OCSP를 이용한 PKC 및 AC 검증방안

김 영진, 지 준웅, 김 지홍

세명대학교 정보보호학과, 세명대학교 대학원 전기전자공학과

The Verification Method for PKC and AC Using OCSP

Kim YoungJin, Chi JunWoong, Kim JiHong

Department of Electronics and Electrical Semyung University

요 약

정보통신기술의 발달로 인터넷상의 PKC를 사용한 전자거래가 활성화되었다. 이에따라 실질적으로 Web Server나 Database Server에 접속하기 위한 접근통제의 방안으로 속성인증서에 대한 연구도 활발히 진행되고 있다. 그러나 현재 제안되고 있는 CRL(Certificate Revocation List) 및 OCSP를 이용한 공개키인증서 검증방법은 속성인증서의 인증상태확인과 적용시킬 수 없다. 따라서 본 논문에서는 기존의 공개키인증서 검증방법인 OCSP 방법에 속성인증서 검증방법을 포함시킴으로서, 공개키인증서와 속성인증서간의 동기문제를 해결하고자 한다.

1. 서론

정보통신 기술의 발달로 사회의 모든 분야에서 인터넷의 활용이 급속히 확산되어 전자상거래, 인터넷 뱅킹 등의 편리한 서비스가 제공되고 있다. 그러나 인터넷을 이용한 모든 거래는 거래당사자간 비접촉, 비대면을 특징으로 하기 때문에, 온라인상의 편리함을 추구할 수 있는 반면에 거래 당사자간의 상호신뢰에 있어서 취약성을 가진다. 이러한 단점을 해결하기 위하여 전세계적으로 공개키기반구조(PKI : Public Key Infrastructure)라는 인증기반구조[7]를 도입함으로서 거래당사자들 간의 신뢰성과 안전성을 추구하고 있다. PKI는 계층구조 형식의 인증기반구조를 채택함으로서, 하위 계층의 인증기관 혹은 사용자에게 공개키인증서(PKC : Public Key Certificates)를 발급하고, 이를 이용하여 온라인상의 안전한 전자거래를 할 수 있도록 하는 방식이다. 따라서 PKI 상에서의 모든 사용자는 공인 인증기관(AC : Certification Authority)으로부터 사용용도에 부합되는 PKC를 발급받고, 이를 이용하여 자신이 정당한 사용자임을 입증할 수 있다.

그러나 이러한 PKC를 이용한 기술은 공개키 정보를 이용하여 사용자 인증정보를 제공하므로 비대면 인터넷 통신에서의 사용자 신원을 입증하기 위해서 유용하게 사용될 수 있지만, 실제 시스템에서의 접근통제를 위한 정보는 포함하고 있지 않으므로, 접근통제를 필요로 하는 분야에서는 속성인증서(AC : Attribute Certificates)와 같은 별도의 형태의 인증서를 이용한 구조가 제안되고 있다.

AC는 속성인증당국(ACA : AC Authority)에서 발급하는 사용자의 속성정보를 저장하는 인증서로서, 사용자의 지위, 권한, 임무 등과 같은 다양한 권한정보를 제공한다. AC에 대한 연구는 ITU-T, IETF 등에서 진행되고 있으며, IETF에서는 Internet Draft 문서[4]와 RFC 3281[6]를 통하여 표준화가 진행되고 있다. 이러한 AC는 사용자의 속성정보와 같은 유용한 정보를 저장하고 있지만, 사용자에 대한 공개키 정보를 가지고 있지 않다. 따라서 AC를 접근통제 분야에 적용하기 위해서는 공개키 기반구조상의 PKC를 첨부하여 접근하거나, 혹은 PKC와 AC를 결합한 형태에 관한 많은 연구가 진행되고 있다[2,3].

본 논문은 이와 같이 접근통제를 요구하는 응용시스템에서 PKC와 AC를 검증하는데 있어서 발생되는 문제점을 밝히고, 이를 해결하기 위하여 기존의 PKC 인증서 검증에 사용된 OCSP 서버의 데이터구조를 변형하여 AC 검증에 사용할 수 있도록 함으로서, PKC와 AC간의 동기성 문

제를 해결할 수 있는 방법을 제안한다.

2. 기초이론

2. 1 AC(Attribute Certificates)

인증서(Certificate)란 여권과 같이 자기 자신의 신분을 증명하기 위해 사용되는 증서로서, 인터넷상에서 신뢰성있는 통신을 위하여 개개인을 입증하기 위해 사용된다. 이와같이 인터넷상에서 사용되는 인증서는 크게 PKC, AC, SPKC로 분류할 수 있다. PKC란 현재 범용적으로 사용되고 있는 PKI에서 사용되고 있는 인증서를 말한다. 본 논문에서는 PKC[7]와 SPKC에 관한 설명은 생략한다. AC는 사용자의 속성정보를 저장하는 인증서로서, 사용자의 지위, 권한, 임무 등과 같은 다양한 권한정보를 제공하며, AC에 대한 표준에는 IETF(Internet Engineering Task Force)와 ITU-T(International Telecommunication Union)가 있다. IETF에서 제안된 AC 표준은 RFC 3281[6]이며, ITU-T와 ISO/IEC에서는 X.509 v4.0 AC와 권한기반구조인 PMI(Privilege Management Infrastructure)에 대하여 기술하고 있다. 두 개의 표준은 상당부분 유사한 구조를 가지고 있으며, IETF에서 제안된 AC에 대한 형식[6,7]은 표 2-1과 같다.

표 0-1 AC 형식

기본영역	사용용도
버전	X.509 V2.0인 경우 "2"
사용자(holder) 이름	AC 사용자이름(X.500 이름)
발급자(issuer) 이름	AC 발급자이름(X.500 이름)
서명알고리즘	서명알고리즘ID 및 관련파라미터
일련번호	AC 일련번호
유효기간	시작일자와 만료일자
속성정보	사용자의 속성정보
발급자 고유 ID	발급자에 대한 부가정보
확장자	AC에 대한 부가정보
서명문	인증서발급자의 서명문

AC의 구성은 기본적으로 PKC 형식과 유사하다. 그러나 사용자의 공개키를 포함하고 있지 않으며, 사용자의 속성정보는 다음과 같다.

- Service Authentication Information : 사용자 ID 및 비밀번호
- Access Identity : AC holder에 대한 정보
- Charging Identity : 과금을 위한 정보
- Group : 사용자가 속한 그룹
- Role : roleAuthority, 사용자의 역할
- Clearance : 보안인가등급

AC는 서버 혹은 데이터베이스 등 시스템 자원에 대한 접근통제를 목적으로 하기 때문에, 인증서 발급주기를 가급적 짧게 하고, CRL(Certificate Revocation List)은 가능하면 사용하지 않는 것을 권장하고 있다.

2. 2 AC를 이용한 응용서버 접속방법

AC 분배 방식으로 Pull 방식과 Push 방식이 있다.

가. Pull 방식을 이용한 접근제어 모델

Pull 방식은 서버 및 클라이언트에 설치된 응용프로그램에 의해 동작되는 방식이다. Pull 방식은 클라이언트의 서비스 요구가 있을 경우에, 응용서버는 클라이언트의 접속요구에 대한 권한 여부를 확인하기 위하여 속성인증기관으로 접속하여 AC를 획득하는 방법이다.

그림 2-1은 Pull 방식을 이용한 서버 접속방식을 도식화하고 있으며, 이와 같은 절차는 다음과 같다.

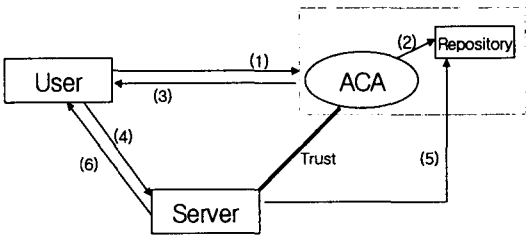

그림 2-1 Pull 방식을 이용한 AC 획득방법

① 사용자는 ACA에 AC를 요청한다. 이때 사용자는 미리 발급받은 자신의 PKC를 첨부하여

ACA로 보낸다.

② 사용자의 접근권한 확인과정을 거친 후, 발급된 AC는 ACA의 별도 저장소에 보관한다.

③ ACA는 사용자에게 AC가 발급되었음을 알려준다

④ 사용자가 서버에 접속하기 위하여 서비스 요구 패킷을 보낸다.

⑤ 서버에서는 사용자의 서비스 요구를 수신하고, ACA에 접속하여 사용자에 대한 AC를 획득한다. 서버에서는 ACA의 서명문을 확인하기 위하여 AC로 ACA의 PKC를 확인한다.

⑥ AC 내용의 적합성을 확인하고, 사용자에게 접속허용여부를 알려준다.

나. Push 방식을 이용한 접근제어 모델

Push 방식은 사용자가 AC를 소지하고, 서버에 접속할 경우에 자신의 AC를 제출하는 방식이다. 그림 2-2는 Push 방식을 이용한 접근제어 모델을 도식화하고 있으며, 이와같은 절차는 다음과 같다.

① 사용자는 ACA에 AC를 요청한다. 이때 사용자는 미리 발급받은 자신의 PKC를 첨부하여 ACA로 보낸다.

② 사용자의 접근권한 확인과정을 거친 후, 발급된 AC는 사용자에게 전달된다.

③ 사용자가 서버에 접속하기 위하여 자신의 AC와 서비스 요구 패킷을 보낸다.

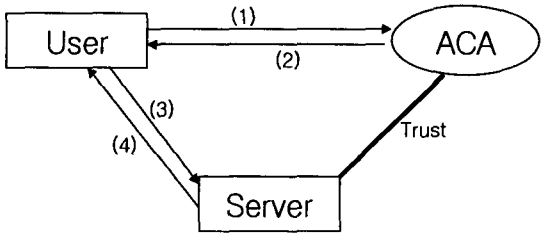

그림 2-2 Push 방식을 이용한 AC 획득방법

④ 서버에서는 사용자의 AC와 서비스 요구패킷을 수신하고, ACA의 서명문을 확인하기 위해 CA에서 ACA의 PKC를 확인한다.

⑤ 사용자에게 AC의 인증결과에 따라 접속허용여부를 알려준다.

2. 3 AC 분배 방식에 대한 비교 설명

Push 모델은 사용자가 서버에 접근할 때, 자신의 AC를 직접 전달하는 방식이고, Pull 모델에서는 AC를 ACA의 보관소를 사용하는 방식이다.

표 0-1 Push 방식과 Pull 방식

	Push 방식	Pull 방식
장점	- 서버 부하가 줄어듬 - AC 관리를 위한 저장소 설치비용 및 검색을 위한 통신비용이 필요하지 않다.	- 속성정보 변경에 유연성을 가진다. - AC 전달, 보관 과정에서 발생하는 문제점을 줄일 수 있다.
단점	- AC 직접 전달 - AC 관리면에서 분실 및 훼손의 여지가 있다.	- AC 관리를 위해 추가적인 비용이 필요하다. - 서비스 사용자의 수에 비례하여 통신 부하가 발생한다.

일반적으로 Push 모델은 서버에서 AC 검색을 위한 추가적인 통신·설치비용이 필요하지 않기 때문에 속도가 향상된다는 장점을 가지고 있으나, AC의 관리상 분실과 훼손 혹은 전달과정에서 해커나 크래커에 의해 도청될 수 있는 단점이 있다. 또한 AC의 짧은 생명주기 측면을 고려할 때, 사용자가 보관하는 것보다 신속한 갱신과정을 수행할 수 있도록 저장소를 사용하는 방식이 선호된다.

2. 4 인증서 상태 조회방법

모든 인증서는 유효기간을 갖는다. 또한 유효기간내의 개인키 분실, 자격상실, 키 변경 등의 이유로 인증서를 취소할 수 있다. 일반적으로 사용자가 인증서에 대한 유효성을 확인하기 위한 방법에는 인증기관이 발행하는 인증서 취소목록(CRL)에 자신이 사용하고자 하는 인증서가 포함되었는지를 확인하는 방법과 OCSP 서버를 이용하여 인증서의 상태를 검증할 수 있는 온라인 인증서 상태프로토콜 (OCSP) 방법이 있다.

(1) 인증서 취소목록 방법

인증서의 취소사유가 발생된 경우에는 인증서 사용자가 해당 CA에 인증서 취소요구를 하여야 한다. CA는 이를 디렉토리 서버(저장소)에 전송하고, 주기적으로 취소된 모든 인증서의 일련번호, 취소시간, 취소이유 등의 정보를 포함한 CRL을 작성하여 서명한다. 따라서 인증서 사용자들은 인증서의 상태정보를 얻기 위하여 CRL에 접속하여 인증서의 취소유무 및 상태정보를 확인한다. 이러한 방법은 CRL이 주기적으로 작성되기 때문에 실시간 상태정보를 제공할 수 없다

(2) 온라인 인증서 상태프로토콜 방법

OCSP 방식[8]은 실시간 인증서 상태를 조회하기 위한 클라이언트의 요구를 처리하기 위하여 OCSP 서버를 사용한다.

OCSP 서버는 클라이언트의 요청에 따라 인증서의 상태정보를 제공하는 서버로서 실시간 정보를 제공한다.

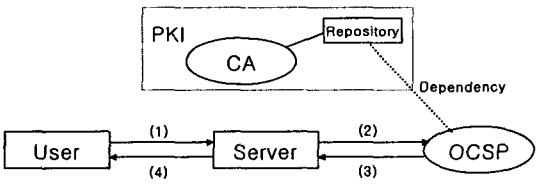

그림 2-3 OCSP 서버를 이용한 인증서 검증방법

(1) 사용자 접속 요구

(2) 사용자의 인증서 상태정보 요청

(3) OCSP의 인증서 상태정보 응답

(4) 사용자 접속 허가

OCSP 서버는 사용자의 OCSP 요구 메세지의 적절성 여부와 이에 대한 응답메시지 전송여부를 결정.

2. 5 AC와 PKC 결합방법

권한 인증을 위하여 AC를 사용할 때, 다음과 같은 이유로 PKC와 AC간의 결합을 필요로 한다.

첫 번째는 표준화된 AC 형식에는 공개키 정보를 포함하고 있지 않다. 사용자가 권한 접근제어를 원하는 시스템에 접근하기 위해서는 PKC를 이용한 사용자에 대한 인증이 우선되어야 한다. 또한 사용자가 ACA에서 AC를 발급받기 위해서는 PKC를 필요로 한다. 또한 접근제어를 요구하는 대다수의 시스템에서 사용자의 PKC를 이용한 사용자 인증을 필요로 한다.

두 번째는 각각의 인증서(AC, PKC)는 전혀 다른 유효기간을 갖는다. 유효기간이 긴 PKC에 유효기간이 짧은 권한 및 속성정보를 포함하는 방법은 CRL관리에 많은 문제점을 발생시킬 수 있다.

세 번째는 AC를 이용하여 시스템에 접근하여 서비스를 요구할 때, 사용자의 서비스요구 패킷은 사용자의 서명이 사용되므로 서버에서는 사용자의 PKC를 필요로 한다.

다음은 AC에 PKC를 결합하기 위한 방법이다.[3].

가. Monolithic Signature 방식 :

CA와 ACA가 역할이 동일한 인증기관인 경우에 적용될 수 있는 방법이다. Monolithic Signature 방식은 PKC의 확장자 영역에 AC를 포함하여 1개의 인증서로 구성하고, 인증기관 서명문을 첨가하는 형태이다.

나. Automatic Signature 방식 :

CA와 ACA가 서로 다른 별도의 인증기관으로 구성되어 있는 경우에 AC에 PKC를 결합하기 위하여 PKC 정보(일련번호, 발급자 ID 등의 일부정보)를 포함시키는 방법이다.

다. Chained Signature 방식

Automatic Signature 방식과 마찬가지로 AC와 ACA이 별도의 인증기관으로 구성되어 있는 경우에, AC에 PKC에 대한 서명문을 포함시키는 방법이다.

이러한 방법들은 모두 기존의 PKC에 역할, 책임과 관련된 속성정보를 포함할 수 없다는 단점에 기인하여, PKC와 AC를 결합시킬 수 있는 방법으로 제시되고 있다. 이와는 별도로 PKC의 확장자 영역에 다중 AC 기능을 할 수 있도록 여러 개의 속성인증기관에서 발행한 AC를 다중

으로 첨부할 수 있는 스마트인증서(Smart Certificate) 라는 방법도 제시되고 있다[2]. 그,러나 이러한 방법도 AC와 PKC의 유효기간의 차이로 인하여 현실적으로 사용할 수 없다.

2. 6 AC와 PKC 검증방법의 문제점

AC는 사용자에 대한 공개키 정보를 가지고 있지 않기 때문에 암호 및 인증서비스를 사용하기 위해서는 PKC와 병행하여 사용하여야 한다. 또한 실제로 접근제어를 원하는 서버에서는 인증서 검증절차를 두 번해야 된다는 단점이 있다. AC에 대한 CRL 검증과 PKC에 대한 CRL 검증이 수행되어야 한다.

이와같은 이유로 최근의 많은 연구에서 AC와 PKC를 결합시키는 방안이 제시되고 있다. 그러나 이와같이 AC와 PKC를 결합하는 방법은 인증기관(CA, ACA)이 별도로 구성된 형태에서는 동시성(concurrency)를 만족시킬 수 없다는 문제점이 제기된다[1]. 즉 AC와 PKC의 유효여부가 동시에 만족되지 않을 수 있기 때문이다. 예를 들면 AC는 유효하지만, PKC가 취소된 상태이면 AC를 사용할 수 없다.

또한 AC 관련 표준화 문서[6]에는 PKC의 유효기간은 긴 반면에, AC의 유효기간은 짧다는 점 때문에 인증기관에서 CA와 ACA의 기능을 병행할 수 없도록 제안하고 있다. 또한 PKC는 항상 CRL을 유지해야 하는 반면, AC의 경우에는 가능하면 CRL을 사용하지 않도록 권고하고 있다. 그러나 AC에 대한 취소 사유 발생시에 이에 대한 처리가 필요하다.

두 개의 인증서의 동시 유효성을 만족시키기 위하여 지금까지 제안된 대부분의 논문은 인증서 형식에 중점을 두고 두개의 인증서를 결합하는 방안을 연구하고 있으나, 본 논문에서는 AC와 PKC에 대한 인증서 상태정보를 보관하는 OCSP 서버를 통합 관리 방법을 제안한다.

OCSP 응답으로 제공되는 상태정보는 Good, Unknown, Revoked로 분류된다[8]. Good 상태는 인증서가 유효함을 의미하고, Revoked 상태는 인증서가 취소되어 무효화됨을 의미하며, Unknown 상태는 요구된 인증서에 대한 상태정보를 가지고 있지 않아, 확인할 수 없음을 의미한다.

PKC는 유효하지만, AC는 취소된 상태를 Partial Good 상태로 정의하고, 특정서버의 서비스를 사용할 권리가 없음을 의미하고, AC가 유효하더라도, PKC가 무효화되면 AC를 사용할 수 없다.

표 2-2 PKC와 AC의 상태정보

종류	PKC	AC	사용가능 여부	상태정보
1	유효	유효	PKC와 AC 모두 사용가능	Good
2	유효	무효	PKC 사용가능, AC 사용불가	Partial Good
3	무효	유효	PKC 사용불가, ACC도 사용불가	Revoked
4	무효	무효	PKC AC 모두 사용불가	Revoked
5	확인 불가		PKC, AC 모두 사용 불가	Unknown

3. 제안 방식

(1) OCSP를 적용한 Pull 방식의 접근제어 모델

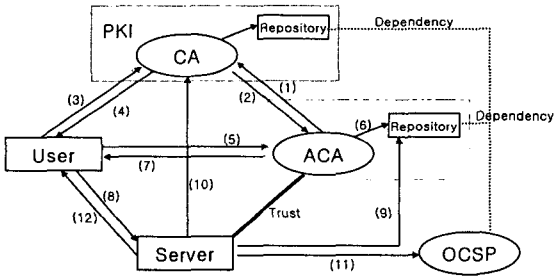

그림 3-1 OCSP를 이용한 Pull 방식

① ACA은 CA에게 PKC를 요청한다.

② CA에서는 ACA의 신원확인, 키발급 절차를 거쳐 PKC를 발급한다.

③ 사용자는 CA로부터 신원확인을 통해 PKC를 요청한다.

④ CA에서는 사용자의 신원확인, 키발급 절차를 거쳐, 사용자에게 PKC를 발급 하여준다.

⑤ 사용자는 ACA에게 AC를 요청한다. 이때 사용자는 자신의 PKC를 첨부하여 전송한다.

⑥ ACA는 사용자의 접근권한 확인과정을 거친 후, 발급된 AC는 ACA의 AC 저장소에 보관한다. 이때 ACA는 사용자의 PKC와 AC에 대한

정보를 OCSP에 전달한다.

⑦ 사용자에게는 AC가 발급되었음을 알려준다.

⑧ 사용자가 서버에 접속하는 경우, 서비스 요구 패킷을 보낸다.

⑨ 서버에서는 사용자의 서비스 요구를 수신하고, ACA에 접속하여 사용자에 대한 AC를 획득한다.

⑩ 서버에서는 ACA의 서명문을 확인하기 위해 CA에서 ACA의 공개키를 확인한다.

⑪ 사용자의 AC와 PKC의 상태여부를 OCSP를 통하여 실시간으로 확인한다. 인증서의 상태를 살펴볼 때, PKC의 상태가 유효하며 AC의 상태가 유효하지 않은 경우(Partial Good), 서버측은 사용자에게 AC를 재발급 받을 것을 권고하고 자원에 대한 접근을 허락하지 않는다. 사용자의 AC 상태가 유효하고 PKC의 상태가 유효하지 않은 경우, 서버측은 사용자에게 PKC를 재발급 받을 것을 권고하고 접속을 해지한다. (또한 ACA에게 해당 사용자의 AC 취소를 요구한다)

⑫ ACA는 사용자에게 접속허용여부를 알려준다.

(2) OCSP를 적용한 Push 방식의 접근제어 모델

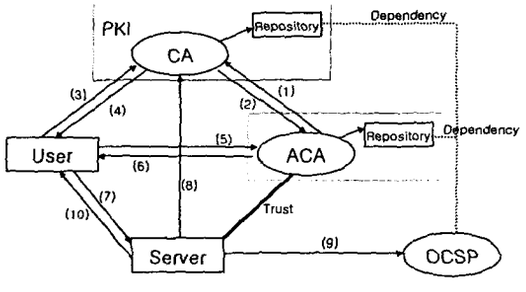

그림 3-2 OCSP를 이용한 Push 방식

① ACA는 CA에게 PKC를 요청한다.

② CA에서는 ACA의 신원확인, 키발급절차를 거쳐 ACA에게 PKC를 발급한다.

③ 사용자는 CA에게 PKC를 요청한다.

④ CA에서는 사용자의 신원확인, 키발급절차를 거쳐, 사용자에게 PKC를 발급한다.

⑤ 사용자는 ACA에게 AC를 요청한다. 이때 사용자는 자신의 PKC를 첨부하여 전송한다.

⑥ ACA는 사용자의 접근권한 확인과정을 거친 후, 발급된 AC를 사용자에게 전달된다. 이때 ACA는 사용자의 PKC와 AC에 대한 정보를 OCSP에 전달한다.

⑦ 사용자가 서버에 접속하는 경우, 사용자의 AC를 첨부하여 서비스 요구 패킷을 보낸다.

⑧ 서버에서는 ACA의 서명문을 확인하기 위해 CA에서 ACA의 공개키를 확인한다.

⑨ OCSP를 적용한 Pull 방식의 ⑪ 과정과 동일하다.

⑩ 사용자에게 접속허용여부를 알려준다.

4. 제안된 방식의 특징

OCSP는 CA의 CRL 기능을 위임받아 사용자에게 인증서의 상태정보를 실시간으로 전달해 주는 시스템이다. 본 논문에서는 기존의 PKC에 대한 CRL 기능을 제공하는 OCSP의 기능을 PKC에 대한 CRL 뿐 만아니라, AC에 대한 CRL 처리기능을 포함시킴으로서, PKC와 AC 간의 동시성 문제를 해결하였다. 이를 위하여 기존의 PKI 구조에서의 OCSP가 CA로 부터PKC 상태정보를 수신하는 것과 마찬가지 방법으로, OCSP는 ACA로부터 AC 상태정보를 수신하는 방법을 사용한다.

따라서 ACA는 OCSP로부터 사용자의 PKC가 유효하다는 응답과 접근권한이 확인되면, 사용자에게 AC를 발급하고, 이를 OCSP에 등록한다. 만일 PKC로부터 취소상태의 응답을 받으면, 사용자에게 AC를 발급하지 않는다.

또한 PKI 구조의 CA처럼 ACA는 인증서 취소사유가 발생되면, 이를 OCSP 서버에 전달함으로서, OCSP 서버는 실시간으로 사용자 인증서(PKC 및 AC)에 대한 상태정보를 제공한다. 그림 4-1과 같이 OCSP는 CA 저장소로부터 상태정보와 ACA 저장소로부터 받은 상태정보를 취합하여 사용자의 인증서 상태정보요구에 대하여 통합적인 상태정보 응답을 제공하는 방법이다.

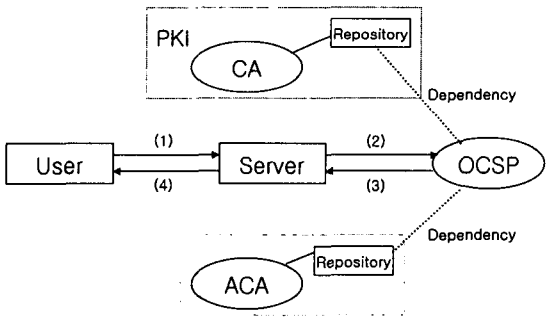

그림 4-1 OCSP에서의 인증서 상태정보 처리방법

본 논문에서는 OCSP 서버에서 PKC와 AC에 대한 상태정보를 각각 CA와 ACA로부터 실시간으로 전달받아 표 4-1과 같이 사용자별 인증서 상태정보에 대한 데이터베이스를 구축함으로서, 사용자의 인증서 상태정보 요구에 대하여 PKC와 AC의 상태정보를 통합 처리할 수 있다. 일반적으로 PKC 1개는 여러 서버에 접속할 수 있는 AC들과 결합시킬 수 있다. 따라서 표 4-1은 PKC의 일련번호를 포함한 Cert ID를 이용하여, 관련된 AC들의 상태정보를 제공하는 구조이며, 표 4-2는 AC의 일련번호를 포함한 AC ID를 이용하여 관련된 PKC의 상태정보를 제공하는 구조이다.

표 4-1 OCSP에서의 PKC와 AC 상태표 (PKC ID 기준)

PKC ID	사용자 DN	PKC			AC				적합성
		CA DN	유효 기간	취소 사유	AC ID	ACA DN	유효 기간	취소 사유	

표 4-2 OCSP에서의 PKC와 AC 상태표 (AC ID 기준)

AC ID	사용자 DN	AC			PKC				적합성
		ACA DN	유효 기간	취소 사유	PKC ID	CA DN	유효 기간	취소 사유	

OCSP를 이용한 단일 절차에 의한 AC와 PKC 검증방법은 OCSP 서버내에서 인증서의 일련번호를 포함한 인증서 ID 별로 PKC와 AC에 대한 상태정보를 결합함으로서, 기존의 인증서 검증을 위한 CRL을 조회하는 과정을 1번으로 단일 절차로 제안하였다.

5. 결론

PKC를 이용한 인증기반구조와는 별도로, 응용서버에 대한 접근권한을 허용하기 위한 AC에 대한 내용을 검토하였다. AC는 사용자의 공개키 정보를 포함하지 않고 있기 때문에, 접근통제를 위한 응용시스템에 사용하기 위해서는 공개키 정보를 포함하고 있는 PKC와 함께 사용되어야 한다. 그러므로 두 개의 인증서를 사용하기 위해서는 PKC에 대한 검증과정과 AC에 대한 검증과정이 병행되어야 한다.

본 논문에서는 기존 OCSP 서버의 기능을 보완하여 PKC에 대한 상태정보와 함께, 권한기반구조에서 사용되고 있는 AC에 대한 상태정보를 제공함으로서, PKC와 AC간의 동기문제를 해결하였다. 이를 위하여 OCSP 서버에서 AC와 PKC를 결합하여 보관하는 데이터구조를 제안하였다. 또한 이와같은 과정을 통하여 인증서 검증과정을 단일화하여 속도를 향상시키고, 비용을 절감시키는 효과를 발생시킬 수 있다.

마지막으로 본 연구 결과는 보다 섬세한 프로토콜 제안을 통하여 보완되어야 할 것으로 생각되며, 기존의 사용자 인증과 권한 인증을 동시에 필요로 하는 응용시스템에 적용될 수 있을 것으로 보인다.

참고문헌

[1] Himanshu Khurana and Virgil D. Gligor, Enforcing Dependencies between PKI Certificates in ad-hoc networks", IEEE International Conference on Telecommunications, Buchartest, Romania, pp. June 2001, 293- 298.

[2] Joon S. Park and Sandhu, "Smart Certificates : Extending X.509 for Secure Attribute Service on the Web", NISSC / 1999

[3] Joon S. Park and Sandhu, "Binding Identities and Attributes Using Digitally Signed Cerificates", ACSAC / 2000

[4] "X.509_4th Edition Draft V8 - Draft ISO/IEC 594-8", May 3. 2001.

[5] Internet Draft "An Internet Attribute

Certificate Profile for Authorization", S.Farrel, Jun 2001.
[6] RFC 3281 "An Internet Attribute Certificate Profile", S. Farrell, April 2002.
[7] RFC 2459, "Internet X.509 Public Key Infrastructure Certificate and CRL Profile", IETF PKIX Working Group, January, 1999.
[8] RFC 2560, "X.509 Internet Public Key Infrastructure Online Certificate Status Protocols - OCSP", IETF PKIX Working Group, 2001.