유·무선 통합검증을 위한 DVCS S/W 개발

차상희* 강명호* 박종욱** 이 용**

*장미디어 인터렉티브, **한국정보보호진흥원

Development of DVCS S/W for wired/wireless Environment

Sang Hee Cha* Myeong Ho Kang* Jong Wook Park** Young Lee**

*Jang Media Interactive, **KISA

요 약

본 논문에서는 데이터 유효성 검증 작업을 수행하는 서버인 DVCS의 효율적인 설계에 대하여 고찰하고, 나아가서 무선 환경에서의 효율적인 DVCS 프로토콜 적용에 관하여 살펴본다. DVCS는 전자서명된 데이터의 유효성 검증 및 특정 시점에서의 인증서 유효성 검증을 위하여 과거에 발행된 CRL들에 대한 분석을 필요로 하는데, 이는 시간 및 공간적인 복잡도가 높은 작업이므로 효율적인 시스템 설계가 요구된다. 또한, 성능이 제한적인 무선 단말기에서 DVCS 프로토콜 및 메시지 처리를 가능하게 하기 위하여, 무선 환경에서의 요구사항 및 이에 대한 해결책을 제시하도록 한다.

I. 서론

DVCS는 데이터 유효성 검증을 요구하는 클라이언트로부터 특정 서비스에 대한 요청을 받은 후, 해당 서비스에 대한 응답을 위하여 데이터 검증 인증서(Data Validation Certificate : DVC)를 생성하는 시스템이다. DVCS는 현재 또는 지정된 특정 시각에 전자서명된 문서 및 인증서의 유효성을 인증 및 검증해 주는 시스템으로서, RFC 3029에 메시지 형식 및 프로토콜이 정의되어 있다.

DVCS의 서비스는 크게 검증 서비스와 인증 서비스로 나눌 수 있다. 인증 서비스는 데이터에 대한 실제 소유를 인증하는 서비스와 데이터 소유에 대한 주장을 인증해 주는 서비스로, 검증 서비스는 전자서명된 데이터에 대한 검증 서비스와 공개키 인증서에 대한 검증 서비스로 나뉜다.

일반적으로, 데이터 유효성 검증을 위한 작업은 시간 및 공간적인 복잡도가 높은 작업이며, 특히, 효력정지 후 회복되었던 인증서에 대한 검증 작업은 기존의 PKI 구성 요소만으로는 해결할 수 없는 작업이므로, 해당 인증기관이 발행했던 모든 CRL들에 대한 효율적인 분석 작업에 대한 심도 있는 연구가 필요하다.

또한, 상대적으로 낮은 성능의 무선 단말기 환경에서 DVCS 서비스를 가능하게 하기 위하여, 무선 환경을 위한 선택적인 서비스 체계 및 부가적인 프로토콜의 설계가 요구된다.

II. DVCS

1. DVCS 서비스 유형

RFC 3029에서 정의하는 DVCS 서비스 유형은 모두 네 가지이며, 이 서비스들은 문서 및 인증서에 대한 유효성을 인증 또는 검증해주는 서비스이다. 각 서비스들은 다음과 같이 정의된다.

1) CPD 서비스

CPD 서비스는 요청자가 지정된 시각에 데이터를 소유하고 있었음을 인증해 주는 서비스이며, 실제 자료가 데이터 검증 서버(data validation server, DVS)에게 전달되었다는 것을 입증해 주는 서비스이다. CPD 서비스는 문서 저장소를 위한 신뢰 기관 서비스에 사용될 수 있다.

2) CCPD 서비스

CCPD 서비스는 요청자가 지정된 시각에 데이터를 소유했었음을 인증해 주는 서비스이며, 데이터 자체가 데이터 검증 서버에게 전달되었음을 인증하는 것이 아닌, 데이터의 메시지 요약값(message digest)이 서버에 전달되어 입증 받는 서비스이다. CCPD 서비스는 타임 스탬핑 서비스에 이용될 수도 있다.

3) VSD 서비스

VSD 서비스는 디지털 서명 문서에 대한 검증을 제공하는 서비스이며, 서명된 문서의 유효성을 명백히 하기 위한 서비스이다. DVCS는 모든 필요한 상태 정보와 공개키 인증서를 이용하여 서명된 문서에 첨부되어 있는 모든 서명이 유효한지 확인한다.

4) VPKC 서비스

VPKC서비스는 공개키 인증서의 유효성 검증 및 유효성 주장에 사용되는 데이터 검증 인증서를 생성해 주는 서비스이다. 공개키 인증서의 유효성 확인은 RFC 2459에 의거하여 특정 시각에서 하나 또는 그 이상의 공개키 인증서의 유효성을 확인하기 위하여 사용된다.

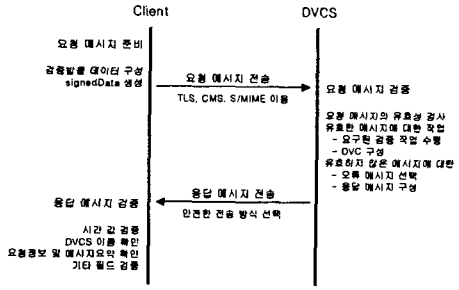

그림 1 DVCS 프로토콜

2. DVCS 트랜잭션

1) DVCS의 기능적 요구사항

DVCS가 요구에 따른 정상적인 응답을 하기 위해서는 다음과 같은 기능이 요구된다.

- 서명된 형태의 응답 또는 오류 응답 기능

- DVC의 내용 명시 기능 및 검증 여부 명시 기능

- DVC에 단조 증가하는 일련번호 (serial number) 기록 기능

- 시간 값 및 타임 스탬프 토큰(Time Stamp Token) 사용 기능

- 자신의 서명 키에 대한 검증 기능

2) DVCS 시나리오

DVCS 서비스는 클라이언트의 요청에 의하여 시작되며, 각 프로토콜은 그림1과 같은 흐름을 갖는다.

III. DVCS 설계

1. 인증서 상태 관리

시간에 따른 인증서의 상태는 DVCS 서비스를 제공하는데 가장 중요한 정보이다. 이런 인증서의 상태는 CA의 CRL이나 OCSP를 통해서 얻을 수 있다. 하지만 이들을 처리하는 것에 몇 가지 문제점이 존재한다.

1) 문제점

① 인증서 효력 정지 및 회복

인증서가 효력 정지되었을 때에는 CRL에 등록되지만 해당 인증서의 효력이 다시 회복되었을 때에는 CRL에서 삭제된다. 즉, CRL에서는 효력 정지에 대한 이력이 관리되지 않으므로, CRL 만으로는 DVCS가 올바른 VPKC 서비스를 제공하기 힘들다.

② CRL 처리 성능

CRL에 등록된 인증서의 수가 늘어나면 이를 처리하기 위한 계산량이 늘어나며 성능상의 문제가 발생하게 된다.

2) 해결 방안

인증서의 효력 정지에 관한 이력은 PKI를 구성하는 다른 서비스에서 관리되지 않으므로, DVCS 서버 자체에서 효력 정지에 대한 이력을 포함한 인증서 상태 관리 기능을 제공해야만 한다. 즉, 자체 데이터베이스를 구축하여 지속적으로 발행되는 CRL을 검토·관리하여 인증서의 효력 정지 상태를 관리하도록 설계하며, 효과적인 구축을 위하여 다음과 같은 방법을 고려할 수 있다.

① Delta-CRL의 사용 : 해당 CA에서 delta-CRL의 발행이 가능하다면, 이를 이용하여 이전 CRL에서 변경된 부분만 처리하게 되므로 성능상의 문제가 해결될 수 있다.

② 상태관리를 위한 전용 프로토콜 사용 : CA 또는 OCSP와의 전용 프로토콜을 사용하여 지속적으로 인증서의 변경 상태를 관리하여 보다 정확한 정보를 유지하도록 설계할 수 있다.

2. 무선 환경에서의 고려사항

무선 단말기 환경에서는 여러 가지 제한 사항이 존재한다. 특히, 통신 대역이 협소하고 무선기기의 낮은 성능은 암호학적 계산에 많은 어려움을 준다. 이런 제한 사항을 바탕으로 고려할 수 있는 문제점은 다음과 같다.

① 통신 대역과 기기 성능의 문제로 큰 데이터 처리가 문제된다.

② 요청 메시지 생성 및 응답 메시지 처리를 위한 전자서명 생성 등 암호학적 계산에 많은 시간이 소요된다.

③ 인증서를 검증하기 위한 CRL 처리 기능이 문제된다.

이와 같은 문제점을 바탕으로, 유·무선 통합 환경에서 DVCS 서비스의 원활한 제공을 위하여

다음과 같은 해결 방안을 모색할 수 있다.

① DVCS의 서비스들 중 무선 환경에 적합한 서비스를 우선적으로 제공한다. 즉, CPD와 같이 자료 전송이 많은 서비스는 무선 단말기의 성능을 고려하여 제외시키도록 한다.

② DVCS 트랜잭션 처리가 가능한 최소한의 기기 스펙을 제시한다.

③ DVC의 서명에 사용된 인증서에 대한 검증 시, OCSP나 DVCS의 VPD 서비스를 이용하여 무선 단말기의 부하를 최소화 한다.

3. 시스템 설계

1) 전체 구성도

각 구성요소들은 DVCS 서버를 중심으로 클라이언트들과 서비스 서버들로 구성되어 있다. 각 구성 요소들의 기능은 다음과 같다.

- 유/무선 클라이언트 : DVCS 서비스를 요청하고 그 응답을 얻는 요소

- DVCS 서버 : 클라이언트로부터 서비스 요청을 접수하여 응답을 보내는 요소로서 인증서 상태관리를 위한 자체 DB를 포함함

- CA : 인증서 발급 서버로써 인증서의 상태에 대한 정보를 제공

- OCSP : 인증서의 실시간 상태를 지속적으로 제공

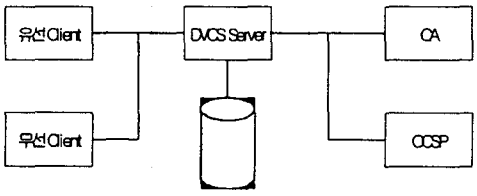

그림 2 전체구성도

2) 서비스 시나리오

DVCS의 모든 서비스들은 동일한 메시지 전달 흐름을 가지고 있다. 그림 3은 DVCS 프로토콜 상에서의 메시지 흐름을 나타낸다.

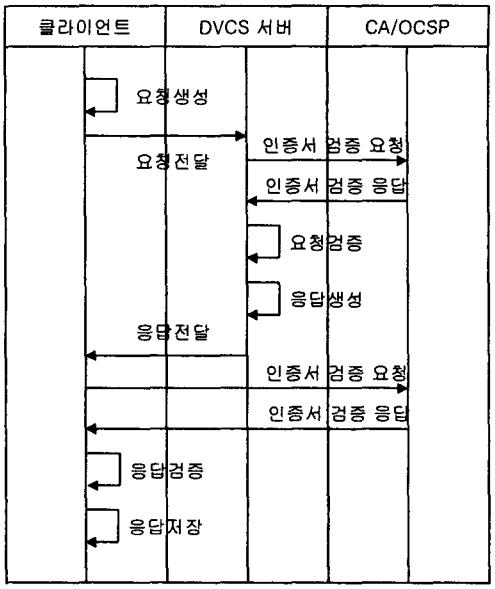

그림 3 서비스 시나리오

3) 인증서 상태 관리

DVCS 서비스의 성능에 가장 큰 영향을 주는 부분은 인증서 상태 관리 부분이며, 앞 장에서 언급한 바와 같이 기존의 PKI 요소들 만으로는 관리되지 못하는 부분이다.

특히, 인증서 폐지 및 회복에 관한 안전한 상태관리를 위해서는, 인증기관이 발행하는 모든 CRL을 관리·조회해야 하며, 이를 위한 작업이 DVC 서버에 가장 큰 부하를 주게 된다.

본 논문에서는 보다 효율적인 인증서 상태 관리를 위하여, delta-CRL을 사용하도록 DVCS를 설계하였으며, 매회 발행되는 delta-CRL을 기반으로 DVCS 내에 자체적인 데이터베이스를 구축·관리하는 방법을 적용하여 구현하였다.

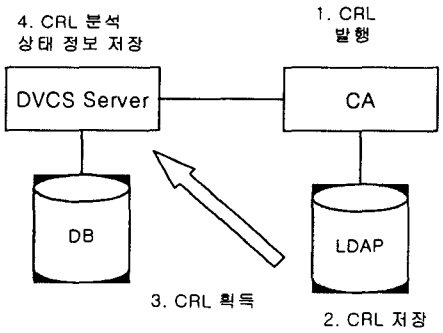

그림 4 CRL 처리과정

그림 4에 도시된 바와 같이, 인증기관으로부터 발행된 delta-CRL을 처리하여 모든 인증서의 상

태를 DVCS 내의 데이터베이스에 저장하도록 구성하였다.

일반적으로, 인증서의 상태 변경은 빈번하게 이루어지는 작업이 아니며, 기업 수준의 환경에서는 CRL 발행 주기 내에 모든 변경된 인증서 처리가 충분히 가능하기 때문에, 설계·구현된 시스템의 처리 용량의 측면에서 안정적이라 판단된다.

IV. 결론

DVCS는 PKI 환경에서 전자서명 및 공개키 인증서에 대한 유효성을 검증하고 이에 대한 인증을 담당하는 신뢰 기관 서버이며, RFC 3029는 이러한 서비스의 제공시 사용되는 프로토콜 및 메시지 형식을 규정하고 있다.

본 논문에서는 RFC 3029 문서를 중심으로 DVCS 서비스 및 프로토콜을 고찰하고, 안전하고 효율적인 인증서 상태관리를 위한 방법 및 무선 환경에서의 DVCS 적용 방안을 제시하였다. 특정 시각에서 정확한 인증서 상태 확인을 위해서는 효력 정지 및 회복에 대한 정보가 요구되며, 이를 위하여 DVCS 자체 내의 상태 정보 데이터베이스를 구성하였다. 또한, 유·무선 통합 환경에서 원활한 DVCS 서비스를 제공하기 위하여 무선 환경에서의 제약 사항을 제시하였다.

이와 함께, DVCS를 위한 서버 프로그램 및 클라이언트 프로그램을 위한 모듈을 설계하고, 설계된 내용을 바탕으로 DVCS의 네가지 기본 서비스에 대한 프로토타입을 구현하였다.

참고문헌

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

[2] C. Adams and S. Farrell, "Internet X.509 Public Key Infrastructure, Certificate Management Protocols," RFC 2510, March 1999.

[3] M. Myers, C. Adams, D. Solo and D. Kemp, "Internet X.509 Certificate Request Message Format," RFC 2511, March 1999.

[4] R. Housley, "Cryptographic Message Syntax," RFC 2630, June 1999.

[5] ISO/IEC 10181-5: Security Frameworks in Open Systems. Non-Repudiation Framework.

[6] C. Adams, P. Sylvester, M. Zolotarev and R. Zuccherato, "Internet X.509 Public Key Infrastructure, Data Validation and Certification Server Protocols," RFC 3029, February 2001.

[7] C. Adams, P. Cain, D. Pinkas and R. Zuccherato, "Internet X.509 Public Key Infrastructure, Time-Stamp Protocol," RFC 3161, August 2001.