SCIS 2003 The 2003 Symposium on
Cryptography and Information Security
Hamamatsu,Japan, Jan.26-29,2003
The Institute of Electronics,
Information and Communication Engineers

Certificate Retrieval and Validation in Online Systems

E. Okamoto [*] J. Lopez [†] E. Dawson [‡] J. M. Gonzalez-Nieto [§] S. Russell [¶]
J. Smith [‖]

Abstract— In order to more effectively deal with certificate management issues in PKIs, there is growing interest in supplementing offline X.509 PKI models with online services. The advantages of online services include a reduction of complexity on client devices and the centralisation of certificate processing policy. An analysis of the security requirements of online models will be presented. Proposed online and delegated processing models will be evaluated in relation to these requirements.

Keywords: pki, certificate validation, certificate revocation, online systems, trust, key management

1 Introduction[1]

!!Introduction to the online paradigm.

In order to more effectively deal with certificate management issues, especially revocation management, there is growing interest in supplementing the offline X.509 style PKI models with online services such as Online Certificate Status Protocol ($OCSP$) [9] and Delegated Path Discovery / Validation (DPD/DPV) [11].

Additionally, the use of online and delegated services allows the complexity of certificate revocation checking to be confined to well defined (and maintained) servers, with the beneficial side-effect that lightweight devices, without the requisite processing and storage capabilities can participate in public key cryptography (PKC) mediated transactions. These delegate servers also provide organisations wth an opportunity to enforce consistent policies with regard to the use of (PKC).

In this paper the security requirements of online validation systems will be identified and the current proposals for online validation services will be evaluated against these security requirements. The following section will describe a generic delegated services model. Section 3 will discuss the security requirements of such a delegated model. An evaluation of the currently proposed models security features will be summarised in Section 4. Examples of implemented delegated services are presented in Section 5. The paper concludes with directions for future work in Section 6.

2 Delegated Services

There are numerous proposals that describe protocols for delegated services. For the purpose of identifiying security requirements we define the following delegated service model (See Figure 1).

A client, that may be a node on a private corporate network or on the wider Internet, wants to ascertain the validity of a public key which we will assume is in the form of a certificate. The client operates in the role of *relying party* and wants to obtain non-repudiatable evidence of the validation of the public key they are relying on.

In order to determine the validity of a certificate, the client formats a request that contains information to identify the certificate being validated (or the certificate itself) and some validation contstraints such as the purpose for which the certificate is being relied on, or trust anchors to be used. This formatted request is then sent to the delegated service that the client chooses or is configured to use.

On receipt of the formatted request, the delegated server will attempt to validate the request. This validation process is likely to include: validation of the certificate referenced by, or contained in the request; the construction of a certificate path to a trust anchor; and validation of each certificate in the certificate chain according to the validation policy. Validation of the end entity certificate and intermediate certificates involves the retrieval of those certificates and verification of the signature on each certificate. Revocation checking via CRLs, OCSP responders, or other delegation servers will also be performed.

Once the delegated server has completed this path building and validation, the validation result is formatted into a response message and returned to the requesting client.

The architecture presented in Figure 1 is in a highly simplified format. It is important to note that a number of the components may make recursive service calls.

[*] Tsukuba University, Japan
[†] University of Malaga, Spain
[‡] Information Security Research Centre, Queensland University of Technology, Australia
[§] ‡
[¶] ‡
[‖] ‡

[1] This research is supported by the Telecommunications Advancement Organisation (TAO) of Japan.

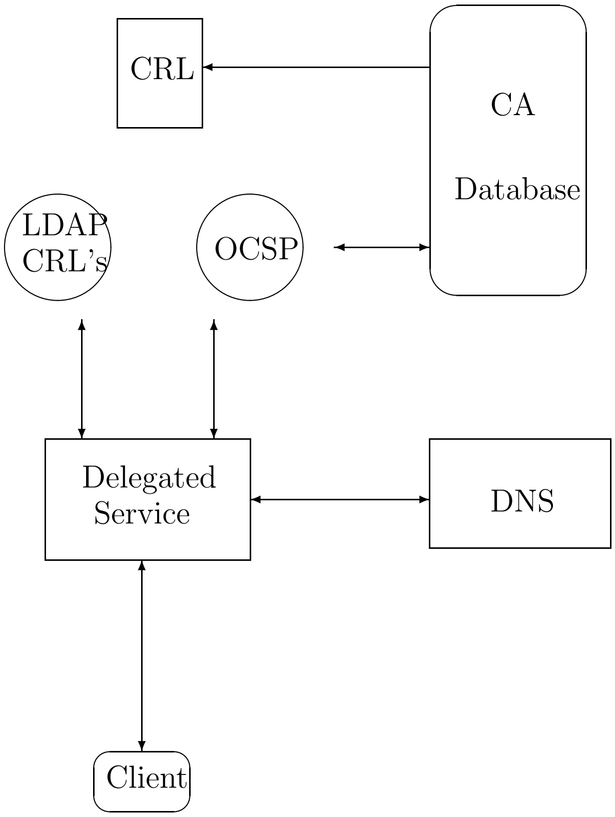

Figure 1: Components of delegated architecture

For example, a delegated service may call another delegated service to perform some, or all, of the requested validation. In handling a service request an OCSP responder may call on another OCSP responder. Likewise, name resoultion request made to DNS may be forwarded to other DNS hosts.

3 Security Consideration in Delegated Services

In this section, we identify security considerations for each of the component interactions shown in Figure 1 that are relevant to the delegated service model.

3.1 Client Considerations

A client sending requests to a delegated server may be on a corporate network or may be a node on the Internet. In submitting a request to a delegated server the client requires the following:

- Server authentication.
- Request message integrity protection.
- Response message integrity protection and non-repudiation.
- Response message replay resistance.
- Validation policy agreement.

In establishing a connection, authentication of the server must be performed. Server authentication can be achieved in a number of ways, the most obvious being by the use of TLS [4] or through the use of an IPSEC Security Association (SA) [5][2]. If strong authentication of the server is not completed, then the client application will be highly reliant on the correct operation of DNS to direct them to the correct server and unless access to the network link between the client and the delegated server is secure (which is not likely in an internetworked environment) validation request messages will be vulnerable to redirection and man in the middle attacks.

Once the client is convinced that it is communicating with the appropriate delegated service it needs to be sure that the contents of the validation request are not modified enroute to the server. If message integrity is not guaranteed an attacker may be able to modify the reference to the certificate being validated, or alter other validation parameters resulting in an unreliable response from the delegated server.

When receiving a response from the delegated server, the client needs to be sure that the response has not been modified, thus altering the outcome or details of the validation attempt.

The client may require that the contents of the server response be able to serve as non-repudiatable proof that validation had been performed on a given certificate. This information may be maintained in the servers logs (suitable for a corporate environment) or may be part of the response. In order to provide non-repudiatable proof, the response message must be digitally signed.

The client needs assurance that the validation response is fresh and not a replay of a previous response for which the validation status may have changed. Replay resistance can be achieved through the use of timestamps or nonces.

Because the validation response may vary depending on the validation parameters used, the client must be sure that the validation policy used by the delegated server is consistent with the validation request parameters. If the server validation policy is inconsistent with the client validation policy request, as may be the case when an organisation is using a delegated service to enforce organisational wide validation polices, the client should be notified of this policy conflict.

3.2 Delegated Service Considerations

The delegated service is a core component of the architecure and has security requirements for each of the components it interacts with. When performing validation, the delegated server may become a client to: other (trusted) delegated servers; OCSP responders; and directories acting as certificate or CRL repositories.

In order to increase performance of the delegated server it may be necessary to implement some form of result caching (see VCM section). If results are cached, the integrity of this cached information must be protected as it becomes the authoritative source of information used in validation requests.

[2] The establishment of a SA between nodes on a corporate network is significantly easier than the establishment of such an association over the open Internet.

3.2.1 Communications with Client

The delegated service receives validation requests from a client, processes the requests and provides a validation response. The delegation service is required to commit a large number of resources in responding to a validation request. These resource include processor time, memory and disk space, and network bandwidth. There is a great risk of denial of service (*DoS*) attack if the delegated service cannot distinguish between legitimate and illegitimate validation requests. If authentication services are to be used to make this distinction, care must be taken that second order denial of service attacks against the authentication protocol are not possible. The use of client puzzles / storage of minimal state may be of benefit[refs]. The ability to detect replayed request messages and the encoding of the target delegation server in the request message may serve to minimise DoS attacks against the delagted server.

3.2.2 Communications with Peer Delegated Servers

When utilising the services of a peer delegated server, the security requirements are similar to those identified in subsection 3.2.1 as the delgated server is a client of the peer server.

3.2.3 Communications with OCSP Responder

When retrieving information about the curent status of the end entity certificate, or an intermediate certificate in the path to the trust anchor, the delegated server may request status information from an authorised OCSP reponder. When requesting an OCSP response, the delegated server must have assurance that the request is not modified in transit and it must be able to verify the signature on the response.

Some OCSP responders support signed requests that would ensure that the OCSP request arrives at the responder unmodified. The processing of signed requests increases the load on the OCSP responder, and unless the business model permits per request charging, this option may not be enabled. If signed requests are not supported or a secure channel (eg SSL) cannot be established to the responder, the delegated server must ensure that the certificate validated in the OCSP response is the certificate that the status was requested for.

3.2.4 Communications with LDAP

The construction and validation of the trust path from end entity certificate to trust anchor is likely to require the delegated server to retrieve certificates and revocation information from a directory service (eg LDAP). The design of traditional PKIs treats the directory as untrusted storage, only storing integrity protected records (eg certificates and CRLs) within the directory.

3.3 Infrastructure Considerations

When strong authentication of servers is not in place (eg LDAP retrievals of CRLs), there is an increased reliance on the DNS to operate correctly. When this is combined with the passive nature of certificate validation using CRLs (ie that a certificate is valid unless you can find evidence that it is not), a disruption to the DNS may result in the delegated service not being able to resolve or contact the authoritative directory storing the certificates and CRLs required for validation.

Secure time services will be required to support validation services and replay prevention mechanisms.

3.4 Trust

Branchaud and Linn [3] survey the implications of delegated validation models and identify the following trust management issues:

- Even where the client has the opportunity to provide the delegate server with policy inputs or trust anchors, such data are advisory as the client can neither control, nor examine the servers internal processing.
- Fully delegated validation permits delegated servers to operate with high levels of autonomy as their clients have no way of verifying the *correctness* of the delegated processing.

In order to be successful, the trust implications of delegated and online services need to be fully considered.

4 Security in Current Proposals

In this section, current proposals for delegated serviocces will be evaluated on the following:

- Client Communications
  - Server authentication mechanisms available.
  - Message integrity protection provided (request and response messages).
  - Response message replay resistance.
  - Validation policy agreement.
- Delegated Server Communications
  - Denial of service resistance
  - Request message replay resistance

4.1 Delegated Path Discovery and Validation (DPD / DPV)

Delegated Path Discovery (DPD) and Delegated Path Validation (DPV) are defined in RFC 3379 [11] and allow clients to offload the cumbersome process of building certificate paths and validating certificates. This path discovery or path validation is conducted according to some validation policy.

There are currently four proposals before the IETF that implement DPD / DPV. The Certificate Validation Protocol (CVP) [10] and Simple Certificate Validation Protocol (SCVP) [8] implement the requirements specified by DPD / DPV. The Data Validation and Certification Server (DVCS) Protocols [1] have functionality that permit DPD / DPV and there was a proposal for implementing DPD / DPV Using OCSP Extensions [7].

4.1.1 Certificate Validation Protocol (CVP)

The Certificate Validation Protocol (CVP) can be used to query the validation or discovery policies used by a CVP server; validate one or more public key certificates according to a single validation policy; or obtain one or more certificate paths according to a single discovery policy.

Degree of integrity, authentication and replay resistance will be summarised.

4.1.2 Simple Certificate Validation Protocol (SCVP)

TBD

4.1.3 Data Validation and Certification Server (DVCS) Protocols

TBD

4.1.4 DPD / DPV Using OCSP Extensions

TBD

4.2 XML Key Management Specification

Separate from the IETF effort to implement DPD / DPV, the W3C is developing the *XML Key Management Specification (XKMS)*. Like the IETF work a primary goal of XKMS is to offload complex PKC operations to trusted services. A distinguishing feature of the XKMS effort is that it does not presuppose that X.509 is the underlying PKI technology and claims to support SDSI / SPKI and PGP certificates in addition to X.509 based certificates.

XKMS specifies protocols for distributing and registering public keys. The following functions are supported by $XKMS$: registration of client or server generated key pairs; retrieval of registered public key material; and validation that a registered public key has not expired or been revoked.

There are two components to the specification, the *Key Information Service Specification (X-KISS)* and *Key Registration Service Specification (X-KRSS)*.

Security services of the KISS component will be discussed here [TBD]

5 Implementation of Delegated Servers

Although there have been many documents regarding how a delegated service system could be employed to advantage and how communications could be made with a system, there has been little written on the implementation of such an entity. One applicable paper in this field was presented by the authors [13], which introduced "virtual certificates", "synthetic certificates", and their managers. This section reviews this work and shows how they are of benefit to delegated services.

5.1 Virtual Certificates and Virtual Certificate Managers

The processing of a single certificate is not seen as a serious difficulty for most devices, but a lengthy chain of certificates can be very difficult or even infeasible to process, particularly for wireless devices [12]. With governments in Asia generally supporting a single hierarchical certificate tree [14] [2], long chains are to be expected for communications between persons in large enterprises. In many internet and extranet situations, the same persons routinely correspond and present their PKI credentials on each occasion.

The virtual certificate concept is a means of avoiding repeated processing of the same PKI certificate string by converting the chain into an equivalent single "certificate"[3], which is much easier to process. However this single certificate does not really exist, but an entity, termed the "virtual certificate manager", can appear to possess such a certificate, and it can rapidly validate a certificate which is the leaf at the end of a long branch in a certificate tree, without reprocessing the entire chain.

A virtual certificate manager (VCM) is only worthwhile when the same certificate chain is processed repeatedly, but there is a significant saving for later (re-)processing. On the first encounter, the VCM has to process the entire chain in a conventional manner, but it builds up some data structures holding the essential content and also data structures allowing later validation of the same chain without costly processing.

In operation, a client presents a chain of certificates to the VCM and the VCM informs the client if the content of the end entity's certificate is validated by the set of certificates. Correct application or usage of the content is generally beyond the scope of the VCM; for example, it is up to the client to determine if a key in a certificate has been used correctly with a message.

5.1.1 Applications

A basic requirement for a virtual certificate system is that the VCM must be trusted by the clients. This will apply for a VCM operated by the enterprise with its users being employees performing their allocated duties. However, a VCM which offered its services to strangers would have to convince them of its trustworthiness.

A VCM is financially viable when the same sets of certificate chains occur frequently in incoming communications, as in large enterprises handling high volumes of transactions with a small set of respondents, as in manufacturing and export. The benefits are proportional to the volume, with a significant performance gain when there are large volumes in a short time frame. There is no benefit over conventional implementations if a chain is received only once. The needs of many large enterprises can be met with an in-house VCM.

If used in a subscription service for the general public, there are likely to be millions of virtual certificates created and stored, with a low reuse rate. Unlike the enterprise VCM and its storage of a relatively low number of frequently accessed virtual certificates in random access memory, different storage and access techniques would be appropriate for the infrequently accessed storage for the general public.

[3] The VCM performs path discover, path and certificate validation, and status and constraints checking.

A VCM is thus attractive for delegated servers for enterprises but is less suitable when used only by individuals.

5.1.2 Strengths and Weaknesses

A VCM is valuable when operated as an intra-enterprise service, but the network and hardware need to be carefully designed to avoid bottlenecks at the VCM machine. For high availability, a design avoiding a single point of failure is preferred.

The link between the client and the VCM is of concern. For best security, encryption is desired, but for faster performance, plain text would be the choice. For in-house situations with all communications running on a well maintained and trusted internal network at the one site, plain text might well be regarded as an acceptable compromise.

The system needs to be protected from attacks in two areas. First, as with many security servers working with time span limited certificates (Kerberos for example), the system time needs to be secure and dependable. The multiple time source approach used by the Distributed Computing Environment design is a good example. Secondly, the VCM uses conventional tests initially and conventional revocation checks subsequently, and so is dependent upon the functionality and security of the network and remote servers.

In some situations the VCM and potential users are separated by untrusted networks. An extension of the virtual certificate, the “synthetic certificate”, can assist here by providing a single certificate for the user to process itself.

5.1.3 Security Requirements Addressed

A VCM meets the application level security requirements of fast economical determination of the validity of a digital certificate, particularly when a long chain is supplied, and the validation of the certificate in turn allows for assurance of integrity, confidentiality, and authentication of origin of the message with which the chain was received.

The VCM does not address the issues of how one communicates with it, the types of channels employed, or network security. Protocols such as SCVP and techniques such as TLS and IPSEC offer suggestions for some of those communication oriented issues.

5.2 Synthetic Certificates and Synthetic Certificate Managers

A limitation of the VCM is that it does not have a single certificate which can be passed on to another entity for subsequent use without reference to the VCM, although in principle the data structures could be handed on. As an alternative to sending the data structures, the VCM can create a certificate containing the end entity information and signed by itself, but not signed by any Certification Authority in the original chain. This is of course useful only to those who trust the VCM, but it does provide those users with a single certificate for the entity in question, with the accompanying simple processing. For revocation checks, the users must go back to the issuer of this synthetic certificate, as with conventional systems. For a VCM, the revocation testing for synthetic certificates is the same as it performs for its virtual certificates, so there is no added burden on the VCM. Although not essential, a VCM is the best choice to issue and manage synthetic certificates, i.e. to be the Synthetic Certificate Manager (SCM).

A user who receives a chain of certificates can approach the SCM regarding the end entity of the chain. If the entity is known to the SCM, it can reply with a synthetic certificate containing validated content. The user then works with the synthetic certificate.

Again trust in the SCM is essential. Intra-enterprise applications are the most likely, e.g. between a head office in Tokyo and a subsidiary in New York.

The benefits are similar to those of virtual certificates, and where a SCM is also a VCM, the strengths and weaknesses are similar.

Synthetic certificates address the same security requirement for efficient low cost validation of an end certificate in a chain.

5.3 Summary

A chain of certificates can be converted to a single virtual certificate so the chain can be validated efficiently when received on subsequent occasions. A user of this system does not process any certificates, relying on the VCM. The content of the virtual certificate can be converted to a real certificate as a synthetic certificate so the holder need process only a single certificate and not a lengthy chain. A synthetic certificate would be used as an alternative if use of a virtual certificate system presented some difficulties.

For cost effectiveness, the chains should be received in large volumes, as in large enterprises. The managers of the virtual certificates and synthetic certificates must be trusted by the users, implying that managers of these certificates would be good candidates for delegated servers within large enterprises.

6 Conclusions and Future Directions

Each of the proposed IETF models requires the relying party to validate a specific certificate. A useful extension to the IETF proposals would be to allow users to submit a unique name for an entity and a purpose, and have the delegated service return a suitable certificate - like Cert’EM.

References

[1] C. Adams, P. Sylvester, M. Zolotarev, and R. Zuccherato. *Internet X.509 Public Key Infrastructure Data Validation and Certification Server Protocols.* ftp://ftp.isi.edu/in-notes/rfc3029.txt, February 2001. RFC 3029.

[2] Shuo Bai. PKI in China. In Kim [6], pages 40 – 50.

[3] M. Branchaud and J. Linn. Extended Validation Models in PKI: Alternatives and Implications. In

1st Annual PKI Research Workshop – Proceedings, April 2002.

[4] T. Dierks and C. Allen. *The TLS Protocol Version 1.0*. ftp://ftp.isi.edu/in-notes/rfc2246.txt, January 1999. RFC 2246.

[5] S. Kent and R. Atkinson. *Security Architecture for the Internet Protocol*. ftp://ftp.isi.edu/in-notes/rfc2401.txt, November 1998. RFC 2401.

[6] Kwangjo Kim, editor. *Proceedings of First International Workshop for Asian PKI*, ICU, Daejeon, Korea, 19–20 October 2001. International Research Center for Information Security, Korea and Institute of Industrial Science, Japan.

[7] M. Myers and C. Adams and S. Farrell. *Delegated Path Validation*, August 2000. Expired February 2001.

[8] A. Malpani, R. Housley, and T. Freeman. *Simple Certificate Validation Protocol*. ftp://ftp.rfc-editor.org/in-notes/internet-drafts/draft-ietf-pkix-scvp-10.txt, October 2002. draft-ietf-pkix-scvp-10.txt.

[9] M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams. *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*. ftp://ftp.isi.edu/in-notes/rfc2560.txt, June 1999. RFC 2560.

[10] D. Pinkas. *Certificate Validation Protocol*. ftp://ftp.rfc-editor.org/in-notes/internet-drafts/draft-ietf-pkix-cvp-01.txt, October 2002. draft-ietf-pkix-cvp-01.txt.

[11] D. Pinkas and R. Housley. *Delegated Path Validation and Delegated Path Discovery Protocol Requirements*. ftp://ftp.isi.edu/in-notes/rfc3379.txt, September 2002. RFC 3379.

[12] Selwyn Russell. Recursive Certificates: a New Paradigm for PKI Certificates. In *Proceedings of Second International Workshop for Asian Public Key Infrastructure*, pages 14 – 20. Chinese Cryptology and Information Security Association, Taiwan, 30 October – 1 November 2002.

[13] Selwyn Russell, Eiji Okamoto, Ed Dawson, and Javier Lopez. Improving Performance in Global PKI using Virtual Certificates and Synthetic Certificates. In *Proceedings of 2002 Symposium on Cryptography and Information Security*, pages 805 – 810, Shirahama, Japan, 29 January – 1 February 2002. The Institute of Electronics, Information and Communication Engineers, Japan.

[14] Satoru Tezuka. Trend of Japanese PKI and International Cross Certification. In Kim [6], pages 22 – 31.