XML 기반 글로벌 Peer-to-Peer 엔진의 보안 관리자 설계

이일수[0], 이재욱, 권태숙, 이승룡
경희대학교 컴퓨터공학과
{primelee, jaeki, sooki, sylee}@oslab.khu.ac.kr

Design of a Security Manager in Global Peer-to-Peer Engine Based on XML

Ilsu Lee[0], Jaewook Lee, Taisook Kwon, Sungyoung Lee
Dept. of Computer Engineering, KyungHee University

요 약

본 논문에서는 PC, 웹, 모바일 환경을 지원하는 글로벌 Peer-to-Peer 엔진에서 인증, 기밀성 및 무결성을 제공하기 위해 암호화/복호화에 필요한 공개키 쌍의 생성/분배/관리 기능을 지원하는 보안 관리자의 설계에 대하여 기술한다. 기존의 Peer-to-Peer 시스템이 단지 인증기관이 발행한 인증서에 의한 공개키의 분배를 지원하는데 비해 제안된 보안 관리자는 인증서뿐만 아니라 인증서가 아닌 다른 전송매체에 의해 분배된 공개키 또한 저장/관리할 수 있는 기능을 제공한다. 이것은 기존의 시스템이 인증서를 발급 받기 위해서 복잡하고 번거로운 신원 확인과 사용료를 부담하는 것에 비해 간편하게 키를 공유하여 보안을 적용시킬 수 있다. 이를 위해 본 논문에서는 키 링(Key Rings)이라는 개념을 도입하는데 이는 각 피어에 해당하는 다수의 공개키/개인키를 저장 및 관리할 수 있는 한 쌍의 자료구조로써 제공된다. 또한 본 논문은 이렇게 저장/관리되는 다수의 공개키를 선택하여 사용하기 위한 알고리즘을 제시한다. 제안된 키 링을 사용하는 방식은 인증서를 사용하는 클라이언트/서버 형태의 Peer-to-Peer 보다 순수한 Peer-to-Peer 모델에 근접한 형태라고 볼 수 있다.

1. 서 론

Peer-to-Peer(이하 P2P)는 동등한 능력과 책임을 가지는 각각의 워크스테이션의 네트워크 형태를 말한다. 즉 P2P는 기존의 인터넷의 지배구조였던 '클라이언트/서버' 구조 중심의 모델로부터 '클라이언트/클라이언트' 구조를 나타내는 'P2P'구조로의 변화를 뜻한다. 이러한 P2P는 익명성이 전제되고, 모든 통신은 개방된 통신망을 통하여 전송되므로 정보가 공격자에게 노출되는 위협이 발생한다. 본 연구는 이러한 P2P 서비스에서의 문제점을 해결하고자 한다.

기존의 P2P 시스템들이 이러한 문제점을 해결하기 위하여 각각의 피어가 공인 인증기관에서 인증서를 발급 받아 사용자를 인증하고 세션키를 공유하여 암호화 통신을 하는 방식을 사용하고 있다[1].

본 논문에서는 PC, 웹, 모바일 환경에서 연동 가능한 글로벌 P2P 엔진에서 사용되는 키의 생성/분배/관리 기능을 지원하는 보안 관리자를 설계 및 구현하는 방법을 제시한다. 이를 위해 키 링이라는 개념을 도입하였는데, 이러한 키 링은 피어들이 각각 생성하여 분배한 다수의 공개키/개인키 쌍을 저장/관리하기 위한 개념으로 공개키 링은 공개키의 신뢰도를 파라미터로 저장하고 있어 나중에 이 신뢰도에 의한 키의 선택이 가능하도록 하며, 개인키 링은 개인키가 외부로 노출되지 않도록 특정한 암호 알고리즘으로 암호화되어 저장된다. 또한 본 논문에서는 이렇게 저장/관리되는 다수의 공개키를 선택하여 사용하기 위한 알고리즘을 제시한다.

본 논문의 구성은 다음과 같다. 2장에서는 P2P와 관련된 연구를 소개하고, 3장에서는 글로벌 P2P 엔진에 관하여 간략하게 소개하며, 4장에서는 보안 관리자의 설계 및 정책에 관하여 설명을 하겠고, 5장에서는 결론을 맺으며 향후 과제에 대해 언급한다.

2. 관련 연구

본 장에서는 기존의 P2P 시스템에 대한 관련 연구와 P2P 시스템에서의 보안에 대한 기존의 연구에 대해 살펴보겠다.

- **누텔라(Gnutella)**

누텔라는 NullSoft가 만들어 공개되었으며 현재 대부분의 P2P 솔루션의 개발에 활용되고 있다. 누텔라는 하이브리드(Hybrid) 형태의 냅스터(Napster)와는 달리 서버가 필요하지 않는 순수한 형태의 P2P 방식으로 플랫폼에 독립적인 파일공유 솔루션이다[2].

누텔라는 Ping-Pong, 검색(Query), 다운로드의 세 가지 기능을 가지며, 누텔라의 메시지는 TCP/IP 프로토콜을 사용하며 다운로드에 관해서는 HTTP를 사용한다. 누텔라에서 호스트가 생성하고 관리하는 정보는 피어의 IP 리스트뿐이며, 텍스트 파일로 저장된다. 누텔라는 자체적으로 보안이 적용되는 부분은 없으며, 누텔라를 활용하여 보안을 적용하는 솔루션들이 현재 개발되어지고 있다.

- **MagiTM**

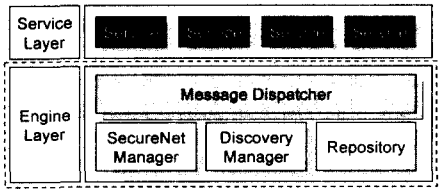
MagiTM은 현재 나온 P2P 솔루션에 관한 개념 중 가장 통합적이라고 할 수 있는 endtech의 솔루션이다. MagiTM은 기존의 웹을 적극 활용하고, 소프트웨어를 컴퍼넌트 형식과 이벤트 형식으로 묶고, 플랫폼의 개념으로 접근하여 피어 자체의 독립성을 최대한 확보하는 4가지 개념으로 접근하고 있는 시스템이다[3]. MagiTM에서는 SSL(Secure Sockets Layer)를 사용하여 암호화를 하는데, 암호화를 위한 키로써 SSL 인증서로 교환된 공개키를 사용하며, 웹과 인터넷에서 일반적으로 네 가지(Basic, MD5, Tokens, SSL Certificates)의 인증 정책을 지원한다[3].

그러나, 이러한 P2P 시스템들은 모두 인증서를 사용하여 인증 및 키 교환을 함으로써 보안을 적용하는데 있어 Hybrid 방식 즉 클라이언트/서버 형식에 가깝다고 볼 수 있다. 그러나 본 논문에서 제안될 방식은 각각의 피어가 생성한 공개키/개인키 쌍을 사용할 수 있기 때문에 보다 순수한(pure) P2P 방식이라고 할 수 있다.

3. Global Peer-to-Peer 엔진

본 장에서는 본 논문에서 제시하는 보안 관리자를 적용하는 글로벌 P2P 엔진에 관하여 설명하고자 한다.

글로벌 P2P 엔진의 구조는 [그림 1]과 같고, 크게 상위의 서비스 계층과 하위의 엔진 계층으로 나눌 수 있다.

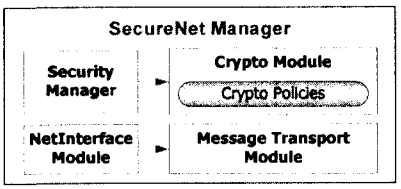


[그림 1] 글로벌 Peer-to-Peer 엔진의 구조도

상위계층의 서비스 레이어는 쪽지, 채팅, 스트리밍서비스와 같은 다양한 서비스들로 이루어져 있으며 각 서비스들이 컴포넌트화 되어있기 때문에 사용자의 취향에 따라 서비스의 종류를 선택할 수 있도록 구성되어 있다.

하위계층의 엔진 레이어는 메시지 디스패쳐, 레파지토리, 디스커버리 관리자, 보안네트워크 관리자로 구성되어 있다. 이중 보안 네트워크 관리자는 암호화 기능을 가지고 있는 모듈로써 사용자의 요구 또는 서비스의 종류에 따라서 다른 피어에게 전송되어진 메시지들을 암호화하거나 송신된 메시지를 복호화 하는 작업을 수행한다.

또한 글로벌 P2P 엔진에서 전달되는 모든 메시지들은 XML 포맷으로 구성되어 있으며, XML 파서를 이용하여 분석 및 처리를 수행하게 된다.



[그림 2] SecureNet Manager의 구조도

보안 관리자는 P2P 엔진의 일부분인 SecureNet Manager ([그림 2])에 포함되어 있으며, SecureNet Manager는 사용되는 키의 생성/분배/관리를 담당하는 보안 관리자와 전송시 암호화/복호화 과정을 수행하는 Crypto 모듈, 데이터 전송을 지원하기 위한 네트워크 관련 기초모듈인 NetInterface 모듈과 실제로 데이터를 전송하는 역할을 하는 Message Transport 모듈로 구성되어 있다.

SecureNet Manager에서 암호화하는 방식은 공개키 암호화 방식과 대칭키 암호화 방식을 모두 사용한다. 보안 관리자는 먼저 공개키 암호화 방식을 사용하기 위하여 수신자의 공개키를 얻어오고 또한 대칭키 암호화 방식을 사용하기 위하여 세션키(대칭키)를 생성하면, Crypto Module에서 데이터를 먼저 자신의 개인키를 이용해서 전자서명한 후, 이 전자서명과 데이터를 세션키로 암호화하며, 이때 사용된 세션키를 수신자의 공개키로 암호화하여 전송하게 된다. 이 방식을 사용하게 되면 기밀성과 인증, 그리고 무결성도 보장되게 된다[4].

이때 사용되는 두 종류의 키(공개키, 대칭키)는 보안 관리자에서 생성/분배/관리되어지는데 본 논문에서는 이러한 보안 관리자에서의 공개키의 생성 및 저장/관리에 대한 연구를 소개한다.

4. 보안 관리자(Security Manager)

Pure한 P2P 환경은 상호인증을 하는 것이 거의 불가능하므로 인증과 암호화를 하기 위한 키 교환을 하기 위해서는 인증기관(CA)과 같은 신뢰할 수 있는 제 3자가 필요하다. 그러므로 보안 서비스가 요구되는 경우에는 인증기관에서 공개키 인증서를 발급 받아 공개키를 분배하는 방식을 사용해야만 한다[5].

그러나 P2P 환경에서는 공개키를 분배하기 위해서 공인 인증기관에서 발급한 인증서를 통한 키 교환만이 아니라 E-mail이라든지 전화나 팩스 같은 전송매체나 물리적인 디스켓을 사용해서 피어의 공개키를 분배할 수도 있다. 이렇게 인증서가 아닌 다른 전송매체로 분배되어지는 공개키들은 각각의 피어에 대하여 여러개의 공개키가 존재할 수 있으며, 전송매체와 방법에 따라 신뢰도가 다르다. 이렇게 신뢰도가 다른 다수의 공개키/개인키 쌍을 사용하기 위해서는 모든 Peer가 효과적이고 효율적으로 사용할 수 있는 체계적인 방법으로 저장 및 관리될 필요가 있다. 본 논문에서는 이러한 필요성을 충족시켜 주기 위하여 키 링의 개념을 도입한다.

4.1 키 링(Key rings)

본 논문에서는 다수의 공개키 쌍이 모든 사용자에게 효율적으로 사용될 수 있도록 체계적인 방법으로 저장/관리되기 위해 레파지토리(Repository)에 각 피어들의 공개키/개인키를 저장하는 방식으로 키 링의 개념을 도입한다. 이러한 키 링은 그 피어가 소유하는 공개키/개인키 쌍을 저장하고, 그 피어에 알려진 다른 사용자들의 공개키를 저장하도록 각 피어에 한 쌍의 자료구조로써 제공되게 된다[6].

공개키 링은 공개키 값을 저장하는데 여기에는 자신의 공개키와 다른 사용자의 공개키가 함께 보관된다. 공개키 링에는 추가적으로 공개키의 신뢰 정도를 나타낼 수 있는 값을 같이 저장하여 신뢰도에 의거한 공개키 사용 정책을 적용한다. 공개키 링에는 PKI 기반의 인증서에 의해 무결성이 입증된 공개키 뿐만 아니라 사용자에 의해서 주고받은 공개키 또한 저장될 수 있다. 이때 신뢰정도는 각각의 파라미터와 사용자가 직접 입력

Time stamp	Key ID	Public Key	Owner Trust	User ID	Key Legitimacy	Signature	Signature Trust
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
Ti	$K_{Ui} \bmod 2^{64}$	K_{Ui}	$trust\_flag_i$	$User_i$	$trust\_flag_i$	$Sig[K_{Ui}]$	$trust\_flag_i$
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

[그림 4] Public Key Ring

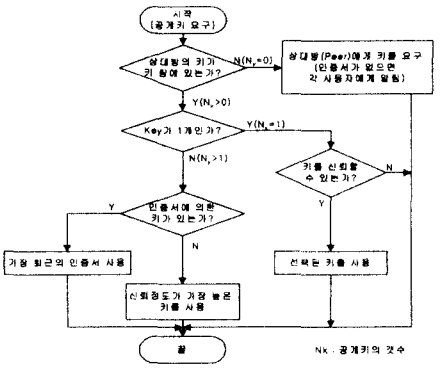
한 신뢰도에 의해서 산출될 수 있다. [그림 4]는 공개키 링의 형태를 나타낸 그림이다.

개인키 링은 공개키 쌍 중에서 자신의 개인키를 저장한다. 개인키를 소유하고 있는 사용자의 시스템에만 저장하고 그 사용자만이 시스템을 액세스한다 하더라도 가능한 한 개인키를 안전하게 보관하기 위해서 개인키 자체를 키 링에 저장하지 않고 특정한 암호 알고리즘으로 암호화하여 저장한다. [그림 5]는 이러한 개인키 링의 형태를 나타낸 그림이다.

Timestamp	Key ID	Public Key	Encrypted Private Key	User ID
⋮	⋮	⋮	⋮	⋮
Ti	K_{Ui} mod 2^{64}	K_{Ui}	$E_{H(Pi)}[K_{Ri}]$	User$_i$
⋮	⋮	⋮	⋮	⋮

[그림 5] Private Key Ring

4.2 공개키 선택 정책



[그림 6] 공개키 선택 순서도

인증서를 이용한 공개키 분배와 인증서가 아닌 다른 전송매체를 이용하여 교환된 공개키를 키 링을 도입하여 사용할 때 하나의 피어에 해당하는 공개키가 여러개 존재 할 수 있다. 따라서 이러한 여러 개의 공개키들 중에 사용할 공개키를 선택하기 위한 정책이 필요하다. [그림 6]은 그러한 정책을 순서도로 나타낸 것이다.

먼저 피어에 해당하는 공개키를 찾고 없으면 상대 피어에게 키를 요구하고 사용자에게 알린다. 다음으로 키가 하나인지 여러개인지를 판별하고 하나일 경우 신뢰도를 측정하여 사용하게 되며 인증서에 의한 키가 있을 경우 우선적으로 사용하고 만약 인증서가 여러개(서로 다른 인증기관에서 발급)가 있으면, 가장 최근의 인증서를 사용한다. 마지막으로 인증서에 의한 공개키가 없고 키 링에 의한 공개키만이 분배되어 있을 경우에는 가장 신뢰도가 높은 공개키를 사용하게 된다.

4.3 적용사례

공인인증기관의 인증서를 사용하게 되면 자신의 신원을 직접 확인해야 하고, 일정액의 사용료를 지불해야 한다. 그러나 P2P 서비스를 사용하는 데에는 금융정보나 증권정보와 같은 중요한 데이터만이 있는 것이 아니라 개인의 사생활처럼 단지 남에게 보이고 싶지 않은 정도의 낮은 중요도를 가지는 데이터도 있다. 따라서, 제안된 방법을 사용하게 되면, 공인 인증기관의 인증서를 받기 위해서 복잡한 절차와 금전적 부담을 줄일 수 있다.

예를 들어 친구와의 편지나 교환일기를 전송할 경우 쌍방은 자신의 파일을 제 3자가 보는 것을 원하지 않을 것이다. 그러나, 이를 위해서 전자상거래에 사용되는 인증서를 발급 받는 것은 부담이 클 것이다. 그러므로 전화나 혹은 디스켓으로 공개키를 분배하거나 잘 알고 있는 제 3자에 의해서 공개키를 분배하고 이를 사용한다면 인증서 발급 부담은 줄일 수 있고, 나름대로의 기밀성은 유지할 수 있다.

5. 결론 및 향후 연구

본 논문에서는 글로벌 P2P 엔진에 보안을 적용하기 위하여 공개키의 관리를 담당하는 보안 관리자의 설계 및 구현에 대하여 소개하였다. 제안한 보안 관리자는 인증서가 아닌 다른 전송매체에 의해 분배된 공개키 또한 저장/관리할 수 있는 기능을 제공하여야 한다.

이를 위해 키 링의 개념을 도입하였고, 이렇게 제안된 키 링의 개념을 사용하는 방법은 인증서에 의한 공개키뿐만 아니라 피어 간에 개인적으로 생성되고 분배된 공개키도 사용할 수 있음으로 보다 순수한 P2P 모델이라고 할 수 있다.

P2P 엔진을 사용하여 전송되는 데이터에는 금융정보나 증권정보와 같은 중요한 데이터만이 있는 것이 아니라, 교환일기나 편지와 같은 금전적인 손해는 없지만 개인의 사생활과 관련된 보다 낮은 중요도를 가지고 있는 데이터 전송시에 효율적으로 키를 분배하여 보안을 지원한다.

또한, 제시된 키 링과 공개키 선택 정책을 사용하여 P2P 서비스를 구현하게 되면 공개키의 인증을 위하여 클라이언트/서버의 방식인 인증기관을 거치지 않고 직접 공개키를 분배할 수 있으므로 순수한 P2P 모델에 보다 근접한다고 할 수 있다.

본 논문에서 제시한 보안 관리자는 실제로 효율적으로 공개키를 공유하여 사용자에게 편리함을 제공할 수 있는지에 대한 검증이 필요하다. 이를 위해서는 글로벌 P2P 엔진과 그 서비스들을 구현하여 보안 관리자와 키 링에 대한 성능을 검증하고, 보안 기능을 제공과 전체 시스템 성능 오버헤드를 최소화할 수 있는 적절한 이해득실(trade-off)을 찾는 것이 향후의 과제라고 보여진다.

6. 참고문헌

[1] Jon Udell, Nimisha Asthagiri, Walter Tuvell, "Security", Peet-to-Peer, O'REILLY, March 2001, pp.354-380

[2] Gnutella, http://www.gnutelladev.com

[3] Gregory Alan Bolcer, Michael Gorlick, Arthur S. Hitomi, Peter Kammer, Brian Morrow, Peyman Oreizy, Richard N. Taylor "Peer-to-Peer Architectures and the Magi Open-Source Infrastructure", December 6, 2000

[4] William Stallings, "Network Security Essentials: Applications and standards", Prentice Hall, Inc., pp. 118-136, 2000.

[5] Marc Branchaud, " A survey of public-key infrastructure", M.S.thesis, Dept. of Computer Science, McGill University, Montreal, 1997

[6] William Stallings, "Protect your privacy: the PGP user's guide", Prentice-Hall PTR, 1995.