An Efficient Certificate Revocation and Verification Scheme from Multi-Hashing

Mengbo Hou
School of Computer Science and Technology, Shandong University
Key Laboratory of Software Engineering, Shandong Province
Ji'nan, 250101, China
Email: danny0128@126.com

Qiuliang Xu, Fengbo Lin
School of Computer Science and Technology, Shandong University
Key Laboratory of Software Engineering, Shandong Province
Ji'nan, 250101, China
Email: {xuqiuliang , linfb}@sdu.edu.cn

***Abstract*—Even though Public Key Infrastructure (PKI) and X.509 certificate has been a prominent security model for a variety of e-commerce applications and large scale distributed computing, it has not been sufficiently investigated in the certificate revocation and verification mechanism. In this paper, we discuss the need and importance of certificate revocation and verification, and analyze the limitations of several certificate validation schemes that are widely used in PKI environments. Then we propose an alternative scheme. The underlying idea is that the certificate holder provides certificate validation proof (CVP) to the verifiers in manner of initiative. According to this scheme, The CVP is a proof issued by a trusted third party (TTP) for the certificate stating whether it was revoked or not. For both parties in any transaction, the certificate holder provides the CVP to the verifier, the verifier knows about the validity status of the certificate by verifying CVP efficiently without any extra information except the certificate. The CVP is created by multi-operations with a HASH function and operations are associated with the current time. The suggested scheme is principally simple with characteristics of distributed processing, high security, low communication costs and good practicability.**

***Index Terms*—Public Key Infrastructure, X.509 certificate, certificate validation, hash function**

I. INTRODUCTION

Security is essential for the sensitive message communication over the widely used Internet. As more and more security infrastructures have been developed, Public Key Infrastructures (PKIs) [17] gained a considerable attention as they seem to hold a promising foundation for secure electronic commerce [2], Grid computation [3], Ad Hoc network [7,20,21,14,16,23-25] and cloud computing *et al*. With cryptographic primitives, such as asymmetric encryption, symmetric encryption, hash function, and message authentication code (MAC), PKI provides data confidentiality, data integrity, authentication and non-repudiation for applications. The wide use of public key cryptography requires the ability to verify the authenticity of public keys. This is achieved through the use of digital certificates to serve as a mean for transferring trust, such as X.509 certificate [4] or PKIX certificate [5]. A digital certificate is a message signed by a publicly trusted third party - Certification Authority (CA) , which includes a subject entity identity, subject public key and additional information, such as serial number, issuer identity, expiration date, and information regarding the key and the subject entity.

When a digital certificate is issued, its validity is limited by a starting date and an expiration date. However, there are circumstances where a certificate must be revoked prior to its expiration date, such as when a private key is revealed, subject's identification data is modified, or subject's affiliation or position is changed. Thus, the verification process of a digital certificate is a necessary but not sufficient evidence for its validity, and a mechanism is needed for determining whether a certificate was revoked. Certificate revocation [5,18,19] is the act of invalidating the association between the public key and attributes embodied in a certificate. However, certificate revocation is inherently difficult [8]. No solution has been found that meets the timeliness and performance requirements of all applications and environments. Certificate revocation is an important business concern to service providers and well as the users of the authentication service.

In a typical PKI environment, in order to deal with the problem of certificate validation, two cases are concerned: certificate revocation and certificate verification. A certificate revocation and verification scheme needs to be fast, efficient, timely and particularly appropriated for large infrastructures. Due to that, it is necessary to reduce the number of time-consuming

Corresponding author : Qiuliang Xu;
Manuscript received April 25, 2011; revised June 11, 2011; accepted July 5, 2011.

doi:10.4304/jcp.7.6.1437-1444

calculations like verification processes of a digital signature and to apply other mechanisms, or to minimize the amount of data transmitted. It is also desirable that a method provides suspending a certificate temporarily and also a reuse.

In this paper, we suggest an alternative scheme called CVS-MH for certificate revocation and certificate verification by adding two extra certificate extensions in the certificate generated by the CA, and adding a component called Certificate Validation Proof Server (CVPS) to the classical CA architecture (deployed as an important component of RA system). The CVPS server acts as the ticket server, to provide Certificate Validation Proof (CVP) which indicates the validity status of the current certificate being used. The CVP is generated by the computation of multi-HASH operations, and can be verified easily and efficiently. The new scheme is principally simple with characteristics of distributed processing, high security, low communication costs and good practicability.

The rest of this paper is organized as follows: In section II, we briefly review several proposed schemes in the literature, such as the Certificate Revocation List (CRL) [5], Online Certificate Status Protocol (OCSP) [9], Certificate Revocation Tree (CRT) [10], and Certificate Revocation System (CRS) [11]). Our new suggested scheme (called CVS-MH) and its analysis of security and performance are described in detail in section III. Finally, in section IV, we give the conclusion.

II. Discussion of Several Certificate Validation Schemes

A. Certificate Revocation List (CRL)

Currently, the most widely accepted standard for certificate revocation is the Certificate Revocation List (CRL) [5]. A CRL is a signed list issued by the CA identifying all revoked certificates by their serial numbers. The list is concatenated with a timestamp (as an indication of its freshness) and signed by the CA that originally issued the certificates. The CRL are sent to the directory on a periodic basis, even if there are no changes, to prevent the malicious replay of old CRL instead of new CRL. A user that wishes to check the validity of a certificate must obtain the most recent CRL and make sure that the serial number of the certificate in question does not appear on the list.

The use of CRL to convey revocation status in public key infrastructures has long been the subject of debate. The main advantages of the scheme are simplicity, easy to implement and deploy, while it has several disadvantages.

1) The cost of CRL management and distribution is too high. Because of the periodical distribution of CRL and potential size of CRL (the CRL may get very large), scaling to large communities can be difficult.

2) CRL are inappropriate for transactions that require real-time revocation status.

3) The CRL distribution period is very hard to make certain: although the verifiers can reduce the communication costs by long distribution period, the verifiers will be at the risk of security, because the most recent CRL may exclude the newly revoked certificates. If else, the verifiers would be compelled to update the local CRL more frequently to keep the list fresh at the client end with high communication costs.

4) CRL do not provide a positive response. Because CRL only identify revoked certificates, the existence of a (non-revoked) certificate cannot be determined solely from validity information.

B. Online Certificate Status Protocol (OCSP)

In order to overcome the limitation inherent to the CRL schemes, several approaches of online certificate validation have been proposed [9, 1, 6]. The most widely used of these is the Online Certificate Status Protocol (OCSP) [9]. OCSP allows CA to set up responders that can, when given a certificate identifier, respond with either “good”, “revoked”, or “unknown”. The response is signed by the issuing CA, a CA-designated responder, or a responder whose public key is trusted by the requester. The protocol specifies the syntax for communication between the server (which contains the certificate status) and the client application (which is informed of that status). OCSP overcomes the chief limitation of CRL: the fact that updates must be frequently downloaded to keep the list fresh at the client end. It makes verifying certificates happen in a rapid, online fashion.

There are still some problems existing in OCSP scheme:

1) The requester must know the proper OCSP responder to query in advance. This information, like the CRL distribution points, can be included in the certificate, but often is not.

2) The responder needs to know about the certificate in question as well as the signing authority. Because an OCSP responder will only have knowledge of a few certificate authorities, OCSP is impractical for validating certificates issued by multiple authorities.

3) Wide-band network and high performance OCSP server are required to ensure the speed of requests and responses.

4) If the responder is centralized, it is vulnerable to Denial of Service attack.

5) Compromise of responder’s private key affects the entire system.

C. Certificate Revocation Tree (CRT)

Kocher [10] suggested the use of Certificate Revocation Trees (CRT) in order to enable the verifier of a certificate to get a short proof that the certificate was not revoked. A CRT is a hash tree with leaves corresponding to a set of statements about certificate

serial number *n* issued by a CA. The set of statements is produced from the set of revoked certificates of every CA. It provides the information whether a certificate *n* is revoked or not (or whether its status is unknown to the CRT issuer). There are two types of statements: specifying ranges of unknown CAs, and, specifying certificates range of which only the lower certificate is revoked. To produce the CRT, the CRT issuer builds a binary hash tree with leaves corresponding to the above statements. A proof for a certificate status is a path in the hash tree, from the root to the appropriate leaf specifying for each node on the path the values of its children.

The main advantages of CRT over CRL are that the entire CRL is not needed for verifying a specific certificate and that a user may hold a succinct proof of the validity of his certificate. The main disadvantage of CRT is in the computational work needed to update the CRT. Any change in the set of revoked certificates may result in re-computation of the entire CRT.

D. Certificate Revocation System (CRS)

Micali [11] suggested the Certificate Revocation system (CRS) in order to improve the CRL communication costs. CRS is intended to increase performance by using a more compact data structure than the full lists in CRL based systems. The underlying idea is to sign a message for every certificate stating whether it was revoked or not, and to use an off-line/on-line signature scheme to reduce the cost of periodically updating these signatures. The directory is updated daily by the CA sending this signature for each certificate.

The advantage of CRS over CRL is in its query communication costs. Although the daily update of the CRS is more expensive than a CRL update, the cost of CRS querying is much lower. The main disadvantage of this system is the increase in the CA-to-directory communication. Moreover, since the CA's communication costs are proportional to the directory update rate, CA-to-directory communication costs limit the directory update rate.

A few criteria have been discussed in literature [12, 13] to establish the metrics with which various revocation approaches can be analyzed. These include:

1) Population Size

The absolute size of the number of potentially revocable certificates can strongly influence the approach taken. A solution intended to address a large population may require more resources and complexity as compared to a smaller group.

2) Acceptable Latency in Revocation

The degree of timeliness relates to the interval between when a CA made a record of the revocation and when it made the information available to the relying parties. A more eager mechanism to update and convey this information will proportionally consume more bandwidth. Moreover, if the interval is small, there might not be anything new to update and most of the bandwidth might be used for passing redundant information.

3) Connectivity.

Does the relying party need to be online in order to ascertain the reliability? Online mechanisms create mission critical components in the overall security design. This dimension of the problem can inform the designers of online mechanisms of the need to facilitate off line caching of prior data.

4) Security Considerations.

In a majority of cases, a certificate will expire without ever being had to be revoked. One of the most troubling scenarios would be the compromise of the private key. Without an effective compromise, a security solution based on PKI is at a risk of general system compromise.

It has been argued at length that CRL is both semantically and technically inferior to other approaches [15], and then he asked whether CRL could, and should, be eliminated in favor of other mechanisms. In most cases, the answer seems to be “yes", and suggested some possible replacement mechanisms. According to his underlying idea, we proposed a new scheme that the certificate verifier can easily make sure of the certificate validity status by the certificate holder showing proper proof directly.

III. OUR ALTERNATIVE: CVS-MH

A. The requirements to design efficient scheme

In any scheme of certificate revocation and verification, the key problem is the generation of proof which indicates the validation of the dedicated certificate, how the verifier can acquire proof of certificate validation, and the proof should be publicly verified efficiently. Generally, such a proof is issued by a third party that is reliable and trusted to the end entity and the relying party. The proper scheme should satisfy the characteristics described below:

1) Easy access of certificate validation proof

The certificate validation proof can be provided by the certificate holder directly or the relying party acquires in manner of initiative. In almost all of the previous proposed schemes, the proof is provided by a thirty party and the verifier should acquire the proof by actively investigation. The later one has some disadvantages, such as in a Client/Server environment, if the server wants to verify the validity of client certificate, it will endure a great burden to acquire the proof by means of other mechanisms, such as query directories or query OCSP server.

2) Low communication cost for certificate validation proof acquirement

With low communication costs, such as small proof data or low proof-querying frequency, the certificate holder and certificate verifier can easily acquire it without any strict demand to the band-width of network.

3) Succinctness and completeness of certificate validation proof

The verifier should be able to determine the certificate validity status efficiently without extra information except the certificate itself and the certificate validation proof, the certificate validation proof is succinct and completeness, so that the processing performance of the verifiers can be much more efficient.

4) Low computational costs for the verifiers to verify the certificate validation proof

The verifier should get verification result quickly with fewer computational costs, so to improve the processing speed markedly. It is extremely essential in the case that the server behaving as a verifier.

5) Low costs for the trusted third party to generate the certificate validation proof

The publicly available trusted third party is responsible for the generation of the certificate validation proof for the entire certificate holders, so the costs should be considered seriously both in computations and communications.

B. Description of the proposed scheme

The proposed CVS-MH scheme, which was inspired by the idea of Micali [22], is based on the typical PKI with some modification to the architecture. It can be deployed along with the existing PKI system. The underlying idea is that the certificate holder provides Certificate validation proof (CVP) to the verifiers in manner of initiative. The CVP is a proof issued by a trusted third party (TTP) for the certificate stating whether it was revoked or not. For both parties in any transaction, the certificate holder provides the CVP to the verifier, the verifier knows about the validity status of the certificate by verifying CVP efficiently, without any extra information except the certificate. The CVP is created by multi-operations with a hash function; the operations are associated with the current time.

1) System Architecture of CVS-MH

Certificate Authority provides a trusted third party that can vouch for the validity of the credentials of both parties in any transaction. It is based upon open standards of which the most important is X.509 [4,5] and PKIX[17] thus allowing it to work with other CA systems that use X.509 certificates. Its architecture is designed to be modular with components defined in key areas of functionality. At the top of the tree type structure, the CA is central to the viability of the system, responsible for generating, publishing and revoking digital certificates. The Certificate Authority is managed by the CA Operator (CAO) and beneath the CA are Registration Authorities (RAs) which act as the interfaces between the end users and the CA, carrying the burden of enrolments and acting as intermediary for authentication. In turn, the RAs are managed by RA Operators (RAOs). Each CA, CAO, RA, and RAO has its own certificate so that each component of the PKI is able to identify itself with other components and communicate securely. Figure 1 shows an example of a classical Certificate Authority system except for the Certificate Validation Proof Server (CVPS) components.

CVPS components are deployed in our scheme in order to issue user's Certificate Validation Proof (CVP) according to the demand of the sub-security domain users. The demand, here, is a query message, indicate to acquire a proof for the current validity status of user certificate.

CVPS is deployed as an important component of RA system, and is administrated by RAO. For each RA, there is a CVPS deployed. With the protected channel (such as SSL Channel), the CVPS communicates with CA component securely. For the common certificate holder and its relying party in the application, CVP is acquired from the CVPS by anyone at any time. There is no serious security requirement in the communication between the CVPS and the requesters, so the CVP of a certificate can be acquired by either the certificate holder or the relying party. In our scheme, the certificate holder is preferred, because the relying party can hardly know which CVPS to query in advance, while the certificate holder knows about that. Due to the distributed design of CVPS system, high performance and convenience are enabled.

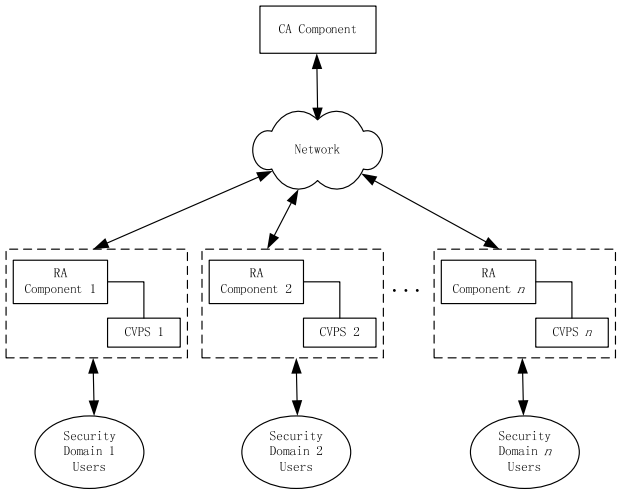

Fig. 1 Architecture of CVS-MH

2) Generation of user Certificate

a) Modification to the X.509 certificate structure

The CA component defines a time interval n, according to the certificate validity period (*e.g.*, with respect to one year for the certificate validity period, define n =365 and increment i represents a

day). Using X.509 certificates, the number of extension fields needs to be extended by two fields (two hash values): *Y* indicates the certificate validity and *N* indicate the certificate invalidity. Because of CA's signature, the authenticity of both values is guaranteed.

b) Generating secrets

The CA choose two pseudo-random number Y_0 and N_0 ($Y_0 \neq N_0$, the length of Y_0 and N_0 is *m*, *m* is a large integer) , for each certificate requester distinctly, kept them secret and send them to the responding CVPS at the RA end in secure manner.

c) Choosing one hash function.

The CA defines a security parameter *l* and constitutes a proper secure hash function *H*(strongly collision-free):

$$H : \{0,1\}^m \rightarrow \{0,1\}^l$$

We define

$$H^1(x) = H(x), \text{ and}$$
$$H^n(x) = H\left(H^{n-1}(x)\right), \text{ for } n = 2,3,\ldots$$

Then the CA calculates as below, and generates certificate for the certificate requester:

$$Y \leftarrow Y_n = H^n(Y_0),$$
$$N \leftarrow H(N_0)$$

3) Revocation and reuse of user certificate

When the certificate holder or the CA wants to revoke certificates for some reasons, proper revocation request message should be submitted to the responding CVPS, the CVPS will revoke the certificate after careful auditing. The revocation operation is merely a symbol marking to indicate the revoked status of the certificate; if the user wants to reuse the formerly revoked certificate, proper reuse request message should be submitted to the responding CVPS, and the CVPS will recover the certificate after careful auditing. The reuse operation is merely a symbol marking to indicate the validity status of the certificate too. Although the revocation and reuse of a certificate are very simple, the authentication of operations should be considered seriously.

4) Generation of users' Certificate validation proof (CVP)

Whenever the end entity wants to communicate with other relying party in secure manner, besides the certificate, a CVP should be provided to indicate the validity status of its certificate. The CVP is generated and acquired from the CVPS in the user's security domain according to the current validity status of user's certificate.

a) If the certificate is currently valid and *i* days have been past from the very beginning of the certificate validity date, the CVPS calculates:

$$CVP_i = H^{n-i}(Y_0)$$

Apparently, we have

$$H^i(CVP_i) = Y$$

b) If the certificate is currently revoked, then the CVPS calculates:

$$CVP_i = N_0$$

Apparently, we have

$$H(CVP_i) = N$$

Theorem 1: Forging of the Certificate validation proof (CVP) is computationally infeasible if the hash function H(x) is strongly collision-free .

In order to prove *Theorem 1*, two Lemmas are introduced:

Lemma 1: *The hash function H (x) is one-way function if the hash function H(x) is strongly collision-free .*

Lemma 2: *The hash function H^n (x) is strongly collision-free if the hash function H(x) is strongly collision-free.*

Below is the proof procedure.

We prove this lemma using mathematics induction.

For *n=1*: $H^1(x) = H(x)$ is strongly collision-free by the hypothesis.

Suppose $H^k(x)$ is strongly collision-free, where *k* is a positive integer. If $H^{k+1}(x)$ is not strongly collision-free, we can get x_1 and x_2 satisfying

$$x_1 \neq x_2$$
$$H^{k+1}(x_1) = H^{k+1}(x_2)$$

Namely

$$H^k(H(x_1)) = H^k(H(x_2))$$

As the result of computing, (x_1, x_2) is a collision-pair of $H(x)$ if $H(x_1) = H(x_2)$, and $(H(x_1), H(x_2))$ is a collision-pair of $H^k(x)$ if $H(x_1) \neq H(x_2)$. This contradicts the hypothesis of induction.

By the principle of mathematics induction, we know that it is computationally infeasible to acquire x_1 and x_2, such that

$$x_1 \neq x_2 \text{ and } H^{k+1}(x_1) = H^{k+1}(x_2)$$

Namely, $H^{k+1}(x)$ is strongly collision-free.

□

Now, let's give the proof of *Theorem 1:*

Suppose the adversary has got the CVP message of day i^{st} *:*

$$CVP_i = H^{n-i}(Y_0)$$

And he wants to forge the CVP message of day j^{st} *:*

$$CVP_j = H^{n-j}(Y_0).$$

We know that

$$\begin{aligned} CVP_i &= H^{n-i}(Y_0) \\ &= H^{(n-j)+(j-i)}(Y_0) \\ &= H^{(j-i)}(H^{(n-j)}(Y_0)) \\ &= (H^{(j-i)}(CVP_j)) \end{aligned}$$

The adversary will have to compute x from y which satisfies $y = H^{(j-i)}(x)$. Due to $H^{(j-i)}(x)$ is strongly collision-free (by *Lemma 2*), $H^{(j-i)}(x)$ is a one-way function (by *Lemma 1*). So the forging of the Certificate Validation Proof (CVP) $H^{n-i}(Y_0)$ is computationally infeasible.

□

5) Operations of the certificate holders

If certificate validity verification is required in any application, the certificate holder can, at any time, acquire its CVP from the CVPS and submit it to the relying party, along with the corresponding certificate in manner of initiative. For example, within a client/server application system with authentication based on certificate that allows the client operators to interact with the server in secure manner, the operator merely acquire the CVP for the first time login, even though he logins for many times in a day, so only one interaction with the CVPS is required. The CVP can be cached locally for all the day until the next day to be refreshed.

6) Operations of the relying party

When the relying party gets the certificate and the CVP of the certificate holder, it verifies the validity of certificate status as following steps (suppose the certificate is not out-of-date):

The relying party computes the number of days between the issuing day and current day, noted as *i,* then computes $Y' = H^i(CVP)$. If $Y' = Y$ (Y is got from the certificate), then the certificate is valid (not revoked); If $H(CVP) = N$ (N is got from the certificate), then the certificate is revoked before now.

C. Analysis of the security and performance

Below, the security issues of the proposed scheme are discussed:

1) The adversary can neither modify Y and N in the certificate, nor reveal Y_0 and N_0 from Y and N. For Y and N, they exist as the certificate extensions and were protected by the signature of the CA. Meanwhile, Y_0 and N_0 are private, randomly generated and occupied securely by CA and CVPS.

2) If the certificate of the dedicated end entity has been revoked, the certificate holder can hardly acquire a valid CVP. Even a previous valid CVP was acquired before the certificate revoked, he still could not construct the valid CVP of current time by the old CVPs (by *Theorem 1*), for the HASH function is a one-way function.

3) Although the adversary could acquire the valid CVP of other users from the CVPS easily, he still could not gain any advantage from that.

4) The security of the scheme mainly focuses on the confidentiality and randomicity of Y_0 and N_0. Accuracy of the system time is also a serious concern, especially in the side of the relying parties, for the validation of the verification is time-related. In order to ensure the correctness of verification, the relying party is responsible for acquiring the correct public time . It is easy to do that for the relying party.

And then we analyze the advantages of the new proposed scheme:

1) The certificate holder submits the CVP to the relying party in manner of initiative. The relying party could verify the validity of certificate without any additional information. It is quite suitable for the client/server or browser/web applications. This model extremely reduces the burden for the server to manage the CVPs.

2) The CVP proof is constructed by a small piece of data with the result of hashing operations, so the communication costs are saved.

3) The verification process is totally operations of hash function computations, the computation cost is really small, so high speed can be guaranteed.

4) The CVPS component deployed in the RA system can work efficiently. The CVP requesters to the CVPS are all from the local security domains, and each CVPS is only responsible for the users of its domain, so the service of the CVPS is distributed. Due to the operations of the CVPS are totally hash function computations, so the computational burden is endurable. Additionally, for each CVPS, the CVP requests are periodical (the CVP can be cached to some extent), so the burden of each CVPS is very limited. Due to the distributed architecture of CVPS, it is much more efficient than other centralized schemes.

5) The CA component defines the time interval n according to the validity time. The parameter n is variable(If n represents the number of days, then the certificate holder need only request CVP proof one time for one day. If n represents the number of weeks, then the CVP proof need not to be refreshed

once a week). To some extent, the value of *n* is measurement of certificate verification security level.

6) The certificate holder merely interacts with one of the distributed CVPSs, but not the centralized CA, so the certificate revocation processes is distributed and without the CA's awareness.

7) This new scheme is designed based on the generalized PKI/CA architecture with few modifications to the existing CA systems, so good compatibility is available.

IV. CONCLUSION

In this paper, we firstly introduced several existing schemes of certificate validation in the literature, and analyzed the limitations of each scheme. Then we designed a new scheme according to the idea that the certificate holder providing the certificate validity proof in manner of initiative. For both parties in any transaction, the certificate holder provides the certificate validity proof to the verifier, the verifier knows about the validity status of the certificate by verifying certificate validity proof efficiently without any extra information except the certificate. The certificate validity proof is created by multi-operations with hash function computations; the operations are associated with the current time. At last, detailed analysis is given covering security and performance. The new scheme is principally simple with characteristics of distributed processing, high security, low communication costs and good practicability.

ACKNOWLEDGMENT

This work was supported by National Natural Science Foundation of China (No. 60873232), also by Natural Science Foundation of Shandong Province, China (No. Y2007G37).

REFERENCES

[1] Ambarish Malpani and Paul Hoffman. "Simple Certificate Validation Protocol (SCVP), " Internet Draft, work in progress, IETF PKIX work group, June 2000.

[2] Stefanos Gritzalis, Socrates Katsikas, Dimitrios Lekkas, Konstantinos Moulinos, Eleni Polydorou. "Securing The Electronic Market: The KEYSTONE Public Key Infrastructure Architecture," Computers & Security, Vol.19, No.8, pp.731-746, 2002.

[3] Ian Foster, Carl Kesselman, Gene Tsudik, and Steven Tuecke, "A Security Architecture for Computational Grids," In Proc. 5th ACM Conference on Computer and Communications Security Conference, 1998.

[4] ITU-T Rec X.509 | ISO/IEC 9594-8: Information technology - Open systems interconnection - The directory: Public-key and attribute certificate frameworks, 2001.

[5] Housley R, Polk W and Solo D. "Internet X.509 Public Key Infrastructure Certificate and CRL Profile"[RFC 3280]. IETF PKIX work group, April 2002.

[6] Phillip Hallam-Baker. "OCSP Extensions," Internet Draft, work in progress, IETF PKIX work group, Sept. 1999.

[7] Marianne A. Azer, Sherif M. El-Kassas, Magdy S. El-Soudani, "Certification and Revocation Schemes in Ad Hoc Networks Survey and Challenges," icsnc, pp.17, Second International Conference on Systems and Networks Communications (ICSNC 2007), 2007.

[8] A. Arnes, H. Meijer, S. Lloyd, M. Just, and S. J. Knapskog. "Selecting Revocation Solutions for PKI, " in Proceedings of The Fifth Nordic Workshop on Secure IT Systems (NORDSEC 2000), October 2000.

[9] M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams. "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP" [RFC 2560], June 1999.

[10] P. Kocher. "A Quick Introduction to Certificate Revocation Trees (CRTs)" http://www.valicert.com/company/crt.html.

[11] Micali S. "Efficient Certificate Revocation," MIT Laboratory for Computer Science, Technical Memo 542b, March 1996.

[12] M. Myers. "Revocation: Options and Challenges," In FC'98 Proceedings of the Second International Conference on Financial Cryptography, LNCS(1465), pp.165-171, 1998.

[13] J.K. Millen, R.N. Wright, "Certificate Revocation the Responsible Way," In Proceedings of Computer Security, Dependability, and Assurance: From Needs to Solutions(CSDA'98), IEEE Computer Society, pp.196-203, 1999,

[14] Genevieve Arboit et al., "A Localized Certificate Revocation Scheme for Mobile Ad Hoc Networks, " Ad Hoc Network, Volume 6, Issue 1, pp. 17-31, 2008.

[15] R. Rivest. "Can We Eliminate Certificate Revocation Lists? ", In Proc. Financial Cryptography'98, LNCS 1465, pp.178-183, 1998.

[16] Y.B. Qian, B.H. Cao, etc. "A certificate revocation scheme for space network," WiCOM'09 Proceedings of the 5th International Conference on Wireless communications, networking and mobile computing , pp. 4458-4462 ,IEEE Press, 2009.

[17] http://www.ietf.org/html.charters/pkix-charter.html, Internet Draft, PKIX Working Group, 2005.

[18] B. Fox and B. LaMacchia. "Certificate Revocation: Mechanics and Meaning", In Proc. Financial Cryptology-FC'98, LNCS 1465, pp.158-164, 1998.

[19] M. Noar and K. Nassim. "Certificate Revocation and Certificate Update, "In: Proc. 7th USENIX Security Symposium, pp. 217-228, 1998.

[20] J. Mo, X.M. Wang, "Distributed Certificate Revocation Scheme for Ad Hoc Network, " Computer Engineer, Vol.36(10), pp.149-151, 2010.

[21] H. Zhong, C.X. Xu, Z.G. Qin , "A Distributed Certificate Revocation Scheme for Ad Hoc Networks, " Journal of University of Electronic Science and Technology of China, Vol.36(3) , pp.496-499, 2007.

[22] S. Micali. "NOVOMODO: Scalable Certificate Validation and Simplified PKI Management," In 1st Annual PKI Research Workshop Proceedings, pp.15-25, 2002.

[23] P. Caballero-Gil and C. Hernández-Goya, "Efficient Public Key Certificate Management for Mobile Ad Hoc Networks," EURASIP Journal on Wireless Communications and Networking Volume 2011, pp.01-11, 2011.

[24] Z. Huan, etc. "A Distributed Certificate Revocation Scheme for Ad Hoc Networks," Journal of University of Electronic Science and Technology of China. V.01.36 No.3:496-499, Jun.2007.

[25] H. He, etc. "New Distributed Certificate Revocation Scheme in Ad Hoc Network," Computer Engineering. Vol.34, No.16:180-182, August 2008.

Mengbo Hou is currently working as an associate professor in the school of Computer Science and Technology, Shandong University, Ji'nan, China. He has received the B.S degree from Hangzhou dianzi University of China, the M.S and Ph.D degree from Shandong University of China, both in Computer Science and Technology. He has published nearly 20 papers in the international refereed conferences and refereed journals. His research interests include cryptography, cryptographic protocols and network security**.**

Qiuliang Xu is currently working as a professor in the School of Computer Science and Technology, Shandong University of China since 2001. He has a Ph. D degree in Applied Mathematics (1999) and an M.S degree in Computational Mathematics (1985) from Shandong University of China. He has published nearly 60 papers in the international refereed conferences and refereed journals. His main research interests are public key cryptography including encryption, digital signature, cryptographic protocol etc.

Fengbo Lin is currently working as an instructor in the school of Computer Science and Technology, Shandong University of China. He has received the B.S degree and M.S degree from Shandong University of China both in Computer Science and Technology. He is now perusing the Ph. D degree from Shandong University of China, His research interests include cryptographic protocol, network attack and defense, system and network security.