changes that have occurred since the last complete posting of CRL. Delta CRL is digitally signed by the CA. A delta CRL is periodically updated and it serves as the update to the previous complete CRL and not the previous posted delta CRL. In this scheme, full CRL is cached on the End-Entities and referred to as base posting, while the delta CRLs are considered incremental postings. The newest revocation information is obtained from the newest base CRL posting and the newest Delta CRL.

C. CRL Distribution Points

This approach extends the CRL scheme by addressing the maximum size of a CRL. The size of a CRL subject population is limited by dividing the total population for a CA into a number of segments. This approach is sometimes known as segment CRL.

Each segment in the segmented CRL is associated with a CRL distribution point, which can be located on different hosts and/or directories on the same host. Each certificate has a pointer to the location of its CRL distribution point, therefore, there is no need to either search through distribution points or have a priori knowledge of revocation information locations.

III. Proposed System

As describe earlier, CRL contains a list of certificates which are still within their expiration date but has been revoked due to certain reason. The revocation scheme that use CRL to distribute certificate status information can be divided into few steps. In the first step, CRL is being generated periodically by a CA for all its revoked certificates. Each CRL has two dates field named thisUpdate and nextUpdate. A CRL is said to be valid if the current date is greater or equal to thisUpdate and smaller than nextUpdate date. A CRL that is no longer valid cannot be used to validate certificate.

In the second step, the CA will send the generated CRL to a repository where all the end-entities have access. Since version 2 of X.509 implementation, CRL Distribution Point has been added to improve the performance by reducing the size of CRL serve by each repository. The third step involves the end-entities querying the repository for a CRL.

In the following discussion, we assume there is no pre-caching of CRL, in other words, the end-entities will only query the CRL repository when it needed to validate a certificate. The second assumption is that once the CRL is obtained from a repository, it is being cache in local directory. Further validation will refer to this cache until it expired before query the CRL repository again.

The existing CRL scheme is very much burdened by the size of the CRL especially for those CA that has

Environment	*Size*	*Validation Rate*	*Validation Period*	*Revocation Rate*
A	100	5/day	2 weeks	10%/year
B	1000	100/day	60 min	10%/year
C	10000	25/day	30 min	10%/year

TABLE I
SIMULATED ENVIRONMENTS

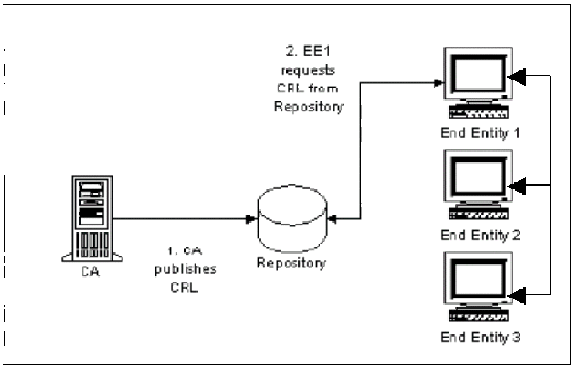

Fig. 1. Architecture

Algorithm *GetCurrentCRL*

```
1.  Check local cache for current CRL
2.  If (no CRL or not CRL valid) then
3.      Check for current CRL in direct peers
4.      If (no valid CRL in peers) then
5.          Get current CRL from repository
6.      Else
7.          use CRL from peer
8.  Else
9.      use CRL in local cache
```

Fig. 2. Algorithm for GetCurrentCRL

large domain. The big file size for CRL will increase the network delay and hence causing the end-entities to wait for longer time. Secondly, when the number of relying end-entities is big, the query that has to be served by repository will be very high.

In our proposed model, we proposed a peer-to-peer collaboration in the end-entities. As shown in figure 1, the architecture of our model is still the same as existing CRL scheme except the collaboration among the end-entities. The procedure for getting current CRL in certificate validation by an end-entity in our proposed model is shown in figure 2.

The procedure is different from existing CRL in that there is extra query to the direct peer for current CRL. Scheme that uses CRL do not need to have trusted repository/directory because the CRL is digitally signed by CA and therefore, the end-entities have to trust the CA alone (i.e. CRL scheme needs only two-party trust model between CA and end-entity). Therefore, trusted relationship between peers is also not necessary.

As seen in the GetCurrentCRL procedure, every hit of valid CRL on local cache and direct peers will reduce the access make to repository and therefore reduced the processing over at repository.

IV. EXPERIMENTS

In order to study the performance of the proposed system, we have run some experiments through simulation. We have used the simulator mentioned in [6] with some modification to include the end-entity collaboration. In the paper, three simulated environments were described in Table I. For details of the simulation setup, please refer to [6]. We used the three simulated environments described in the paper for our performance study on the original CRL scheme and our proposed schemes with 5 and 10 peers. Figures 3,4,5 show the result of the simulation for environment A, B and C respectively.

As shown in the figures, our proposed model performed better in all environments compare to the original CRL scheme that does not utilise collaboration between end-entity. We also observed that for CRL with longer validity period (environment A), the performance difference between 5 peers and 10 peers of our proposed model is negligible. However, for shorter CRL validity period (environment C), the setting with 10 connected peers reduces the number of requests to repository significantly as compared to 5 connected peers and the original scheme.

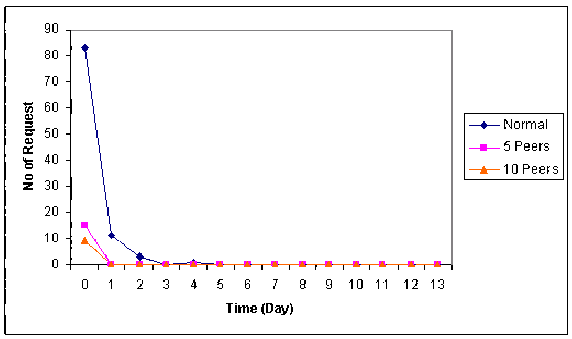

Fig. 3. Environment A

V. CONCLUSION

In this paper we looked into the use of Peer-to-Peer (P2P) technology to effectively and efficiently distribute the revoked information in PKI. P2P is an emerging paradigm that is now viewed as a potential technology that could re-formulate well known distributed architectures (e.g., the Internet). It is a network

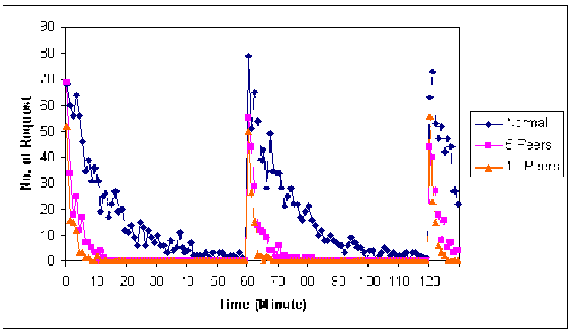

Fig. 4. Environment B

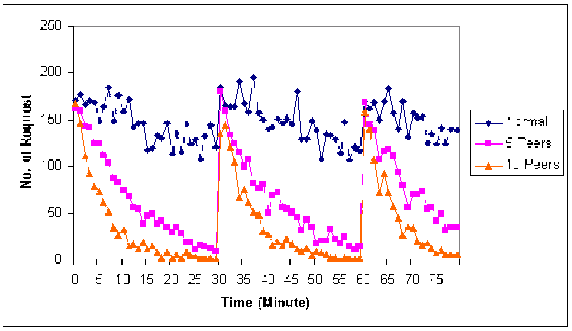

Fig. 5. Environment C

architecture in which all participating computers (or nodes), in most cases, have equivalent capabilities and responsibilities. Certificate revocation schemes such as Certification Revocation Lists, which has the potential to distribute very large list, will definitely benefit from the P2P implementation.

For our future work, we will implement the proposed model on top of our BestPeer platform [4]. The proposed model will be implemented as a middle-ware and is tranparent to the user.

VI. Acknowledgement

This research is partially supported by a grant from Agency of Science and Technology (ASTAR) Singapore.

References

[1] B. Schneier, "Applied cryptography, 2nd ed." in *John Wiley and Sons*, 1995.

[2] Gnutella, "The gnutella protocol specification v0.4, june 2001," http://www.clip2.com/GnutellaProtocol04.pdf.

[3] I. Clarke, O. Sandberg, B. Wiley, and T. Hong, "Freenet: A distributed anonymous information storage and retrieval system," in *Proc. of the ICSI Workshop on Design Issues in Anonymity and Unobservability*, 2000.

[4] W. S. Ng, B. C. Ooi, and K.-L. Tan, "Bestpeer: A self-configurable peer-to-peer system," in *ICDE*, 2002.

[5] R. H. et al, "Internet x.509 public key infrastructure certificate and crl profile, ietf internet draft," http://www.ietf.org/internet-drafts/draft-ietf-pkix-new-part1-11.txt, 2001.

[6] A. Arnes, M. Just, S. J. Knapskog, S. Lloyd, and H. Meijer, "Selecting revocation solutions for pki," in *NORDSEC*, 1995.