컴포넌트 개발방법을 이용한 보안 강화 엔진 구현

김은아°, 김미희, 채기준
컴퓨터학과, 이화여자대학교
{dmsk999°, mihui, kjchae}@ewha.ac.kr

Implementation of Security Enforcement Engine Using COM Programming

Eunah Kim°, Mihui Kim, Kijoon Chae
Dept. of Computer Science and Engineering, Ewha Womans Univ.

요 약

네트워크 보안의 요구 사항을 만족시키기 위한 필수적인 보안 서비스에는 인증과 권한부여가 있다. 이러한 서비스를 제공하는 보안 강화 엔진을 구현함에 있어 컴포넌트 개발방법을 이용하면 구축 비용 및 시간을 절감할 수 있고 바이너리 단위로의 재사용이 가능하며 유지 보수의 용이성 및 확장성이 뛰어나다는 장점을 갖는다. 이에 본 논문에서는 컴포넌트 모델 중 마이크로소프트사의 COM(Component Object Model)을 기반으로 보안 강화 엔진을 구현하였다.

1. 서 론

인터넷의 확산과 통신 기술의 급속한 발전은 고도의 정보 통신망 구축을 가능하게 하였을 뿐 아니라 각종 정보를 공유하도록 함으로써 사회 전반에 큰 변화를 가져왔다. 그러나 이와 더불어 인터넷 등의 컴퓨터 통신망을 통하여 정보가 위조 또는 변조되고 허락 없이 유출되는 등의 불법적인 행위가 발생함으로써 정보화로 야기되는 역기능 또한 심각하다. 그러므로 이와 같은 위험을 사전에 방지하고 네트워크 보안에 대한 요구 사항을 만족시키기 위한 방법이 필요하다. 보안 서비스는 1) 인증, 2) 권한부여, 3) 데이터 기밀성, 4) 데이터 무결성, 5) 부인봉쇄, 6) 가용성 등으로 구분될 수 있다. 이들 중 인증과 권한부여는 보안 서비스에서 필수적인 요소로서 통신상의 노드나 데이터, 사용자 등을 신뢰할 수 있는지를 판단하고 이를 바탕으로 시스템 및 응용에 대한 접근을 제한하고 제어하도록 한다. 이러한 보안 서비스는 프로토콜 계층에서 제공되거나 보안 정책 혹은 보안 메커니즘의 구현을 통하여 이루어 질 수 있다[1].

인터넷의 빠른 보급으로 인한 또 다른 변화는 분산 시스템의 대중화이다. 분산 처리가 가능한 컴퓨터 환경에서는 전통적인 집중식 소프트웨어 개발 방법에서 벗어나 컴포넌트 기술을 지원하는 개발로의 변화가 필요하다. 하지만 새로운 기술과 환경에서의 개발은 기존의 시스템과의 접목 또한 병행되어야 한다. 이런 요구에 대응할 수 있는 최적의 해결책으로 대두된 것이 컴포넌트 기반 소프트웨어 개발방법이고 이를 이용한 다양한 측면의 개발이 시도되고 있다[2]. 본 연구는 컴포넌트 기반의 개발방법을 이용하여 분산된 이질적인 환경에서 보안 서비스를 제공하기 위한 보안 강화 엔진 컴포넌트 구현에 목적을 둔다. 다음은 보안 강화 엔진을 컴포넌트로 구현함으로써 얻을 수 있는 이점들이다[3-6].

- 분산 환경에의 적합성: 증대된 대역폭과 네트워크의 중요성으로 인해 네트워크를 통해 각각의 부분이 분산되어 구성된 어플리케이션에 대한 필요성이 대두되었고, 컴포넌트는 이러한 어플리케이션을 쉽게 구성할 수 있다.
- 재 사용성: 보안 엔진 컴포넌트는 웹을 통한 서비스나 DCOM(Distributed Component Object Model) 혹은 소켓을 통한 바이너리 차원에서 코드를 재사용할 수 있게 한다.
- 플랫폼 및 언어 독립성: 컴포넌트는 소스코드의 캡슐화와 인터페이스를 통한 접근을 허용하여 이질의 플랫폼에서 동작될 수 있다. 따라서 네트워크 상의 서로 다른 플랫폼을 가진 노드에서 보안 모듈을 필요로 할 때 유용하게 사용될 수 있다. 또한 서로 다른 언어로 제작된 클라이언트 모듈로부터의 접근도 용이하다.
- 유지 보수 및 확장의 용이성: 컴포넌트의 소스코드와 인터페이스를 분리하고 컴포넌트와 클라이언트의 구현에 관한 사항을 인터페이스에 반영하지 않기 때문에 인터페이스의 변형 없이 컴포넌트를 업그레이드 할 수 있다.

본 논문의 구성은 다음과 같다. 2장에서는 보안 강화 엔진의 기본 엔진 구조를 살펴보고 동작 원리를 기술한다. 3장에서는 보안 강화 엔진 구조를 바탕으로 구현된 보안 강화 엔진 컴포넌트를 설명한다. 4장에서는 결론 및 향후 연구 방향을 제시한다.

2. 보안 강화 엔진 구조

네트워크 상에서 보안 기능을 제공하기 위해서 기본적으로 제공되어야 할 서비스에는 인증과 권한부여가 있다. 인증은 네트워크 노드의 신분을 증명하는 것으로써 다른 보안 서비스가 올바르게 실행될 수 있도록 하는 기반이 되는 보안 서비스이다. 따라서, 권한부여 서비스가 이루어지기 위해서는 인증서비스가 우선이 되는 구조를 가진다. 전체적인 보안 서비스의 흐름을 살펴보면, 먼저 인증 받고자 하는 노드는 네트워크 상의 웹 서버, 데이터베이스 서버 등의 자원을 사용하고자 하는 사용자측이 될 수 있다. 이러한 노드들이 암호학적으로 안전하게 처리된 메시지를 통신로 상에 보내게 된다. 자원을 할당할 서버 혹은 노드의 보안 모듈에서는 암호화 된 메시지를 복호화 하고 무결성 검사를 한다. 이후 인증 모듈에서 사용자 혹은 노드가 적합한지 인증을 한다. 인증이 되면 권한부여 모듈에서 자원 접근에 대한 정당성을 검증하고 권한을 부여한다[7]. 그림 1은 네트워크에서 안전한 통신을 위한 요구사항을 바탕으로 설계한 보안 강화 엔진의 구조로 본 논문에서는 인증과 권한부여 모듈을 구현하였다.

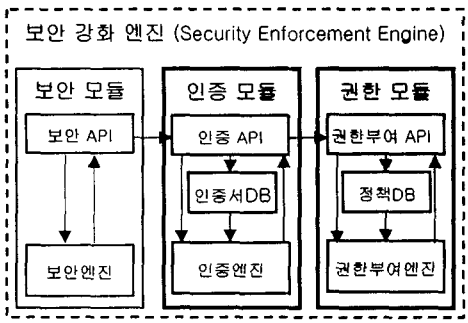

그림 1. 보안 강화 엔진 구조[7]

2.1 인증 모듈

이 모듈에서는 인증 받고자 하는 노드의 신원을 인증하는 기능을 수행한다. 인증 모듈이 실행되기 위해 필요한 데이터는 인증기관(Certificate Authority)으로부터 받은 해당 사용자에 대한 인증서와 인증 받고자 하는 사용자 측에서 보내온 인증서이다. 본 논문에서의 인증서는 X.509 형식을 이용하였다. 인증을 위하여 인증을 수행하는 노드는 인증 받고자 하는 노드의 메시지로부터 인증서 필드를 추출하고, 자신의 인증서 데이터베이스에 저장중인 인증기관으로부터 받은 인증서와 비교하여 유효성을 검사한다. 또한 인증서의 유효 기간을 검사하여 만기된 다른 인증서를 사용하지 않았는지에 대한 검사도 한다. 유효하지 않다고 판단되면 인증 실패 결과를 출력하고 처리를 종료한다.

2.2 권한 부여 모듈

이 모듈은 인증 모듈에 의하여 인증이 완료된 후에 사용자가 실제로 사용할 수 있는 네트워크 자원의 권한을 검사하여 부여하는 기능을 담당한다. 자원을 할당할 서버와 같은 노드들은 권한부여에 대한 정책들을 저장한 데이터베이스(정책DB)를 보유한다. 정책DB에는 각 사용자 혹은 호스트 별 자원 접근 권한에 대한 정책데이터들이 저장되어 있고, 이들은 서비스 별로 분류될 수 있다. 사용자는 자신이 사용하고자 하는 서비스에 대한 권한부여를 요청하고 기존의 정책에 따라 적합한 사용자라고 판단되면 권한을 부여받는다. 정책에는 모드, 서비스, 호스트, 사용자, 권한, 시간 정보 등을 포함한다. 모드는 1) 자원할당에 대한 요구, 2) 새로운 정책 추가에 대한 요청, 3) 기존 정책 삭제에 대한 요청으로 구분하였다. 서비스는 사용자가 사용하고자 하는 어플리케이션이나 서비스를 명시하기 위하여 사용되고, 호스트와 사용자는 해당 서비스에 접근하고자 하는 호스트와 사용자를 명시한다. 시간정보는 서비스를 사용할 수 있는 제한 시간을 명시한다.

3. 보안 강화 엔진의 컴포넌트 구현

3.1 컴포넌트 구현 기술 동향

컴포넌트는 소프트웨어 시스템에서 독립적인 업무 또는 독립적인 기능을 수행하는 모듈로서 시스템을 유지보수 하는데 있어서 교체가 가능한 부품과도 같다. 컴포넌트는 실행코드(Executable)를 기반으로 재사용이 가능하도록 구현되고 개발에 필요한 모든 관련 정보들이 패키지화 되어 있어서 독립적으로 배포 가능하다는 장점을 가진다.

컴포넌트 관련 기술의 표준화는 EJB(Enterprise JavaBeans), COM 또는 CCM(CORBA Component Model) 등으로 이루어지고 있다. 마이크로소프트의 COM+는 COM과 MTS(Microsoft Transaction Server)를 통합시킨 컴포넌트 모델로 윈도우2000과 접목하여 분산시스템에서 많은 연구 사례를 보이고 있다. 본 논문의 구현 사례도 COM+로 확장시킬 수 있다.

본 논문에서는 이러한 컴포넌트 모델 중 마이크로소프트의 COM을 기반으로 보안 강화 엔진 컴포넌트를 구현하였고, 윈도우 환경에서 보안 강화 엔진의 동작을 확인할 수 있도록 테스트용 COM 클라이언트 프로그램을 구현하여 실험하였다.

3.2 보안 강화 엔진의 컴포넌트화

기존의 보안 서비스 모듈들은 네트워크 상의 노드 혹은 라우터에서 운용되도록 하기 위하여 많은 부분 UNIX나 LINUX환경에서 구현되고 있다. 그러나 마이크로소프트 환경 하의 윈도우 시스템을 기반으로 한 서버 혹은 네트워크 노드들 역시 보안 서비스가 중요하므로 본 논문에서는 이러한 노드를 대상으로 하였다. 이들 프로그램에 대한 소스 코드 재사용과 개발비용을 최소화하는 동시에 컴포넌트화 하기 위하여 COM을 사용하였다.

COM은 동적으로 소프트웨어를 구성하는 것을 가능하게 하는 소프트웨어 아키텍처이다. COM을 통하여 클래스의 구현으로부터 인터페이스를 분리하여 바이너리 차원의 코드 재사용과 유지 보수의 용이성을 달성할 수 있다. COM 컴포넌트를 생성하기 위해서 순수 C++를 사용할 수 있지만 반복적인 코드(예를 들면, IUnknown과 클래스 오브젝트)를 추가하는데 많은 시간을 낭비하게 된다. 또한 MFC를 이용할 수도 있지만 경량의 COM 클래스를 작성하기란 쉽지가 않다. 이에 본 논문에서는 반복적인 코드를 작성하

지 않고 MFC 아키텍처의 모든 기능을 사용하지 않고도 COM 컴포넌트를 구현하기 위한 방법으로 마이크로소프트의 컴포넌트 저작 도구인 ATL(Active Template Library)을 사용하여 효율을 얻고자 하였다. 구현 플랫폼으로는 Windows XP, Microsoft Visual Studio.NET 환경을 사용하였다.

구현한 엔진 컴포넌트에 대한 인터페이스는 그림 2와 같이 6가지로 분류된다. 먼저 실행에 필요한 입력 파라미터를 설정하는 데 있어서 엔진이 실행되는 컴퓨터 내에 저장된 인증서와 정책데이터를 설정하는 SetLocalParameter 인터페이스와 인증 받고자 하는 사용자로부터 받은 인증서와 정책데이터를 설정하는 SetInParameter 인터페이스가 있다. 실제 인증과 권한부여를 위한 인터페이스로는 Authen와 Check_authorization가 사용된다. 권한부여는 정책데이터의 모드가 1일 때 수행되며, 모드 2와 모드 3일 경우에는 Add_policy와 Del_policy 인터페이스로 엔진에 접근할 수 있다.

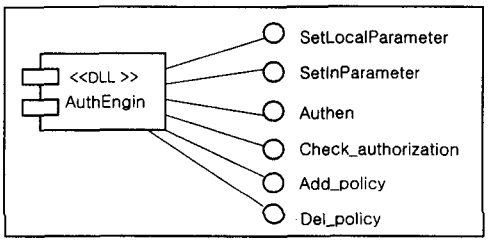

그림 2. 보안 강화 엔진 컴포넌트 인터페이스

3.3 컴포넌트 실행 결과

보안 강화 엔진 컴포넌트는 어플리케이션 구축 시 바이너리 단위의 재사용을 가능하게 한다. 따라서 윈도우 기반의 데스크 탑이나 서버 등의 어플리케이션에 컴포넌트를 올리는 경우 사용자의 프로그램을 통하여 혹은 웹 환경에서는 브라우저를 통하여 컴포넌트의 실행 결과를 보여줄 수 있게 된다.

본 논문에서는 윈도우 폼 형태의 테스트용 COM 클라이언트를 구현하여 보안 강화 엔진 컴포넌트의 실행 결과를 쉽게 알아볼 수 있도록 하였다. COM 클라이언트는 입력 파라미터를 설정하는 부분과 실행 결과를 메시지 형태로 보여주는 부분으로 구성된다. 인증과 권한부여는 각각의 인터페이스로 분리하였기 때문에 독립적으로 수행된다. 그림 3은 성공적으로 인증과 권한 부여를 수행한 결과를 보여준다.

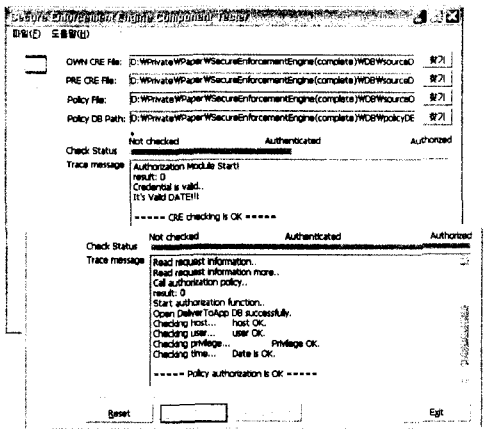

그림 3. 성공적인 인증 및 권한부여 수행 결과

4. 결 론

본 논문에서는 인증과 권한부여 서비스를 제공하는 보안 강화 엔진의 컴포넌트화 작업을 수행하였다. 이를 위하여 보안 서비스에서 필요한 사항과 인증 및 권한부여 메커니즘을 정리하고 컴포넌트화를 통하여 보안 강화 엔진이 네트워크 상에서 다양하게 적용될 수 있는 장점들을 고려하였다. 그러한 장점들로 인해 구현된 보안 강화 엔진 컴포넌트는 네트워크 상의 보안상 취약성을 갖는 노드들에 쉽게 적용될 수 있다. 이는 서로 다른 언어로 구현된 어플리케이션을 갖거나 서로 다른 플랫폼에서도 엔진을 바이너리 코드 차원에서 재활용할 수 있음을 의미한다.

본 논문을 통해 구현된 보안 강화 엔진 컴포넌트에는 앞서 소개한 보안 강화 엔진의 보안 모듈은 포함되어있지 않다. 보안 모듈은 메시지의 무결성을 검증하기 위해 암호화, 복호화와 전자 서명을 수행하는 부분으로 이들 기능 또한 인증, 권한부여와 같이 보안 서비스에서 필요한 부분이므로 향후 이를 위한 알고리즘들을 포함하는 독립적인 컴포넌트를 구현하고자 한다.

참 고 문 헌

[1] William Stallings, "Network Security Essentials," Prentice Hall, 2003.

[2] 최성운, "전자사전 컴포넌트 구현" , 정보처리학회지, 제8-D권, 제5호, pp.587-592, 2001년 10월.

[3] "Component Object Model," http://msdn.microsoft.com/library/default.asp?url=/library/en-us.

[4] "컴포넌트란 무엇인가" ,http://www.sds.samsung.co.kr/support/epartner/ecpaper/.

[5] George Shepherd, "Visual C++.NET" , 정보문화사, 2003.

[6] Julian Templeman, JohnMueller, "COM programming with Microsoft.NET," Microsoft press, 2003.

[7] 김옥경, 임지영, 나현정, 나가진, 김여진, 채기준, 김동영, "액티브 네트워크 상에서 액티브 노드의 보안 강화를 위한 보안 엔진 구현" , 한국정보처리학회논문지, 제10-C권, 제4호, 2003년 8월.