SCIS 2002 The 2002 Symposium on
Cryptography and Information Security
Shirahama,Japan, Jan.29-Feb.1,2002
The Institute of Electronics,
Information and Communication Engineers

Improving Performance in Global PKI using Virtual Certificates and Synthetic Certificates [*]

Selwyn Russell [†] Eiji Okamoto [‡] Ed Dawson [§] Javier Lopez [¶]

Abstract— A digital certificate may be used to inform the world of the public key of its owner. To guard against impersonations and fraud, the receiver needs to perform a series of checks. When a hierarchy of certificates is involved, and when there are large volumes of messages between two parties, as is frequent in commerce, the repeated validation of the same chain of certificates consume significant resources. This paper presents new concepts of *virtual certificate* and *synthetic certificate* which can be used to speed up repetitive processing of a chain with improved efficiency.

Keywords: public key infrastructure, certificate, virtual, synthetic

1 Introduction

The goal of a Public Key Infrastructure (PKI) is to facilitate the use of public key cryptograpy. It does this in part by simplifying the process of determining the public key of a particular entity in the community. One of the components of PKIs as presently being investigated by many governments [10] [1] [8] around the world is the digital certificate, invented in 1978 [6] as containing a person's globally unique identity, the public key cryptosystem and public key used by the person, and the time span (start and end times) of the usage of that key, and digitally signed by a "well known" trusted entity, the Certification Authority (CA), which had rigorously verified the identity of the person before issuing the certificate. The current content of digital certificates has undergone considerable enhancement from the original proposal, there now being a number of different groups and opinions [11] [5] [4] [2].

There are currently two major classes of certificates in PKIs, *identity* certificates and *attribute* certificates. The former is used to tell of the public key used by a particular entity within a specified time period. This paper is concerned with identity certificates which may be issued from a hierarchy of issuers, leading to a chain of certificates. Chains arise in PKIs which seek to provide strong assurance of the identity of the certificate subject, such as might be used in commerce and government activities. It addresses the processing problems incurred by the repeated processing of the same long chain of certificates, as could arise in communications between large organisations.

The following section provides a brief background of the problem. Subsection 2.1 outlines terminology used in this paper. Subsection 2.2 discusses the practical problems for a receiver in validating a chain of certificates. Section 3 introduces new concepts which can lead to processing efficiencies. A summary and brief outline of future work conclude the paper.

2 Problem Background

2.1 Terminology In This Paper

The certificate issuer at the top of a hierarchy is the *root CA*. The letter A without a subscript may be used to identify a root Certification Authority or the community of users with certificates issued by the root Certification Authority or a subsidiary Certification Authority. A root CA may be referred to as *Root CA* (RCA). A CA which is lower in the certificate chain than the RCA may be referred to as a *Subordinate* or *Subsidiary CA* (SCA).

A_i refers to a subordinate CA of the root CA, A. $A_{i,j}$ refers to the j-th end user at the level served by the subsidiary CA, A_i. A certificate issued by CA, X, regarding entity Y is written as $C_{X,Y}$.

2.2 Chains of Certificates

A member $A_{j,k}$ at level j of a group with top level Root Certificate Authority A has available to it the chain of certificates $C_{A,A}$, C_{A,A_1}, C_{A_1,A_2}, ..., C_{A_{j-1},A_j}, $C_{A_j,A_{j,k}}$, where A is the RCA, A_1, ... A_j are the subrodinate issuers of the lower level certificates. End entity $A_{j,k}$ has a certificate $C_{A_j,A_{j,k}}$ issued to it by issuer A_j. The set of certificates may be sent by $A_{j,k}$ in communications. The receiver must sort these into a chain and then can work through the chain, starting with the "well known" public key of the RCA and validate each certificate in turn until that of the sender has been processed.

[*] This research was partially supported by the Telecommunications Advancement Organisation of Japan

[†] Information Security Research Centre, Queensland University of Technology, Brisbane, Australia. S.Russell@qut.edu.au

[‡] Faculty of Science, Toho University, 2-2-1, Miyama, Funabashi, 274-8510 Japan. Okamoto@sci.toho-u.ac.jp

[§] Information Security Research Centre, Queensland University of Technology, Brisbane, Australia. E.Dawson@qut.edu.au

[¶] Department of Computer Science, University of Malaga, Spain. jlm@lcc.uma.es

For each certificate, the validator must perform the following actions, not necessarily in this order:

- the components must be extracted,
- the start time and end time must be checked to ensure the certificate can be active,
- the revocation reference location given in the certificate must be determined and contacted to test for revocation before the expiry date,
- the signature and algorithms used by the signer must be determined,
- other possible restrictions must be determined, e.g. key usage policies, and checked for compliance. This analysis and assessment can be a problem, e.g. in many of the more complex X.509 certificates using version 3 or later extensions.
- the public key of the signer must be obtained,
- the signature on the certificate must be verified.

Large enterprises with multiple locations, e.g. large factories or processors in a number of cities or countries, may have a CA and root key for the enterprise, possibly a number of secondary keys for daily use by that CA, and additional local CAs for each region. For added security, these local CAs may have secondary keys for daily use, leading to four levels of certificates so far inside the enterprise. Adding levels for a national CA and outsourced CAs might add another four certificates, leading to a chain of around a dozen for a bottom entity in a large enterprise. In commerce and global P-KI, the volumes of communications between these low level entities in large enterprises are likely to be quite large, and involve the processing of the same chain time after time, consuming significant resources over a period of time. Even though there is a very low probability of fraud, these checks need to be carried out each time to comply with fiducuary responsibilities. Can repeated validations be handled more efficiently? The next section provides some foundations for improvements.

3 Improving Processing of Repeated Communications

To improve the process of frequent verification of a chain, we introduce two new concepts, a *virtual digital certificate* and a *synthetic certificate*, as described below. We then continue by showing how these may be applied in practice.

3.1 Virtual Digital Certificates

3.1.1 Introduction

Consider when an entity X in one community receives a message supposedly from Y in another community. A chain of certificates may be received indicating that Y has a certain public key $PublicKey_Y$. Entity X wants to determine if that is really the public key of Y, and if the digital signature of the message indicates its source is Y. If X's community has not encountered Y before, the certificate chain will have to be processed by X or another entity in X's community or by their agent, or X might ask a trusted reference for the public key of Y. For subsequent messages from Y to X, the first method is time and resource consuming, and we will pursue the option of asking a trusted reference.

In theory, it is not necessary to have a long chain of certificates from a root CA to the end entity Y, the root CA could issue a certificate directly to Y, and then the X community would have a simple task of verification. Of course, this ideal single certificate is not scalable or practical, but it would speed up processing by the receiver if it did.

In this paper, we propose an entity in the X community which acts as if such an ideal certificate for Y did exist and it has possession of it, thereby being able to divulge the public key of Y very efficiently to X and to others who enquire. Because this single level certificate does not really exist, we call it a *virtual certificate*.

Next we provide a more formal specification and details of components and validation, followed by the related concept of a *synthetic certificate*.

3.1.2 Specification

Definition 1. A *virtual digital certificate* (or *virtual certificate*) is a data set which is derived from a chain of certificates, containing the information which would be in a digital certificate issued by the first Certificate Authority in the chain to the end entity if one had been issued, but such a certificate has not been issued.

Definition 2. *Existence:* for a chain of certificates, which are compatible in policies and in extensions, beginning at entity E_1 and extending to entity E_N, a virtual certificate VC_{E_1,E_N} exists at a time T if and only if there exist valid certificates C_{E_1,E_2}, C_{E_2,E_3}, ..., C_{E_{N-1},E_N}, where $C_{X,Y}$ indicates that a digital certificate has been issued by entity X attesting to the public key of entity Y and is valid at time T.

Notes:

- The data set is used by the entity which created it, the *Virtual Certificate Manager* (VCM), and by other entities which trust the work of the VCM.
- The data set is not expected to be made available to other parties, but retained by the VCM, which acts as an authority on the value and status of the public keys of the end entities for which it has built virtual certificates. Other parties who want the data set information should instead seek the corresponding "synthetic" certificate, discussed below in 3.2.
- The contents of a VC are probably not released by the VCM.
- The formation of the VC may require complex processing to determine the compatibility of policies and extensions.
- From definition 2, a virtual certificate ceases to exist if any of its component certificates expires

or is revoked. This is very important because it greatly simplifies validations after the first, avoiding the complex rechecking of policies, paths, etc., as discussed below in 3.1.4.

3.1.3 Components

Components of a virtual digital certificate include standard items:

- Issuer. Optional, the issuer of the first certificate in the chain. Might not be used in practice.
- The commencement validity date (CVD): the latest of the commencement validity dates of the component certificates

  $CVD_{VC} = \mathit{latest}(CVD_{C_m})$ for all m certificates C_m.
- The expiry validity date (EVD): is the earliest of the expiry validity dates of the component certificates

  $EVD_{VC} = \mathit{earliest}(EVD_{C_m})$ for all m certificates C_m.
- The Subject: the identity of the final entity in the base certificate chain.
- The public key information (PK): the public key information in the final certificate of the chain, designating the algorithm concerned and the actual public key.

  $PK_{VC} = PK_{C_{i,j}}$

The X.509 items *version*, *serialNumber*, *signature*, have no meaning or practical use here. The items *issuerUniqueID*, and *subjectUniqueID* seem to be rarely used and are not essential to the principles given here.

A new component, the *component certificate revocation list*, will be added, as described below.

3.1.4 Validation

Definition 3. Validation at a particular time means the process of determining if a certificate is intended by its signer to be current at that time and has not been revoked, or, in the case where a chain of certificates is involved, whether every one of the certificates in the chain is intended to be current at that time and has not been revoked.

If a virtual certificate exists, the corresponding chain of real certificates is valid, from Definition 2. So, for purposes of validation of a chain of real certificates, it suffices to determine the existence of the virtual corresponding certificate. Hence, if a virtual certificate is known to exist at a time T, then its existence at time $T+dt$, provided $T+dt$ is earlier than its expiry date, is true if and only if every one of its components has not been revoked. In practice, this principle reduces the revalidation of a certificate path to a series of revocation checks, without requiring a repetition of the N hash and digital signature calculations. Therefore to speed up the revocation checks, the Virtual Certificate will need a new item, a *Component Certificate Revocation List* (CCRL), which identifies all of the components from which it was formed, so that each can be checked if required for revocation. For each component certificate of the virtual certificate, the Issuer and the unique identity assigned by the Issuer to the component certificate, along with the status / revocation check point (or, for CRLs, an issuing distribution point) if available, should be adequate for the purpose.

Definition 4. *Component Certificate Revocation List:*
$CCRL_{VC} = \mathit{set}(Issuer_m, CertID_m,$
$RevPointType_m, RevPoint_m)$
for all m certificates C_m.

Revocation checks could be carried out by a message receiver but we expect that the Virtual Certificate Manager would provide a service whereby the receiver would make a revocation enquiry of the Virtual Certificate Manager and the Virtual Certificate Manager would run the revocation checks using the $CCRL_{VC}$ and report the result, storing it for re-use over the short term. Most users are not interested in the content of the Virtual Certificate, only whether the claimed public key which they have received from the sender can be trusted. As such, it is better to adhere to the information hiding principle, and have repetitious functions performed by the Virtual Certificate Manager.

3.2 Synthetic Certificates

3.2.1 Introduction

A VC for entity Y is managed by a VCM and would probably not be circulated to other entities. Other processing entities which trust the VCM may wish to have a single level certificate for the entity Y, so the VCM, which is convinced of the public key of Y, could issue a certificate for Y, $C_{VCM,Y}$, which could then be used by the other processing entities. It would contain a reference to the VCM as the point to which enquiries for revocation should be directed. We use the term *synthetic certificate* or *synthesized certificate* to describe this certificate issued by a VCM which has no direct relationship with the party whose public key is being certified. It has most of the content of a virtual certificate and is signed by an entity, the Synthetic Certificate Manager (SCM). A difference from the virtual certificate is that the synthetic certificate normally has no revocation list, only the revocation contact point for the issuer. However, a revocation list could be provided in an extension field, but this shifts some of the repeated processing to the receiver, which we are trying to avoid. Revocation checks are conducted by the Synthetic Certificate Manager.

A synthetic certificate may be available even though the corresponding Virtual Certificate is not. Even though a Synthetic Certificate and a Virtual Certificate are theoretically related, in practice one could be in use without the other, but we envisage that a Virtual Certificate Manager would have available the Synthetic Certificate, and a Synthetic Certificate Manager would make available the Virtual Certificate.

3.2.2 Specification

Definition 5. A *Synthetic Certificate* (SC) is a digital certificate constructed by an entity which is trusted by some other parties, stating the link between an entity to which it is not directly related and its public key, the content having been derived from other sources.

Definition 6. The *Synthetic Certificate Manager* (SCM) is the entity which constructs the synthetic certificate and provides revocation status information to enquirers.

Notes:

- A synthetic certificate could be easily created by a VCM from a VC.
- The SCM normally would not be involved in the certificate chain, $C_{E_1,E_2}, C_{E_2,E_3}, ..., C_{E_{N-1},E_N}$ from which the virtual certificate, VC_{SCM,E_N}, was derived.

3.2.3 Components

Components of a synthetic digital certificate include standard items:

- Version, at the discretion of the Virtual Certificate Manager.
- SerialNumber, a unique identifier at the discretion of the Virtual Certificate Manager.
- Issuer, now the Virtual Certificate Manager.
- Signature (algorithm) used by the Virtual Certificate Manager.
- The commencement validity date, as in the Virtual Certificate, but probably unnecessary in practice if checks are carried out through the Virtual Certificate Manager.
- The expiry validity date, as in the Virtual Certificate.
- The Subject, as in the Virtual Certificate.
- The public key information, as in the Virtual Certificate.
- A revocation check point, e.g. a server process ID or port.
- The type of revocation check point, e.g. OCSP [7], SCVP, etc.

The X.509 items *issuerUniqueID*, and *subjectUniqueID* can be omitted here.

3.2.4 Validation

The entity certifying the public key of the end entity of the certificate chain has synthesized a single certificate from the information contained in the public chain after validating each and every one of the components in the chain. Thus the user of a synthetic certificate need not repeat the expensive validation checks already run by the issuer of the synthetic certificate.

A synthetic certificate is valid only if the corresponding virtual certificate is valid.

If a synthetic certificate is known to be valid at time T, then it is valid before the expiry date at time $T+dt$ if and only if the virtual certificate has not been revoked. Hence, an entity which has accepted the synthetic certificate at one time, and seeking to revalidate it at a time prior to expiry, need only check with the signer, i.e. the VCM, for revocation.

3.3 Uses for Virtual Certificates and Synthetic Certificates

3.3.1 In-house Applications

For in-house operations at one site, the computer network is generally considered low risk, and there is a predefined relationship amongst the nodes. Often little need is seen for public key security for internal communications. For applications in this environment, a workgroup or enterprise server might construct a virtual certificate for use by an end user and store it for later use. There is relatively little benefit to be gained from having the enterprise server conduct the first processing, perhaps a short time due to processing on a faster machine. The benefits grow from the repeated requirement to process the same chain, in which case the contents of the corresponding Virtual Certificate expedite the decision.

For the first processing of the chain, the Virtual Certificate is constructed, stating that the public key of remote entity X is $PK.Val$, and some auxiliary information to be used later if required. Later, when a chain with an end target identified as X is received, the current set of Virtual Certificates is consulted and those containing X are examined for one with a public key of $PK.Val$. If a suitable unexpired Virtual Certificate is located, it is sufficient to validate it in accordance with the method outlined above. Probably in most cases, entity X will have only one certificate chain and therefore only one active Virtual Certificate, so the later validation of the public key of X will be quite fast, particularly if the relationship between the validator and the revocation reference sites allows hash based communications [9].

More explicitly, on the first occassion, the operations are

- validate the end entity using conventional methods, involving checking of hash values, digital signatures, policy information, and any extensions;
- create the virtual certificate with its contents of Issuer, commencement validity date, expiry validity date, subject identity, public key information, revocation information list, and anything else deemed to be necessary for local validation later.

On later occasions, given an identity, a presumed public key, and perhaps the identity at the top of the certificate chain,

- find a virtual certificate with a matching subject identity;

- compare the offered public key with that in the Virtual Certificate;
- if there is no match, look for another Virtual Certificate;
- complete the validation by ensuring that no revocations of components have occurred.

Because there is no need to reprocess the whole chain, later verifications of the public key should be relatively fast.

3.3.2 Public Applications

In the above example, an internal network of satisfactory security was assumed, and the users of the Virtual Certificate were internal entities. Where the receiver of a frequent certificate chain can have prior registration with the Virtual Certificate Manager, hashing of communications can provide efficient secure communications, e.g. [9], and a Virtual Certificate can be used by entities outside of the Virtual Certificate Manager's organisation, i.e. the public. For many situations, prior registration will not be feasible, and other means need to be used for secure communications. In most cases, providing the machines involved are satisfactorily secure, the use of a network link secured by SSL / TLS [3] and signed data structures, e.g. the synthetic certificate, are probably adequate.

3.3.3 Abolition of Certificate Chain after Initial Contact

Initially the VCM uses the certificate chain from entity Y to verify the identity of the communicant and to build the VC for Y. Thereafter, when a member X of the VCM's community receives a message from Y, there is no need for Y to send the chain again. This is of benefit where the communication bandwidth between X and Y is limited, as in wireless applications. For verification of a message allegedly from Y, X enquires of the VCM for the public key of Y. If Y's public key remains the same, the enquiry will be processed quickly by the VCM. If there has been a change in the public key of Y, the VCM will still return the same key (unless there has been a revocation somewhere in the chain) but the signature check will fail, and the VCM will be called upon to re-initialise the VC for Y or issue an additional one, depending on the situation.

3.3.4 Wireless Networks and Connected Limited Devices

These devices suffer from resource limitations because of their size and weight, factors which are determined by the public preference rather than by technical considerations. Because of their available resources they would have difficulties in processing a chain of certificates. A VCM would relieve the device of processing of a chain, and even processing of a single certificate, and would act as a reference which could be consulted to determine if a received public key is still valid.

3.4 Virtual Certificate Directory

Virtual certificates could be made available online for low power devices and limited network devices, which are members of a community served by a VCM.

For example, the Service Provider is a trusted party for users of mobile telephones, and the telephone network is relatively secure compared with the Internet, with the exception of the air link to the closest Base Station. The Service Provider or Network Operator could establish a VCM as a value added service for subscribers. One option would be to maintain VCs for senders specified by the subscriber, e.g. business associates and staff. Another is to maintain a general directory which could be queried on demand, and could contain public key information of frequently requested entities, as revealed by statistics.

3.5 Synthetic Certificate Directory

An online directory storing synthetic certificates would aid receivers of certificate chains. It is functionally similar to the above Virtual Certificate Directory in that information on a public key is available without processing a chain of certificates by the receiver. The SCM will be listed as the reference point for revocation checks and will perform the repetitive work involved on request, e.g. via an OCSP enquiry.

4 Summary

The repeated validation of a chain of certificates can be time consuming and expensive over a period of time when done without memory of previous validations. Converting a chain into a Virtual Certificate will improve validation within an enterprise. Converting a chain into a Synthetic Certificate extends the usefulness to clients in other communities who trust the Synthetic Certificate Manager.

5 Future Work

Further research is underway into techniques of generating and using virtual and synthetic certificates and appropriate directories, for both private and public environments.

References

[1] Peter Alterman. The U.S. Federal PKI and the Federal Bridge Certification Authority , 7 May 2001.

[2] Stefan A. Brands. *Rethinking Public Key Infrastructures and Digital Certificates, Building In Privacy.* MIT Press, Cambridge, Massachussets, 2000.

[3] T. Dierks and C. Allen. *RFC 2246: The TLS Protocol Version 1.0*, January 1999. RFC 2246.

[4] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen. *SPKI Certificate Theory*, September 1999. RFC 2693.

[5] R. Housley, W. Ford, W. Polk, and D. Solo. *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, January 1999. RFC 2459.

[6] Loren M. Kohnfelder. *Towards a Practical Public-key Cryptosystem.* May 1978. MIT B.S. Thesis.

[7] M Myers, R Ankney, A Malpani, S Galperin, and C Adams. *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP.*, June 1999.

[8] Office of Government Information Technology. Gatekeeper, a strategy for public key technology use in the Government. online, 6 May 1998. http://www.govonline.gov.au/-publications/GatekeeperStrategy.pdf.

[9] Selwyn Russell. Fast Checking of Individual Certificate Revocation on Small Systems. In Jeremy Epstein, editor, *Fifteenth Annual Computer Security Applications Conference*, Radisson Resort S-cottsdale, Phoenix, Arizona, 6–10 December 1999. Annual Computer Security Applications Conference. http://www.acsac.org/1999/papers/thu-a-1300-russell.pdf.

[10] Satoru Tezuka. Trend of Japanese PKI and International Cross Certification. ICU, Daejeon, Korea, 19–20 October 2001. International Research Center for Information Security, Korea and Institute of Industrial Science, Japan. http://www.iris.re.kr/iwap01.

[11] P. R. Zimmermann. *The Official PGP User's Guide.* MIT Press, Cambridge, Massachussets, 1995.