A View on How to Design a PKI System

Jun Yoshitake

Mitsubishi Electric Corporation

Abstract

Design methods for a PKI (Public Key Infrastructure) system have to be established to build a practical and efficient PKI system. The author has discussed the methods, indicating points such as information contained in a certificate, distribution route of certificates and of CRL's (Certificate Revocation Lists), entity configuration, etc. In this paper, the discussion is focused on information contained in a certificate and on revocation management not to use CRL's. Three criteria for design of information contained in a certificate are proposed, and it is shown that they are also considered to work for access control. A view to revocation management not to use CRL's is also shown.

1. Introduction - Necessity for Design Methods and Previous Works

As the internet[7] is used more and more, PKI (Public Key Infrastructure) gets more and more important as literally infrastructure. There, design methods for a PKI system have to be established to build a practical and efficient PKI system. If it is designed badly, for example, there are possibilities that confidential information is exposed, that certificate management costs much, or that a directory server is overwhelmed by the rush access of CRL (Certificate Revocation List) retrieval, etc.

The author has discussed the methods[10,11], indicating points based on real system design and operation experience such as information contained in a certificate (subject, profile, and exposure issue), distribution route of certificates and of CRL's, entity configuration, the role of a root CA (Certification Authority), thoroughness of policy and flexibility of operation, and certificate profiling freedom of the subordinate CA's under a CA. Figure 1[11] shows the relationship between design work and the first three points above.

It shows:

(1) What entities exist in the application greatly influences the design of entity configuration (which are CA's and which are end entities ?)

(2) Data flow in the application greatly influences design of distribution route of certificates and of CRL's.

(3) Application features influence many design points, especially exposure issue (Can the certificate be used for other applications ? Is everyone allowed to access the directory ?).

(4) Entity configuration and application features influence design of the subject of the certificate (what is a subject strictly? - a person, a credit card number, or ...?)

(5) Subject and application features influence the design of the certificate profile (subject naming, validity period, extensions, ...)

(6) Entity configuration, distribution route of certificates and of CRL's, subjects, and profile influence certificate validation process. In other words, it works to consider what certificate validation process is like in designing a PKI system.

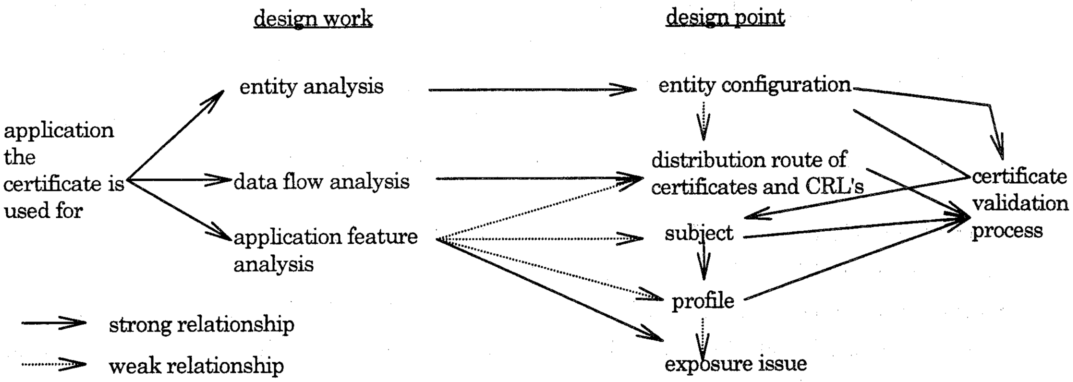
design work
design point
application the certificate is used for
entity analysis
data flow analysis
application feature analysis
entity configuration
distribution route of certificates and CRL's
subject
profile
exposure issue
certificate validation process
strong relationship
weak relationship

Figure 1: Relationship between design work and design point [11]
(originally written in Japanese)

Looking at the methods shown in Figure 1 today, it seems that some items should be improved. Among those items, based on design experience after the previous works and on discussion in some mailing lists[2,4,8], the following issues are discussed in this paper:

(A) information contained in a certificate

(B) revocation management not to use CRL's

2. Information Contained in a Certificate

Bad design on information contained in a certificate causes not only increase of certificate management cost but also security breach such as exposure of confidential information. But the confusion is often seen on what kind of information should be contained in a certificate. Although a certificate is the proof of the binding between a public key and its owner(subject), there are some cases where one tries to put information in a certificate, such as address, telephone number, .etc, which isn't (or shouldn't be) bound to the public key.

The author would like to propose 3 criteria by which one can determine if information X should be contained in a certificate or not.

Criterion-1 : Information X isn't a confidential information.

Criterion-2 : For the applications of the certificate, there exist some reasons that information X should be bound to the public key as identity information.

Criterion-3 : Information X doesn't have to be bound to the public key, but the information X has to be contained in the certificate for some reasons for the applications. For example, there is no other good way to convey the information X in the application than to contain it in the certificate.

Criterion-1 and Criterion-2 both have to be satisfied, or the criterion Criterion-1 and Criterion-3 both have to be satisfied.

Criterion-1 comes from the fact that a certificate is basically transferred through open networks . If one would like to send confidential data, one should do it after one gets the recipient's public key and encrypts the data with the key.

Criterion-2 comes from the fact that a certificate is essentially the proof of the binding between a public key and its owner(subject).

Criterion-3 is a kind of relief measure when Criterion-2 can't be satisfied and when there is no other good way than to contain information X in the certificate.

There is another idea behind those 3 criteria above. It is that it would be better to keep the data, certificate, simple. If a certificate contains superfluous data, it causes the following problems:

(1) The management cost of the certificate increases. The typical bad example is to put the address, telephone number, position in the company, etc. besides the name of the person(subject) in a certificate. In this kind of case, every time one of those data items is changed, the certificate has to be reissued (and for the old certificate, a CRL has to be issued). Unless there exist any reasons that the address or anything must be contained in the certificate (that is, unless those criteria above are satisfied), one shouldn't put those data items in the certificate. One should store those data items in a database or something, and use them after successful validation of the certificate (which means successful authentication of the entity). *1)

(2) One may get confused on what the certificate proves. In the example in (1) above, one

*1) A person once indicated in a mailing list that a certificate is not a record of a database. This seems to represent well one aspect of this issue.

may not be able to tell which is a real subject of the certificate. A person ? An address ? A telephone number, etc.

To keep a subject separate from related data items also brings benefits in access control which is often taken up as one of applications of a certificate.

First, if a person's position in a company, qualification, or something is changed, the certificate hasn't to be reissued and only the record in the database or something has to be changed. Secondly, when attribute certificates will be used in the future, one only has to put in the attribute certificates those qualification data items which will have been managed separate from the subject in the certificate.

In the end of discussion on the information contained in a certificate, two issues should be noted. The first issue is that there exists a case where information doesn't seem to be confidential at first. A typical example will be address.*2) The second issue is the case where people want to limit access to certificates, for example, by adopting access control mechanism to a directory, although information in a certificate is transferred in plaintext form through public networks when people send or receive messages using those certificates. This seems to be contradiction to the statement in X.509[5], "Because certificates are unforgeable, they can be published by being placed in the Directory, without the need for the latter to make special efforts to protect them", but it is often argued. The author isn't necessarily against access control to a directory, but it should be noted that information in a certificate is transferred in plaintext form when it is used.

3. Revocation Management not to Use CRL's

The bad design of revocation management causes not only performance problems , such as directory overwhelmed by a large number of accesses by clients, but also too long CRL latency problem, which sometimes grows trust chain breach.

The author showed that data (message) flow in an application influences much CRL distribution route design in the previous works[10,11]. Of course, this design is also much influenced by the numbers of entities, transaction volume, the geographical dispersion of entities, etc. In this paper, the author would like to indicate that one should think an alternative not to use CRL's in design of revocation management.

Revocation management including design of CRL distribution route is one of the biggest issues in PKI design. For this very reason, many technics have been designed such as CRL distribution points[5], enhanced CRL distribution options[1], etc. And how directories are geographically configured is related to this issue and is also one of biggest issues.*3) One can do without all those issues if one can design revocation management without CRL's.

A proprietary PKI in the reference[6] is a good example of this case. An example of proprietary

*2) In a mailing list, it was introduced that a woman had pointed out female users might not like inclusion of full name and address in a certificate because a stalker might know it.

*3) A person said in a mailing list that SET designers wanted to avoid this kind of problem, and they decided not to use directories.

PKI is one for internet-banking where a bank issues a certificate to its customer and provides banking services. In this case, there are at least two features that influence revocation management design. (The following content is described in the reference[6], but it is reconfigured from the view-point of revocation management design.)

(1) client-server type communication
A customer program requests a banking service to the bank server, and in return the server provides the service. There is no direct communication between clients (no peer-to-peer communication). All messages come to the bank server.

(2) The bank is a CA and service provider.
The bank identifies a customer and issues a certificate for the customer. The bank is also a service provider when the customer request a service with the certificate.
Of course, the alternative can be considered that a trusted third party plays the role of CA and that the bank plays the only role of the server. But thinking of the cost of identification and authentication of customers, and thinking of the benefit that CA knows the context where a certificate is used, it would be better for banks to be CA's.

From (1), a CRL of the customers doesn't have to be distributed, and only the server has to have it. And from (2), the bank server is considered to be managed so closely tied to the CA, then the bank server only has to have just a "revocation" list (not an X.509 CRL), or has only to query to the CA.

And in this example, certificate distribution doesn't require directories because a client accesses the fixed server and because it just gets the server's certificate (and a CRL of the server) from the very server. Therefore, one can do without directories and one doesn't have to think of issues such as where to set directories, whether they have duplicated data or not, etc.

4. Conclusion

As PKI design issues, information contained in a certificate and revocation management are discussed in this paper. Those three criteria are proposed for design of information contained in a certificate, and it is shown that they are also considered to work for access control. And the view to alternative not to use a CRL for revocation management is also shown. The author shall keep designing real PKI systems by using the methods in this paper and improving the methods to contribute to implement practical and efficient PKI systems.

Acknowledgements

The discussion in this paper is based on the discussion in cert-talk, ietf-pkix, seis mailing lists, the discussion with people in Information Security Department in Information Tchnology R & D Center in Mitsubishi Electric Corporation, and the discussion with other people. The author expresses here gratitude to those people all.

References

[1] P. Hallam-Baker and W. Ford, "Internet X.509 Public Key Infrastructure ENHANCED CRL DISTRIBUTION OPTIONS", draft-ietf-pkix-ocdp-01.txt, 1998 (work in progress).

[2] cert-talk mailing list, Web page, http://www.structuredarts.com/mailinglists.htm/

[3] R. Housley, W. Ford, W. Polk, and D. Solo, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile", draft-ietf-pkix-ipki-part1-11.txt, 1998 (work in progress).

[4] ietf-pkix mailing list, Web page, http://www.imc.org/ietf-pkix/

[5] ITU-T Recommendation X.509 version 3, "Information Technology - Open Systems Interconnection - The Directory: Authentication Framework", 1997.

[6] S. Kent, "How Many Certification Authorities Are Enough?", the Proceedings of MILCOM 97, vol.1, pp. 61-68, IEEE Press, 1997.

[7] J. Murai, "The internet", Iwanami-Shinsho 416, Iwanami-Shoten(1995).

[8] seis mailing list, Web page, http://www.nc-forum.se/seis/

[9] "SET Secure Electronic Transaction Specification" Version 1.0, May, 1997.

[10] J. Yoshitake and H. Sakakibara, "The points in design of an authentication system", the Proceedings of the 55th Annual Convention of Information Processing Society of Japan, vol.4, pp.4-459 - 4-460, 1997.

[11] J. Yoshitake, "A View on how to build an authentication system", Proceedings of The 20th Symposium on Information Theory and Its Applications (SITA97), vol.1, pp.69-72, 1997.