제 18 회 한국정보처리학회 추계학술발표대회 논문집 제 9 권 제 2 호 (2002.11)

DRM 유통시스템에서의 보안 통신 모듈 설계 및 구현

성수련*, 정인성, 신용태, 이준석, 정연정
*숭실대학교 컴퓨터학과
e-mail : ssl@cherry.ssu.ac.kr

Security Communication Module Design and Implementation in the DRM distribution system

Su-Lyun Sung*, In-Sung Jung, Yongtae Shin, Junsuk Lee, Yonjung Jung
*Dept. of Computer Science, Soongsil University

요 약

DRM(Digital Rights Management)은 디지털 컨텐츠를 암호화하여 인터넷상에서 안전한 거래를 보장하며, 저작권자의 권리를 보호하는 기술로, 디지털 컨텐츠 유통 문제점에 관한 해결방안을 제시하고 있다. 그러나, 현재 디지털 컨텐츠 유통 주체 상호간 인증을 바탕으로 한 비밀 통신, 통신의 무결성 등을 위한 보안 통신은 제대로 이루어지지 않고 있다. 또한, SSL(Secure Socket Layer)은 현재 보안 통신 메커니즘으로 가장 널리 사용되며, PKI기반 암호 통신 메커니즘이기 때문에 안정성이 높게 평가된다. 이에 본 논문은 DRM 유통 시스템에서 보안 통신을 제공하기 위한 최적화된 SSL 모듈을 설계하며 구현을 목적으로 한다.

1. 서론

최근 인터넷과 통신 기술의 발전으로 인해 기존의 수많은 아날로그 정보가 디지털 정보로 급속하게 변하고 있다. 디지털 정보의 편리성은 정보 소비자의 디지털 컨텐츠에 대한 수요를 더욱더 증가시키고 있으며, 또한 디지털 정보는 제작, 유통이 용이하여 정보 제공자의 디지털 컨텐츠 제작이 급증하고 있다. 이러한, 다양한 디지털 컨텐츠를 판매하는 인터넷 상거래 시스템은 최근 더욱더 활성화되고 있으며, 계속적으로 급성장할 것으로 기대된다.

그러나 디지털 컨텐츠는 복제, 변형, 유포 등이 용이하고, 안전하지 않은 인터넷을 통해 유통되고 있어, 보안과 저작권 문제가 중요한 쟁점으로 대두되고 있다. 따라서 디지털 컨텐츠의 가치가 점차 고도화되는 상황에서 반드시 해결되어야 하는 필수적 사안이다. DRM(Digital Rights Management)은 디지털 컨텐츠를 암호화하여 인터넷상에서 안전한 거래를 보장하며, 저작권자의 권리를 보호하는 기술로, 디지털 컨텐츠 유통 문제점들의 해결방안을 제시하고 있다[1].

이와 같은 DRM을 사용한 유통 시스템은 컨텐츠 생성자, 유통업자, 클리어링 하우스, 소비자 등 다양한 유통 주체로 구성된다. 이때, 컨텐츠와 관련된 저작권을 비롯한 다양한 유통 주체들의 권리는 DRM 유통 시스템에 의해 보호 받을 수 있어야 한다. 이를 위한 선결 요건으로 유통 주체 상호간 인증을 바탕으로 한 비밀 통신, 통신의 무결성 등을 위한 보안 통신이 충족되어야 한다.

현재 보안 통신 메커니즘으로 가장 널리 사용되고 있는 것은 SSL (Secure Socket Layer) 또는 TLS (Transport Layer Security) 이다[2].

SSL은 인터넷 환경, 특히 웹 환경의 발달과 함께 널리 사용되어 왔으며 현재 많은 웹브라우저, 웹서버에서 지원하고 있다. SSL은 어플리케이션에 독립적인 네트웍 레벨의 보안을 제공하며, 공개키 기반 (Public Key Infrastructure) 암호 통신 메커니즘 특성 때문에 현재 웹 환경은 물론이고 일반 기업에서도 자신의 어플리케이션 환경에 맞춰서 최적화된 SSL을 활용하고 있다[3].

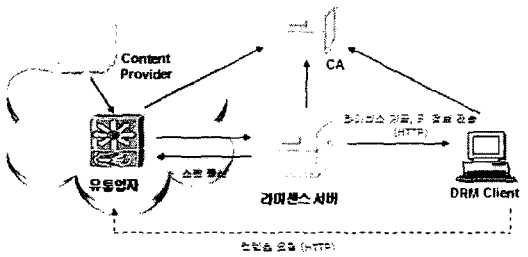
이에 본 논문은 DRM 유통 시스템에서 보안 통신을 제공하기 위한 최적화된 SSL 모듈을 설계하며

구현 결과를 제시한다.

2. 관련 연구

2.1 DRM 유통 시스템 구조

DRM 유통 시스템 구조는 [그림 1]과 같다. DRM 유통 시스템은 컨텐츠를 제공하는 유통업자와 라이센스를 제공하는 라이센스 서버, 컨텐츠를 요청하는 DRM 클라이언트로 구성된다. 유통업자는 컨텐츠키를 관리하며 컨텐츠 키를 이용하여 컨텐츠 공급자가 제공한 컨텐츠를 암호화하여 패키저의 형태로 컨텐츠를 저장한다.

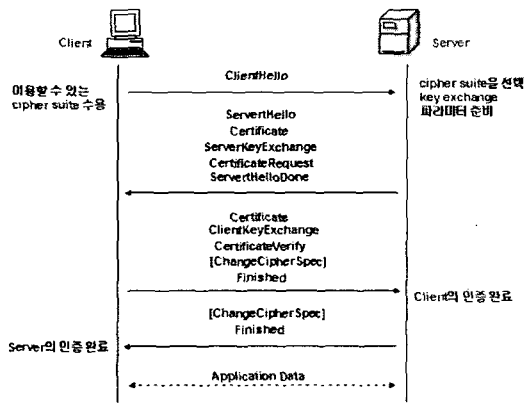


[그림 1] DRM 유통 시스템 구조

라이센스 서버는 해당 컨텐츠 ID 에 대한 라이센스를 발행하며, 컨텐츠 ID 사용 규칙에 따라 라이센스 등록과 라이센스 키 관리를 담당한다. DRM 클라이언트는 컨텐츠를 요청하는 주체가 된다.

2.2 SSL(Secure Socket Layer)

SSL 은 Secure Socket Layer 로써 어플리케이션간의 안전한 통신을 위하여 사용되며 항상 TCP 위에서 동작한다. 또한 클라이언트-서버 모델을 기본으로 한다. 사용자 인증과 비밀키 암호 등이 클라이언트와 서버 사이에 교환되는 데이터를 안전하게 보호하기 위하여 사용된다. SSL 의 동작 절차는 [그림 2]와 같다.



[그림 2] SSL 의 동작 절차

3. SSL 을 이용한 보안 통신 설계

유통업자는 컨텐츠 등록 파일과 컨텐츠키를 함께 사용하여 라이센스 서버에 등록한다. 이때, 컨텐츠 키는 암호화된 채널을 이용하여 라이센스 서버에 안전하게 전송되어 유통업자의 권리를 보호 받을 수 있어야 한다. 또한 라이센스 서버가 DRM 클라이언트에게 라이센스 파일과 사용 규칙 정보의 전송 시, 이 데이터 또한 암호화된 채널을 이용하여 안전하게 전송되어야 한다. 즉, 유통 주체 상호간 인증을 바탕으로 한 비밀 통신, 통신의 무결성 또한 함께 제공되어야 한다.

DRM 유통 시스템에서 보안 통신을 제공하기 위한 모듈은 기본적으로 SSL 메커니즘을 사용한다. 이때, DRM 클라이언트에서 동작하는 보안 모듈의 크기는 클라이언트의 특성을 고려하여 최소화되어야 한다.

이에 본 보안 통신 모듈은 기존의 openssl을 경량화 하며, 소켓 통신과 HTTP 통신에 상관없이 최적화 되어 동작하도록 설계된다.

3.1 접근 방법

암호 통신 개발을 위한 접근 방안은 다음과 같은 3가지 경우로 분류되며 각각의 장단점은 다음과 같다.

1. 자체 메커니즘 개발

openssl을 새로 구현하는 방안이다. 필요한 기능만을 넣어 최적화 하여 개발할 수 있으나 개발된 모듈의 보안성 및 안정성에 대한 검증이 어렵다.

2. 기존의 openssl의 경량화

개발시간과 비용을 줄일 수 있으나 기존의 openssl은 많은 종류의 암호화 알고리즘 및 기능을 지원하기 때문에 모듈이 커질 수 있다는 단점을 가진다.

3. 자체 암호 라이브러리 기반 경량 SSL 개발

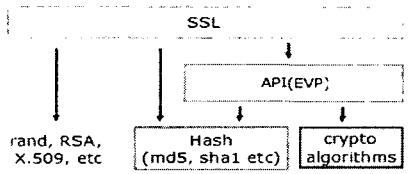
SSL 전체 구조를 그대로 유지함으로 보안성 및 안정성의 잇점을 살릴 수 있지만 자체 암호 라이브러리로 대체하는 과정에서 어려움을 가질 수 있다.

위의 3가지 방안 중 본 논문은 openssl의 경량화와 보안 모듈의 안정성 측면에서 3번 방안을 채택하여 암호 통신 모듈을 개발한다.

3.2 Openssl API 분석

기존의 openssl 모듈을 경량화 하기 위해 openssl을 분석한다. openssl 모듈은 확장성을 위한 목적으로 EVP, BIO와 같은 API를 사용하여 개발되었다. openssl은 크게 SSL 부분과 crypto 부분으로 분류되며 그 사이에 EVP와 BIO 같은 인터페이스가 존재한다. openssl 모듈의 전체 API간의 구조도는 [그림 3]과 같다. random, RSA, x.509와 같은 API들은 SSL API에서 직접 호출되어 사용하지만 DES, RC4, RC5같은 API는 EVP를 거쳐서 사용된다. 또한 md5, sha와 같은 API는 앞의 두 가지 방법 모두로 사용된다.

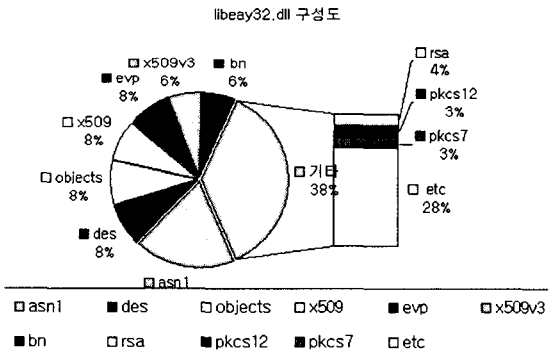
본 논문은 SSL 전체 구조를 살려 기존 openssl의 확장성과 유연성 측면의 장점을 이용한다. 이러한 목적으로 EVP와 같은 중간 인터페이스를 수정하지 않는 방법으로 SSL을 경량화 한다.



[그림 3] openssl API 구조도

3.3 Openssl 경량화 방안

Openssl 의 전체구조를 변경하지 않는 방법으로 openssl 을 경량화 시키기 위해 Amdahl's law 에 의해 많은 부분을 차지하는 API 를 중심으로 경량화 시키는 접근 방법을 취한다. 이에 따라, openssl 의 libeay32.dll 의 API 구성을 살펴보면 [그림 4]와 같다.



[그림 4] Openssl 의 libeay32.dll 구성도

Openssl 의 전체 구성도를 보면 특정 API 가 openssl 의 많은 부분을 차지하는 것이 아니라, 여러 API 에 의해 점유되고 있다. 그렇기 때문에, openssl 경량화 작업은 많은 부분을 차지하는 API 중심이 아닌 각각의 API 를 중심으로 필요 없는 기능을 삭제하는 방안을 취한다.

또한, 크기가 큰 모듈에 속하는 x509 나 DES, RSA 와 MD5, SHA1 과 같은 것들은 자체 개발된 암호화 모듈로 교체하여 전체 openssl 의 효과적인 경량화를 꾀한다.

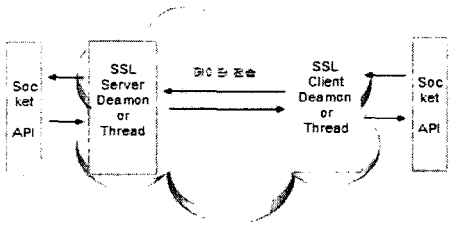
경량화 결과 openssl 전체 성능에는 영향을 미치지 않으며 전체 dll 모듈의 크기를 줄임으로써 클라이언트의 openssl 이용 시 더 낮은 지연 시간을 가지며 보안 서비스를 받을 수 있다. 또한, 클라이언트가 이동 노드일 경우, 이동 노드는 크기가 큰 모듈의 탑재는 불가능 하므로, 이러한 환경에서도 우리의 경량화된 openssl 모듈은 효과적으로 적용할 수 있다.

3.4 DRM 시스템과의 인터페이스 고려

경량화된 openssl 은 유통업자와 라이센스 서버간의 소켓 통신과 라이센스 서버와 DRM 클라이언트 간의 HTTP 통신 시 이용된다. 이 때, DRM 시스템과 openssl 간의 인터페이스 방식은 [그림 5]와 [그림 6]와 같다.

1. 소켓 통신

소켓 통신시, openssl 은 데몬 형식으로 유통업자와 라이센스 서버에서 동작하며 두 개의 프로그램간에 파일을 공유하는 형식으로 인터페이스가 제공된다.



[그림 5] 소켓 통신 인터페이스

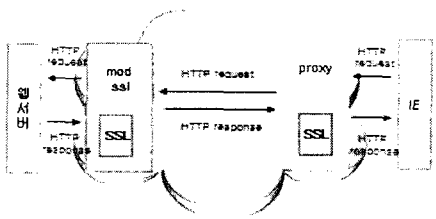
유통업자와 라이센스 서버간의 통신은 ssl 클라이언트와 ssl 서버간의 안전한 채널을 통하여 이루어 지며, 각각 유통업자와 라이센스 서버에서 동작하고 있는 모듈에서 ssl 관련 API 를 호출하거나, 각각 모듈에서 생성된 소켓과 파일을 ssl 관련 모듈로 넘김에 의해 보안 통신을 이용한다.

2. http 통신

웹서버와 openssl 간의 통신은 mod-ssl 을 필요로 한다. 라이센스 서버가 mod-ssl 을 설치하여 SSL 서버의 역할을 수행하게 된다.

라이센스 서버가 클라이언트에게 보내는 HTTP reponse 메시지는 mod-ssl 을 통해 암호화되어 전송된다. DRM 클라이언트에는 프록시를 설치하며 이때 프록시는 웹 브라우저를 대신하여 SSL 세션을 맺는 SSL 클라이언트의 의미를 가지는 프로그램이 된다.

클라이언트가 서버에게 http request 메시지를 전송하면 클라이언트에 있는 프록시는 그 메시지를 받아 openssl 을 이용하여 암호화하여 서버에게 전송한다.



[그림 6] http 통신 인터페이스

4. 실행

경량화한 ssl 을 이용하여 클라이언트와 서버의 실행 결과는 [그림 7]과 [그림 8]과 같다.

아래의 결과는 소켓 통신 시 사용할 ssl 모듈의 실행 결과이다.

[그림 7] ssl 서버의 실행

[그림 8] ssl 클라이언트의 실행

각각의 DRM 모듈은 위의 ssl 모듈을 사용하여 보안 통신을 이용하게 된다.

5. 결론

본 논문은 DRM 유통 시스템에서 보안 통신을 제공하기 위한 최적화된 SSL 모듈을 설계하며 구현하였다.

현재 디지털 컨텐츠 유통 주체 상호간 인증을 바탕으로 한 비밀 통신, 통신의 무결성 등을 위한 보안통신이 제대로 이우러 지지않고 있기 때문에 본 논문은 현재 보안 통신 메커니즘으로 가장 널리 사용되며 안정성이 가장 높게 평가되는 SSL을 이용한 보안 통신 모듈을 설계한다.

이 때, 기존의 openssl 모듈은 자체 개발한 암호 라이브러리로 교체된 경량 화한 SSL 개발하여 DRM 유통 시스템에서 보안 통신을 위해 사용하며, 이와 같은 점은 DRM 유통 시스템의 인프라가 이동 통신일 경우 이동 노드에 효과적으로 적용될 수 있다. 또한, 기존의 Internet explore를 사용하여 통신하지 않는 유통업자와 라이센스 서버간의 소켓 통신 시 인터페이스를 제시하며, 기존의 Internet explore를 이용하여 통신 시 mod ssl을 이용한 인터페이스를 제시하였다.

향후, 본 보안 모듈을 이동 시스템에 적용할 방안을 연구하며, 실제 구현을 통해 구체화할 계획이다.

참고문헌

[1] Joshua Duhl, Susan Kevokian, "Understanding DRMSystem", An IDC White Paper, 2001

[2] AAP, Digital Rights Managemnet for Ebooks : Publisher Requirements version 1.0, 2000.

[3] Chor, B., A. Fiat, and Naor, "Tracing Traitors", in Advances in Cryptology, Proceeding of CRYPTO '94, vol. 839 of Lecture Notes in Computer Science, Spinger-Verlag, pp 257-270, 1994.

[4] Boneh, D. and J. Shaw, "Collusion-secure Fingerprinting for Digital Data", IEEE Transactions on Information Theory, vol. IT-44, no. 5, pp.1897-1905, Sep. 1998.

[5] Pfitzmann, B., and M. Schunter., "Asymmetric Fingerprinting," in advances in Cryptology, Procedings of Eurocrypt '97, vol. 1233 of Lecture Notes in Computer Science, Springer-Valag, pp. 88-102, 1997.

[6] 이창열, MPEG-21 기반 방송 컨텐츠 유통 프로토타입 시스템 개발, 한국전자통신연구원, 연구결과보고서, 2000 년 12.