Efficient Offline Path Validation

Byoungcheon Lee[1], Kwangjo Kim[1], Moonseog Seo[2], and Weonkeun Huh[2]

[1] Information and Communications University,
58-4, Hwaam-dong, Yusong-gu, Taejon, 305-732, Korea
{sultan,kkj}@icu.ac.kr
[2] SECUi.COM,
647-9, Yeoksam-dong, Kangnam-gu, Seoul, 135-080, Korea
{krmsseo,abcxyz}@secui.com

Abstract. To decide whether or not to trust a certificate of a user, the certification path from the root CA (RCA) to the end CA (ECA) should be verified together with the user certificate itself. Because CAs are a kind of stable and responsible entities, path validation can be treated in a special way different from the verification of end user certificates.
In this paper we propose two efficient offline path validation schemes using trusted entities. In the first scheme, ECA has additional role of executing the path validation operation by himself and publishing the result as a self path validation statement (SPVS). In the second scheme, a trusted server executes the path validation operations for all CAs in the domain of RCA and publishes the result as a revoked CA certificate list (RCACL). End users can skip path validation operation if he can trust SPVS or RCACL. We discuss the efficiency and trust issues of proposed schemes compared with previous ones.

Keywords: certificate, public key infrastructure, path validation, self path validation statement, revoked CA certificate list

1 Introduction

1.1 Path Validation

Assume that a sender S transmits a signed message to a receiver R together with his certificate $Cert_S$. Then, R has to check the validity of $Cert_S$ for him to accept the signed message as authentic. Because R does not trust the end CA of the sender (ECA-S) but trusts only the root certification authority (RCA), he has to check the validity of the certification path from RCA to ECA-S together with the verification of sender's certificate $Cert_S$. Therefore, we can divide the certificate validation operation into the following two sub-processes:

- Certificate verification: verify the end user's certificate $Cert_S$ and corresponding CRL.
- Path validation: verify the validity of the certification path from RCA to ECA-S.

The computation for the certificate verification is inevitable in most cases. Since end user's identity and public key is not so stable (the end user can lose his private key, he can move to another company, he may not want to use the certificate any more, or the private key can be compromised), the CRL issued by ECA should be verified when the certificate is used. On the other hand, the computation for the path validation is avoidable in some cases. Moreover, it is redundant because many end users have to execute the same verification repeatedly.

1.2 Related works

We briefly review previous methods of path validation. The basic and first approach is that the end user executes the full verification by himself. But if a trusted entity is employed to execute the path validation instead of end users, path validation can be improved. Recently online validation methods such as OCSP or SCVP are attracting great attention.

- Full validation [CRL]: The receiver executes all the path validation process by himself and validates ECA-S based on his trust in RCA. It requires $O(n)$ verifications of certificate and CRL where n is the number of node in the certification path. The computation for path validation is redundant because many end users have to repeat the same computation.
- Cross-certification [CMP]: CAs can issue cross-certificates to certify each other, even between different domains of trust. To validate ECA-S, the receiver can trust cross-certificate issued for ECA-S by his end CA (ECA-R), if it is available. It is an efficient solution, but it depends on whether ECA-R has issued a cross-certificate for ECA-S in advance. Moreover, issuing and managing cross-certificates is a burden in PKI.
- Online Certificate Status Protocol [OCSP]: The receiver queries the status of $Cert_S$ to a trusted OCSP server and the OCSP server returns signed answer. Then he validate $Cert_S$ based on his trust in the OCSP server. The request-response pairs defined in this protocol are online revocation status (ORS), delegated path validation (DPV) and delegated path discovery (DPD). It is efficient in computation, but it requires communication delay. It can be used only when the user is connected to network and the OCSP server is available anytime.
- Simple Certificate Validation Protocol [SCVP]: The SCVP protocol allows a client to delegate certificate handling to a trusted SCVP server. The server can give a variety of valuable information about the certificate, such as whether or not the certificate is valid, a chain to a trusted certificate, and so on. It is similar to OCSP and they are competing to be an Internet standard.

1.3 Our Approach

The motivation of this paper is that CAs are different from general end users in many aspects.

– The total number of CAs will not be huge in a domain of RCA.
– CA is a kind of qualified entity; he is financially qualified, has responsibility under the law, and is liable for an accident.
– CA is a kind of stable entity; CA will keep his private key in more secure way than general end users and his business or position will be rather stable. So, revocation of CA's certificate will be very few compared with general end users.

Certification path from RCA to ECA is more stable than end user certificates, but the computation for path validation is very expensive. Therefore, to improve overall performance path validation has to be treated in a different way from the certificate verification of general end users. If a trusted entity executes the path validation as proxy and guarantees the validity of certification path, then end users can skip expensive path validation safely.

Main issues of delegated path validation are who is trusted and how the service is provided. In online path validation method OCSP/SCVP server is trusted and path validation result is provided through online communication. In cross-certification method ECA-R is trusted and the cross-certificate is published in advance by ECA-R. In this paper we consider how to improve the path validation operation in offline method.

1.4 Our Contribution

We have shown that path validation is different from certificate verification in many aspects. Based on this motivation we propose two efficient offline path validation schemes using trusted entity. In Section 2, we suggest an offline path validation using the self path validation statement which is published by ECA-S. In Section 3, we suggest another offline path validation using revoked CA certificate list which is published by a trusted entity. We compare the proposed schemes with previous results in terms of performance and trust in Section 4. Finally, we conclude in Section 5.

2 Self Path Validation Statement

If a trusted entity executes the path validation as proxy in advance and guarantees the validity of certification path, then end users can safely skip expensive path validation. One of the candidates who can serve as a trusted entity for executing path validation is ECA-S.

2.1 Definition of SPVS and CRL-SPVS

A self path validation statement (SPVS) is a signed statement of ECA that the certification path from RCA to himself is valid, i.e., his certificate is alive. Periodically ECA executes the path validation operation by himself and publishes the result as a SPVS.

Because SPVS is published periodically by ECA, it can be combined with CRL very easily. Therefore, we propose to implement SPVS as an extension of CRL. CRL-SPVS is a CRL which has a SPVS extension field. The SPVS field is a simple "Yes/No"statement. If the SPVS field has a value of "No", it represents that ECA does not state anything about the validity of the certification path, which means that end users have to validate the certification path by themselves. If it has a value of "Yes", it represents that ECA guarantees the validity of certification path and ECA will be responsible for any result if the SPVS is turned out to be flawed.

2.2 Roles of ECA

Generally, the main roles and services of CA are issuing certificates for his customers and issuing certificate revocation list (CRL). Additionally CA can issue cross-certificates for other CAs in the same domain of trust or among different domain of trust.

In this paper we suggest the following additional role of ECA to improve path validation.

- Periodically ECA executes the path validation operation by himself for the certification path from RCA to himself.
- He publishes CRL-SPVS which is a CRL with self path validation statement (SPVS) extension field.
- When any argument occurs, he has to provide proof for the validity of SPVS, i.e., the validity of certification path from RCA to himself. So he has to maintain all the proof materials (certificates and CRLs of upper CAs).
- If the SPVS is turned out to be flawed, he is responsible for any result of flawed SPVS.

The proposed additional role is very special because traditional ECA provides certification only for his customers and cannot certify the upper certification path. How can a receiver trust ECA-S? But if ECA-S guarantees the validity of his certificate and has responsibility for flawed SPVS, the receiver can trust ECA-S. SPVS is an important customer service of ECA which is demanded by general end users.

2.3 Efficient Offline Path Validation using CRL-SPVS

If a valid CRL-SPVS issued by ECA is available to the receiver, the certification path validation can be executed easily. We consider two cases. If the receiver encounters ECA-S's certificate for the first time, he cannot trust ECA-S and has to execute the path validation by himself. Although CRL-SPVS issued by ECA-S is available and SPVS field is "Yes", he cannot trust it. But if a receiver tries to update trust for ECA-S whose certificate and certification path have been verified before, he can use CRL-SPVS. If SPVS field of a valid CRL-SPVS is "Yes", he can safely skip path validation.

2.4 Profit and Risk

If ECA provides the SPVS service through CRL-SPVS, general end users can validate certification path very efficiently. Since verification of CRL is inevitable for certificate verification, path validation is executed with no extra cost. With this additional service of SPVS, ECA can make money or he will be preferred in the market than the traditional ECA which does not provide SPVS.

There is no risk in using SPVS if we agree on the additional role and responsibility of ECA as mentioned above. If ECA is honest, he does not take any risk with the SPVS service. His job of executing path validation and publishing SPVS is not difficult and is very typical (anyone will give the same result). If ECA tries to cheat and publishes a flawed SPVS on purpose, then end users can be fooled temporarily and lose their business by trusting ECA, but cheating ECA will be caught and punished. End users can prove the flaw very easily using public proof materials (certificates and corresponding CRLs of the certification path). Moreover ECA has to prove that his SPVS is not flawed. Therefore, general end users do not take any risk for using CRL-SPVS.

3 Revoked CA Certificate List

In this Section we introduce a trusted entity who executes path validation as proxy and reports the result as a revocation list.

3.1 Definition of RCACL

Revoked CA certificate list (RCACL) is a signed list of revoked CA certificates in the domain of RCA and is generated by a trusted proxy entity called RCACL server. Periodically, he executes the path validation operations for every CA certificates in the domain of RCA and publishes the result as a signed list of revoked CA certificates. The data included in RCACL is as follows.

Table 1. Data format of RCACL

The name of root CA
This update time
Next update time
Issuer of RCACL
Signature algorithm
List of revoked CA certificates
Extensions
Signature on above information

RCACL is similar to CRL because it is a signed list of revoked certificates and is published periodically, but it is different from CRL in the following sense.

– The candidates of revoked certificates listed in RCACL include all CAs in the domain of RCA while the candidates listed in CRL include all customers of ECA.
– It is generated by a RCACL server who is trusted by every users in the domain of RCA while CRL is generated by CA who is trusted only by his customers.
– The RCACL server has no authority to revoke a CA certificate, but is just a proxy agent who executes the path validation operations and reports the result. But CA has the authority to revoke a customer.
– Flawed RCACL can be proven easily and the RCACL server is responsible for his RCACL if it is flawed. But there is no flawed CRL because CA has the authority to issue CRL.

3.2 Role of RCACL Server

The RCACL server periodically executes the path validation operations for every CA certificates in the domain of RCA and publishes the result as a signed list of revoked CA certificates. Here we assume that the list of all CA certificates in the domain of RCA is available to RCACL server. When any argument occurs, he has to provide proof for the validity of RCACL. So he has to maintain all the proof materials that his RCACL is flawless. The proof materials include certificates and corresponding CRLs of all CAs. If the RCACL is turned out to be flawed, he is responsible for any result of flawed RCACL. Only a responsible entity under the law can serve as a RCACL server. In this model RCACL server is trusted by all users in the domain of RCA. He is a trusted entity, but he has no authority to revoke a certificate of CA.

3.3 Efficient Offline Path Validation using RCACL

Using RCACL path validation can be executed very efficiently in offline way. Because RCACL is trusted by every users in the domain of RCA, it can be distributed to every users very efficiently in centralized way. If a valid RCACL is available and ECA-S is not included in RCACL, a receiver R can trust ECA-S without executing the whole path validation operation by himself.

3.4 Profit and Risk

A RCACL server can make money with his additional service of publiching RCACL and general end users can validate the certification path very efficiently using RCACL.

There is no risk in using RCACL if we agree on the role and responsibility of RCACL server. If RCACL server is honest, he does not take any risk for publishing RCACL. His job is very typical and any honest party will give the same result. Because he is maintaining all the proofs, he can prove his honesty easily. If RCACL server tries to cheat and publishes a flawed RCACL (the flawed

RCACL may not include a revoked CA certificate or it may include a valid CA certificate), end users can be fooled temporarily and lose business, but cheating RCACL server will be caught and punished. End users can prove any flaw of RCACL easily and RCACL server is responsible for any result of flawed RCACL. Therefore, general end users do not take any risk in using RCACL.

4 Comparison

Figure 1 shows various path validation schemes in hierarchical PKI. In Table 2, we compare the proposed CRL-SPVS scheme and RCACL scheme with previous path validation models in terms of performance and trust.

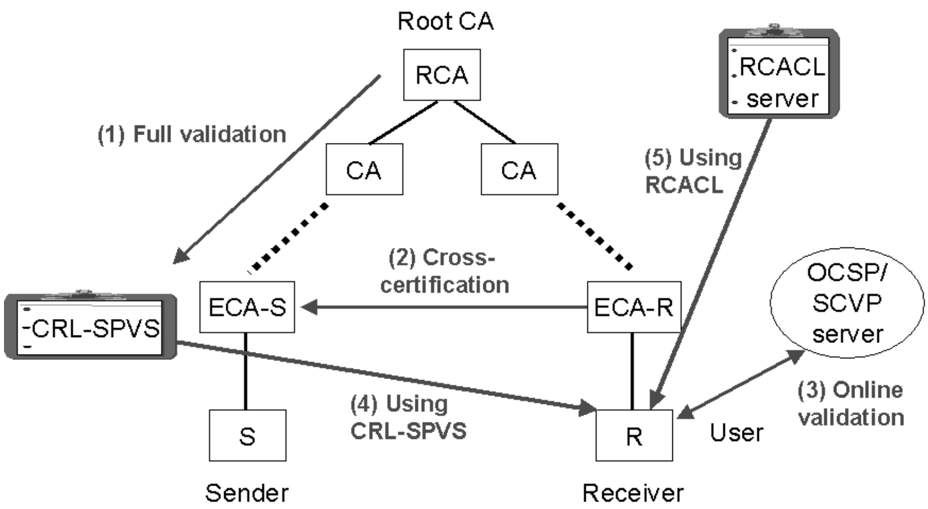

Fig. 1. Various path validation schemes in hierarchical PKI

In full validation, end users execute all the path validation operations by themselves based on the trust in RCA, which requires $O(n)$ verification of certificates and CRLs where n is the number of nodes in the certification path. But in other schemes, end users utilize the path validation result given by other trusted entities, so computation by end users is reduced to $O(1)$. CRL-SPVS model does not require any extra cost for path validation to end users because the verification of CRL-SPVS is included in certificate verification operation. Both CRL-SPVS and RCACL schemes are offline path validation where the path validation result is generated by a trusted entity in advance and given as a form of signed message. On the other hand, OCSP/SCVP models are online path validation where OCSP/SCVP server provides answer for end user's request through online communication.

Table 2. Comparison of path validation schemes

Features	Full validation	Cross certification	Online validation	CRL-SPVS scheme	RCACL scheme
Comm. model	Offline	Offline	Online	Offline	Offline
Comp. by end user	$O(n)$	$O(1)$	$O(1)$	No cost	$O(1)$
Service model	-	Distributed	Distributed	Distributed	Centralized
Trust point	RCA	ECA-R	OCSP/SCVP server	ECA-S (trust update)	RCACL server

In terms of service model, RCACL service can be provided in a centralized way because a single RCACL server is enough in a domain of RCA. But in other models path validation service is provided in distributed way. OCSP/SCVP service should be provided in distributed way because it requires online communication between end users and server and a single server cannot cover all end users in the domain of RCA.

In terms of trust, only RCA is trusted in the full validation model, but ECA-R, OCSP/SCVP server, ECA-S, and RCACL server are trusted in cross certification, OCSP/SCVP, CRL-SPVS, and RCACL scheme, respectively. Full validation and cross-certification are very clear in terms of trust because RCA and ECA-R are intrinsically trusted by end users. OCSP/SCVP server and RCACL server are newly created entities who provide path validation service in online and offline way, respectively. If end users can trust them, they can use their path validation service. On the other hand, ECA-S is not trusted by end users in traditional model. But if the additional role of ECA-S (providing SPVS service) is agreed, then end users can trust ECA-S once they had verified him as valid ECA through other path validation method. One drawback of CRL-SPVS is that it can be used only for trust update not for the first-time trust. It is expected that ECA-S who provides SPVS service will be preferred in the market.

Summarizing the result, CRL-SPVS scheme and RCACL scheme are efficient offline path validation schemes compared with previous path validation models. CRL-SPVS scheme is efficient because it is provided by ECA-S and does not require any extra cost, but we have to agree on the additional role and responsibility of ECA-S. RCACL scheme is efficient in the sense that the trust model is very simple and the service is centralized.

5 Conclusion

In this paper we have introduced the necessity that path validation from RCA to ECA can be (has to be) treated in different way from the verification of end user's certificate. Based on this motivation we have proposed two efficient offline path validation schemes using trusted entities. CRL-SPVS scheme is efficient because it is provided by ECA-S and does not require any extra cost, but we have to agree on the additional role and responsibility of ECA-S. RCACL scheme is efficient in the sense that the trust model is simple and the service is centralized. Typically the best way in our common trust model is that the root CA who is trusted by every users provides RCACL as an additional service. Although we have described RCACL only in the domain of RCA, it can be used inter-domain situation if the RCACL server is trusted by users of plural domains.

In this paper we have introduced two offline path validation schemes in very simplified way. But to apply the proposed schemes in real world PKI, we have to consider several aspect of managing PKI. For example, each CA in the certification path can have different time interval of issuing CRL and different policy. In terms of RCACL we also have to consider which is more appropriate among a revocation list or an alive list.

References

[CRL] RFC 2459, Internet X.509 Public Key Infrastructure Certificate and CRL Profile, IETF, 1999, http://www.ietf.org/html.charters/pkix-charter.html

[CMP] RFC 2510, Internet X.509 Public Key Infrastructure Certificate Management Protocols, IETF, 1999.

[OCSP] RFC 2560, Internet X.509 Public Key Infrastructure Online Certificate Status Protocol - OCSP, IETF, 1999.

[SCVP] Internet draft, Simple Certificate Validation Protocol, IETF, 2001.