Butterfly key expansion 알고리즘을 적용한 ECQV에 관한 연구

선설희* · 김은기*

*국립 한밭대학교

A study on ECQV applied the butterfly key expansion algorithm

Seol-hee Sun* · Eun-gi Kim**

*Hanbat National University

E-mail : whrhghgh@naver.com

요 약

ECQV(Elliptic Curve Qu-Vanstone)는 ECC(Elliptic Curve Cryptography) 기반의 묵시적 인증서(implicit certificate)를 생성하는 방법으로서 기존에 사용되던 명시적 인증서에 비해 크기가 작고 빠르기 때문에 메모리가 충분하지 않거나 대역폭이 작은 제한된 통신환경에서 유용하게 사용 될 수 있다. 또한, Butterfly key expansion 알고리즘은 하나의 공개키를 사용하여 여러 개의 인증서를 발급할 수 있도록 지원하는 방법이다. 본 연구에서는 기존 ECQV에 Butterfly key expansion 알고리즘을 적용하여, 추후 자동차 통신 환경에서 유용하게 사용될 수 있는 인증서 발급 방법을 제안한다.

ABSTRACT

The ECQV(Elliptic Curve Qu-Vanstone) is a implicit certificate scheme based on ECC(Elliptic Curve Cryptography). Implicit certificates are smaller and faster than a traditional explicit certificate. Therefore, it can be used in a memory or bandwidth constraint communication environments. Also, the butterfly key expansion algorithm is a method to issue many certificates by using only one public key. In this study, by applying the butterfly key expansion algorithm to ECQV, we suggest a new useful issuing certificate method that can be used in vehicular communication environments.

키워드

ECQV, Butterfly key expansion algorithm, implicit certificate, vehicular communications

Ⅰ. 서 론

최근 자율 주행 차 기술이 빠르게 진화하면서 차량 통신 시스템과 관련된 보안 연구 또한 지속적으로 이루어지고 있다. 그중에서도 방대한 양의 인증서를 발급하고 관리하기 위해 Butterfly key expansion 알고리즘을 적용한 PKI 시스템이 연구된 바 있다[1]. Butterfly key expansion은 하나의 공개키를 사용하여 여러 개의 인증서를 발급할 수 있도록 지원하는 방법이다. 따라서 본 연구에서는 이를 ECC 기반의 묵시적 인증서인 ECQV에 적용하여 메모리가 충분하지 않거나 대역폭이 작은 자동차 통신에서 유용하게 사용될 수 있는 인증서 발급 방법을 제안한다.

Ⅱ. 본 론

2.1 original ECQV

기존의 명시적 인증서를 발급받고자 하는 호스트는 공개키와 개인키를 생성한 후, 자신의 공개키를 CA(Certificate Authority)에게 보낸다. 이를 수신한 CA는 호스트의 공개키와 자신의 전자서명을 포함한 인증서를 호스트에게 발급한다[2]. 반대로 묵시적 인증서는 호스트가 CA로부터 발급받은 인증서를 통해 호스트의 공개키와 개인키를 추출할 수 있다. 따라서 묵시적 인증서의 한 종류인 ECQV는 기존의 명시적 인증서와 비교했을 때, 그 크기가 상당히 작고 빠르기 때문에 메모리가 충분하지 않거나 대역폭이 작은 제한된 통신환경에서 유용하게 사용된다[3]. 다음 그림 1은 ECQV

묵시적 인증서의 발급과정을 자세히 나타낸다.

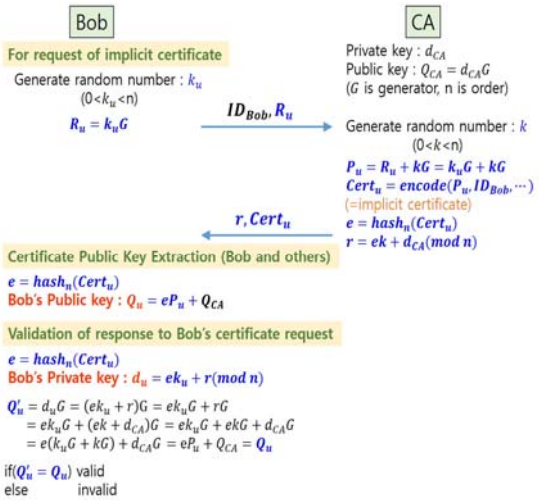

그림 1. 기존 ECQV 발급과 키 추출과정

위 그림 1에서 CA는 Bob으로부터 받은 R_u와 자신의 랜덤 값 k를 이용하여 P_u를 계산한다. P_u는 ID_{Bob}과 함께 묵시적 인증서에 포함된다. 따라서 CA의 공개키인 Q_{CA}를 알고 있는 모든 호스트가 Bob의 묵시적 인증서를 받으면 Bob의 공개키를 계산할 수 있다. 그리고 Bob은 CA에게 발급받은 인증서와 랜덤 값 k_u를 사용하여 Bob의 개인키를 추출하고 인증서의 유효성을 확인할 수 있다[4].

2.2 Butterfly key expansion 알고리즘

호스트가 많은 양의 인증서를 발급받고 관리하기 위해서는 모든 인증서에 대한 공개키와 개인키를 저장하고 있어야 하므로 메모리가 제한된 통신환경에서 한계가 발생할 수 있다. 따라서 하나의 공개키를 사용하여 여러 개의 인증서를 발급할 수 있는 Butterfly key expansion 알고리즘을 적용한 PKI 시스템이 연구되었다[1].

이 알고리즘은 ECC 기반에서 CA가 호스트로부터 전송받은 하나의 공개키를 확장 함수 $f_k(i)$와 계산 후, 그 결과 값을 자신의 랜덤 값과 생성자 G와 함께 연산하여 여러 개의 Butterfly 공개키를 생성하고, 이를 사용하여 여러 개의 인증서를 발급할 수 있는 방법이다. 여기서 확장 함수 $f_k(i)$는 AES와 같은 암호화 방식을 사용한다. 예를 들어, NISTp256curve에 있는 한 점을 호스트의 공개키와 개인키로 사용한다면 $f_k(i) = AES_k(0^{128} XOR\ i) \| AES_k(1^{128} XOR\ i) \| \cdots$이 된다. 이때 k는 AES 암호화 함수에서 별도로 사용되는 입력키이며 $i < 2^{128}$이다[1,5].

아래 그림 2는 Butterfly key expansion 알고리즘을 설명한다.

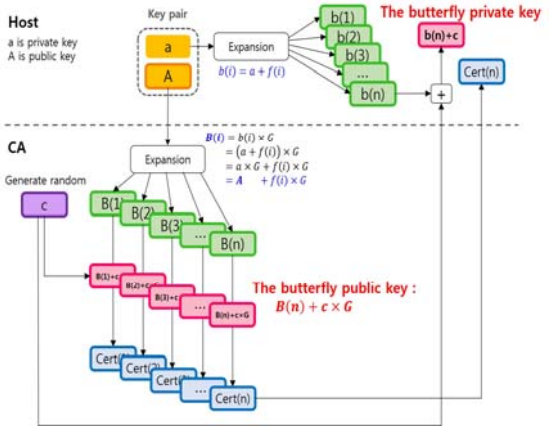

그림 2. Butterfly key expansion 알고리즘

위 그림 2에서 CA는 Host의 인증서 요청으로 받은 공개키 A와 확장 함수 $f_k(i)$를 사용하여 여러 개의 B(n)을 생성한다. 이때 B(n)은 발급할 인증서의 개수만큼 생성된다. 따라서 공개키 A와 확장 함수 $f_k(i)$만 알고 있으면 누구나 B(n)을 계산할 수 있다. 그리고 CA는 랜덤 값 c를 사용하여 위 그림과 같이 여러 개의 Butterfly 공개키를 생성한 후, 이를 인증서 Cert(n)에 각각 포함하여 c와 함께 Host로 전송한다. 인증서를 발급받은 Host는 c를 이용하여 각 인증서에 해당하는 Butterfly 개인키를 계산할 수 있다.

2.3 Butterfly key expansion 알고리즘을 적용한 ECQV

본 연구에서는 기존의 명시적 인증서에 비해 크기가 작고 빠른 묵시적 인증서의 발급 방법인 ECQV에 하나의 공개키로 여러 개의 인증서를 발급할 수 있는 Butterfly key expansion 알고리즘을 적용하여 메모리의 효율을 더욱 높일 수 있게 하였다. Butterfly key expansion이 적용된 ECQV의 발급과정은 그림 3에서 자세히 설명한다.

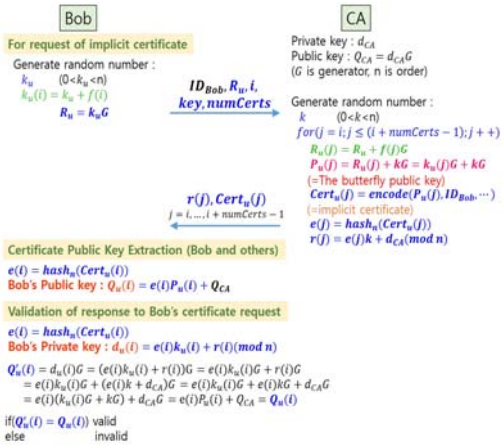

그림 3. Butterfly key expansion 알고리즘을 적용한 ECQV 발급과 키 추출과정

그림 3에서 numCerts는 발급 받고자 하는 인증서의 개수를 나타내고, 랜덤 값인 i는 확장 함수 $f_k(i)$의 입력 값으로 사용된다. 또한, 확장함수 $f_k(i)$는 AES와 같은 암호화 함수를 사용하기 때문에 이를 위한 키가 별도로 필요하다. 따라서 위 그림 3에서 key는 확장 함수의 키로 사용된다.

인증서 발급 요청을 받은 CA는 numCerts만큼 Butterfly 공개키를 생성하고, 이를 각각 포함시킨 묵시적 인증서를 생성한다. 이후 여러 개의 인증서를 발급받은 Bob은 기존의 ECQV와 같은 과정으로 인증서의 유효성을 확인하고 공개키와 개인키를 추출할 수 있다.

III. 결 론

본 연구에서는 기존의 명시적 인증서보다 크기가 작고 빠른 묵시적 인증서의 발급 방법인 ECQV에 하나의 공개키로 여러 개의 인증서를 발급할 수 있는 Butterfly key expansion 알고리즘을 적용하여 메모리의 효율성을 높일 수 있는 방법을 제안하였다. 이는 메모리가 충분하지 않거나 대역폭이 작고, 빠른 속도를 요구하는 자동차 통신환경에서 유리하게 사용될 수 있을 것으로 예상된다.

추후에는 본 연구에서 다룬 내용을 ECQV SEC4 규격을 기반으로, OpenSSL 라이브러리가 제공하는 API를 사용하여 소프트웨어로 구현하고 성능분석을 할 계획이다.

ACKNOWLEDGMENTS

This research was financially supported by the Ministry of Trade, Industry and Energy (MOTIE) through the Promoting Regional specialized Industry (No. R0003847)

참고문헌

[1] William Whyte, A Security Credential Management System for V2V Communications, IEEE Vehicular Networking Conference, pp. 1-8, Dec. 2013.

[2] Wikipedia, Public key infrastructure [Internet], Available:http://en.wikipedia.org/wiki/Public_key_infrastructure.

[3] Wikipedia, Implicit certificate [Internet], Available:http://en.wikipedia.org/wiki/Implicit_certificate.

[4] SEC, Elliptic Curve Qu-Vanstone Implicit Certificate Scheme (ECQV), SEC std 4, 2013.

[5] National Institute of Standards and Technology. (1999, July) Recommended elliptic curves for federal government use [Internet], Available:http://csrc.nist.gov/groups/ST/toolkit/documents/dss/NISTReCur.doc