모바일 클라우드 환경에서 효율적인 키 교환 방식에 관한 연구

이승현*, 정성민*, 정태명**
*성균관대학교 전자전기컴퓨터공학과
**성균관대학교 정보통신대학
{shlee87,smjung}@imtl.skku.ac.kr*, tmchung@ece.skku.ac.kr**

A Study on Efficient Key Exchange in Mobile Cloud Environment

Seung-Hyun Lee*, Sung-Min Jung*, Tai-Myoung Chung**
*Dept of Electrical and Computer Engineering, Sungkyunkwan Univ.
**College of Information and Communication Engineering, Sungkyunkwan Univ.

요 약

무선 인터넷의 접근성 개선과 스마트폰의 등장으로 인해 개인 컴퓨팅 환경이 PC 중심에서 모바일 중심으로 이동하고 있다. 이에 따라, PC 환경에서 존재하던 클라우드 서비스를 모바일 환경에 적용한 모바일 클라우드 서비스가 새롭게 부각되었다. 모바일 클라우드 서비스란 사용자의 모바일 단말기로 언제 어디서나 일반 PC와 동일한 환경의 컴퓨팅 환경을 사용할 수 있는 서비스를 의미한다. 하지만 일반적인 PC 환경에서 사용하던 서비스를 성능이 낮은 모바일 단말에서 수행함으로써 수행시간, 통신 지연 등의 여러 문제가 발생한다. 본 논문에서는 모바일 클라우드 환경에서 통신의 보안을 보장하기 위한 다양한 기술들의 사용과 이러한 기술 사용에 관한 성능에 따른 효율적인 키 교환 방식을 제안한다.

1. 서론

최근 IT 자원의 효율적인 활용을 위한 핵심 기술로 클라우드 컴퓨팅 기술에 관심이 높아지고 있다. 클라우드 컴퓨팅이란 '네트워크를 통해 다수의 서버를 기반으로 한 대 단위 컴퓨팅 자원에 접근하고 이용하는 것'으로 정의할 수 있다. 즉, 사용자는 IT 자원을 서비스 부하에 따라서 실시간으로 확장성을 지원받으며 필요한 만큼 사용하고, 사용한 만큼 비용을 지불하는 서비스를 의미한다[1]. 이러한 PC 중심의 클라우드 컴퓨팅 기술이 무선 인터넷의 접근성 개선과 스마트폰의 등장으로 모바일 환경에 적용되어지고 있으며, 이를 모바일 클라우드 서비스라고 한다. 모바일 클라우드 서비스는 기존 PC 환경과 같이 IT 자원을 사용한 만큼 비용을 지불한다. 그리고, PC 환경과 다르게 모바일 단말기로 언제 어디서나 동일한 환경의 컴퓨팅 환경을 사용할 수 있다는 이점이 있다. 하지만 PC 중심의 서비스를 모바일 환경에서 수행함으로써 암·복호화, 인증, 키 교환 방식 등의 보안요소들로 통신에 성능적인 문제가 발생 가능하기 때문에, 모바일 클라우드는 모바일 단말기 성능과 보안의 강도에 따라 통신에 영향을 줄 수 있다. 따라서, 본 논문에서는 각 모바일 단말기의 성능에 따라 어떤 암호화 방식과 얼마만큼의 키 길이를 사용해야하는지에 대한 여부와 이 같은 방식을 어떤 방법으로 수행해야 하는지 알아본다. 또한, 모바일 클라우드 환경에서 단말기의 성능에 따른 효율적인 키 교환 방식을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 일반적으로 모바일 클라우드에서 사용하는 통신인 SSL을 정의하고, SSL에서 사용되는 보안 기술을 설명한다. 또한 SSL 보안 기술의 특징 및 동작 과정을 알아보고, 모바일 통신 환경에 미치는 영향에 대해 알아본다. 3장에서는 통신 보안 기술에 따른 모바일 단말의 성능 향상을 위한 효율적인 통신방식을 제안한다. 마지막으로 4장에서는 본 논문의 결론을 기술하고 향후 연구 방향을 제시한다.

2. 관련연구

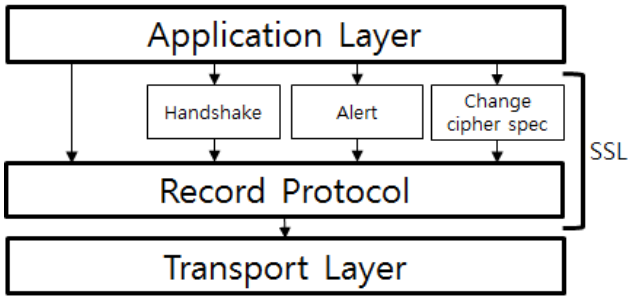
2.1 모바일 클라우드

모바일 클라우드 서비스 형태는 서버 기반의 클라우드 컴퓨팅을 이용하는 단말의 형태가 데스크탑이나 노트북에서 모바일 기기로 변경됨에 따라, 이에 관련한 서비스를 제공하는 것을 목적으로 한다. 모바일 클라우드는 서버와 모바일 단말 간의 통신, 모바일 단말과 단말간의 통신으로 나눌 수 있다. 서버와 모바일 단말간의 통신은 모바일 단말-서버 간 싱크/캐쉬 기술, 작은 크기의 많은 데이터를 처리하기 위한 기술, 모바일 단말과 서버에 대한 보안 기술에 대해 고려해야 한다. 모바일 단말과 단말간의 통신에서는 클라우드 구성 에이전트 기술, 자원 공유를 위한 기

술, 단말의 안정성 기술에 대해 고려해야 한다. 이러한 통신 기술들을 위해 현재 모바일 클라우드는 Secure Sockets Layer(SSL)을 이용한 통신을 주 통신기술로 사용한다. SSL을 사용함으로 모바일 클라우드의 이동성, 안정성, 접근성, 확장성의 기술적 요구사항을 만족한다[2].

2.2 Secure Sockets Layer(SSL)

SSL은 1994년 넷스케이프사에 의해 제안 된 인터넷 통신 규약 프로토콜을 의미한다[3]. 이 프로토콜은 TCP/IP 프로토콜의 어플리케이션 레이어와 TCP레이어 사이에 안전한 통신을 위한 표준이며, 다양한 응용계층의 프로그램들과 쉽게 보안 설정을 할 수 있는 장점을 갖는다. 또한, SSL은 기밀성, 무결성, 인증 기능을 제공한다[4]. 보통 유선 인터넷의 주요 보안 프로토콜로 사용되어져 왔으며, 현재 SSL은 무선 인터넷에서도 활용되어 진다. SSL은 Hand shake, Change cipher spec, Alert 프로토콜로 구성되어 있다. Handshake 프로토콜은 레코드 프로토콜과 보안 파라미터의 요청을 담당하고, Change cipher spec 프로토콜은 Handshake 프로토콜 과정에서 설정 변경을 담당한다. 마지막으로 Alert 프로토콜은 통신 과정에서 발생하는 에러에 대한 메시지 전달을 담당한다[5]. (그림 1)은 SSL의 내부 프로토콜에 대한 구조를 보여준다.



(그림 1) SSL의 내부 프로토콜

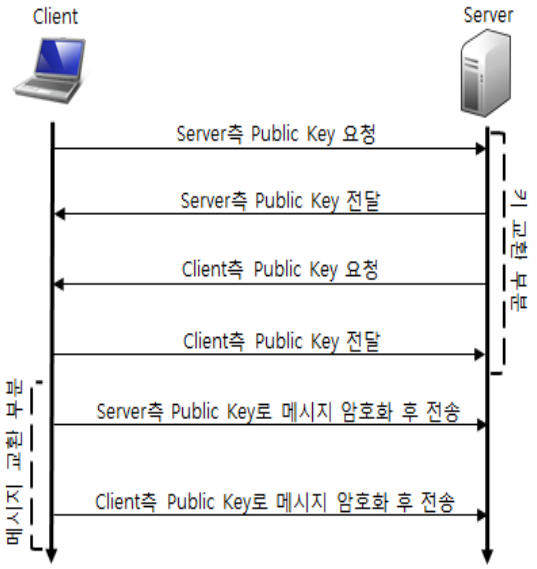
SSL은 웹 보안을 위한 목적으로 개발한 보안 프로토콜이며, 암호화를 위해서 대칭키 알고리즘을 사용한다. 또한 전자서명과 키 교환을 위해 공개키 알고리즘을 사용한다. SSL에서 사용하는 공개키 알고리즘은 DH-anon, RAS, Kerberos, Elliptic Curve 등이 있고, 대칭키 알고리즘은 IDEA, RC4, DES, 3DES, AES 등이 있다. 또한 사용하는 해쉬 알고리즘에는 MD5와 SHA-1이 있다.[6] 아래 (표 1)은 SSL에서 지원하는 알고리즘에 대한 표이다.

(표 1) SSL 지원 알고리즘

	공개키	대칭키	해쉬 함수
SSL	DH-anon DHE-DSS RSA DSA Kerberos Elliptic-Curve	IDEA RC2/RC4 DES/3DES AES Seed Camellia	MD5 SHA

2.3 SSL의 암·복호화 및 키 교환

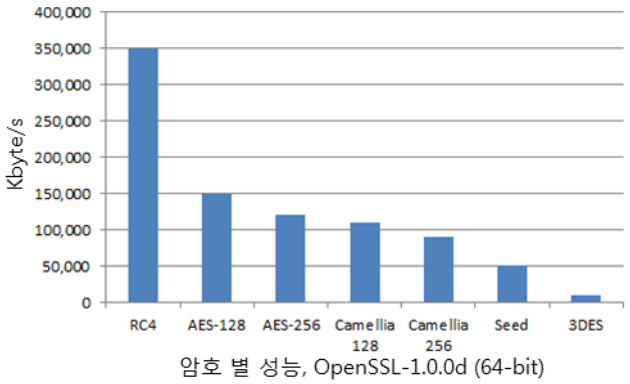
SSL에서 암호화/복호화 과정과 키 교환 과정이 존재하며, 이 과정에서 암호화를 통해 메시지의 기밀성을 보장한다[7]. 방식은 SSL의 1-way와 2-way 방식이 존재하며, 현재 모바일 클라우드는 2-way를 기반으로 동작한다. 1-way SSL방식은 클라이언트에서 서버로 보내는 메시지만 암호화 할 때 사용된다. 이 경우, 서버측은 개인키/공개키를 세트로 갖고 있어야 한다. 또한 클라이언트가 서버로 메시지를 전달하려고 할 때, 서버 측에서 공개키를 획득하여 메시지 암호화 후 전송한다. 수신한 서버측은 그에 해당하는 개인키를 사용하여 메시지를 복호화 하여 메시지를 확인 할 수 있다. 2-way SSL 방식은 양 방향 메시지를 모두 암호화 하며, 서버와 클라이언트 각각 개인키/공개키 세트를 갖고 있어야 한다. 메시지를 송·수신하기 전 서로의 공개키를 획득하고, 이를 통해 요청/응답 메시지를 암호화하여 전송한다. 수신한 메시지는 개인키를 통해서 복호화 한다. 이 같은 동작을 통해 SSL은 기밀성을 보장할 수 있다. 아래의 (그림 2)는 2-way SSL의 동작과정을 나타낸다.



(그림 2) 2-way SSL 동작 과정

2.4 모바일 클라우드 환경에서 SSL의 문제점

SSL은 현재 OpenSSL 프로젝트로 오픈 소스가 존재한다. C 언어로 작성되어 있는 라이브러리 내부에는, 기본적인 암호화 기능 및 여러 유틸리티 함수들이 구현되어 있다. SSL은 보안통신을 위해 보안적인 요소들을 추가한 기술임에 따라, 일반적인 통신보다는 오버헤드가 존재한다. OpenSSL 프로젝트에서는 암호화 성능에 대한 부분을 (그림 3)과 같이 다루고 있다[8]. 하지만 성능과 암호화 강도는 반비례하기 때문에, 성능이 증가하면 암호화 강도는 낮아지게 된다. 또한 SSL 통신의 성능은 클라이언트 측 단말기의 성능에도 영향을 받는다.



(그림 3) OpenSSL통신에서의 암호별 성능

아직까지 모바일 단말과 PC와는 성능적인 차이가 크다. 따라서 현재 PC 환경에서 사용하는 암호화의 성능은 모바일 단말에서는 낮아지게 될 것이고, 모바일 단말에서 3DES와 같은 경우는 오버헤드가 높아 사용이 불가능 할 것이다. 또한, 모바일 단말의 성능은 가지각색이기 때문에 하나의 암호화방식을 채택해서 사용할 수 없다는 문제점이 존재한다. 만약 암호화 방식을 채택하여 사용한다면, 성능이 낮은 단말기에서는 보안 통신에 대한 오버헤드가 커지게 될 것이기 때문이다.

3. 제안 기술

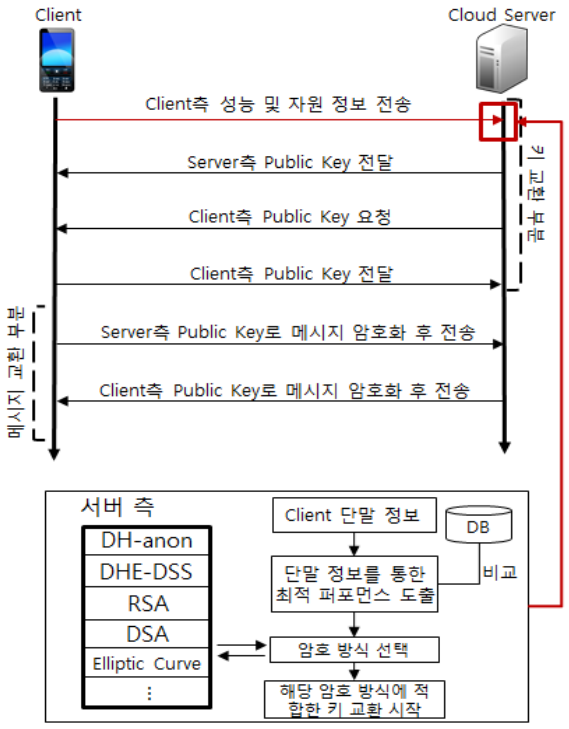
2.4절에서 다룬 문제점의 해결방안으로 본 논문에서는 모바일 클라우드 환경에서 SSL 통신의 효율적인 키 교환 방식을 제안하고, 이를 통해 SSL 통신의 성능적인 면을 증가 시킨다.

3.1 SSL에서 효율적인 키 교환 방식

암호화에는 RSA/Elliptic Curve 등 여러 가지 방식들이 존재한다. 그리고 암호화를 하기 위해서는 암·복호화를 할 수 있는 키가 존재해야 하며, 서버와 클라이언트는 각각에 맞는 키에 대한 정보를 알고 있어야 한다. 각각의 암호화 방식에서 사용하는 키 길이는 서로 다르며, 암호화 강도와 오버헤드 또한 다르다. 이 같은 암호화 방식은 PC와 같이 충분히 암호화를 수행 할 수 있는 성능이면 문제가 되지 않는다. 하지만 모바일 단말의 성능일 경우, 암호화방식이 복잡하거나 키의 길이가 긴 방식이면 오버헤드의 문제가 발생한다. 이에 따라 본 논문에서는 키 교환 전에 모바일 단말의 성능적인 부분의 정보를 취득하고, 그에 알맞은 암호화 방식을 사용하도록 제안 한다. 이러한 방식은 성능이 좋은 단말인 경우에는 복잡한 암호화 방식을 사용하거나 긴 키 길이의 암호화 방식을 사용하여 보안적인 부분을 강화할 수 있다. 또한, 성능이 좋지 않은 단말인 경우에는 다른 암호화 방식보다 짧은 키 길이의 암호화 방식이나 단순한 암호화 방식을 사용하여 단말과 서버간의 암호화 과정에서의 오버헤드를 줄일 수 있는 효과가 있다.

3.2 SSL에서 제안 방식의 동작 및 암호화 과정

제안하는 SSL 통신 방법에서 단말의 성능에 따라 암호화를 선택할 수 있으며, 선택한 암호화를 기반으로 해당 암호키를 교환을 한다. (그림 4)는 제안한 방식에 대한 내용이다.



(그림 4) 제안한 키 교환 방식 동작 구조

1. 단말기는 자신의 CPU, Memory 등 성능적인 정보와 현재 단말 내에서 사용 가능한 자원 정보를 클라우드 서버로 전송한다.
2. 단말기로부터 수신한 정보를 바탕으로 모바일 클라우드 서버는 최적의 성능을 계산하고, 성능에 따라서 사용되어질 암호화 방식이 미리 구분되어 있는 데이터베이스를 기준으로 그에 해당하는 암호화 방식을 선택한다.
3. 서버는 단말의 성능에 따라 선택한 암호화 방식을 클라이언트 측에 자신의 공개키와 함께 전송한다.

4. 서버의 공개키를 수신한 클라이언트는 서버에게 자신의 공개키를 전송한다. 또한, 수신한 서버의 공개키로 자신의 메시지를 암호화 하고, 서버에게 전송한다.
5. 클라이언트로부터 수신한 암호화 메시지를 서버는 자신의 개인키로 복호화 하여 메시지를 확인한다.

이와 같은 키 교환 방식을 통해 SSL 통신에서 단말의 성능에 따른 암호 방식을 선택하는 것이 가능하며, 선택한 암호방식에 따라 SSL 통신에서 오버헤드에 대한 결과가 달라진다.

4. 결론 및 향후연구

일반적인 PC와 성능의 차이가 있는 모바일 환경에서는 암호화 및 키 교환방식에 따른 오버헤드가 통신의 영향을 미친다. 따라서 본 논문에서는 모바일의 SSL 통신 환경에서 기존 PC환경과는 다른 통신 방법을 제안하여, 성능이 낮은 모바일 단말기 혹은 부하중인 단말기이더라도 기존보다 효율적인 통신이 가능하다 라는 결과를 얻었다.

향후 연구에서는, 제안한 통신 방법과 암호화 방식에 따른 실질적인 비교 및 평가가 진행되어야 하고, 클라우드 서버 측과 해당 단말기와의 통신에서 암호화방식에 따른 최적의 성능을 계산하는 과정을 자세히 다루어야 한다.

참고문헌

[1] 김학범, “클라우드 컴퓨팅 환경에서의 보안 관리에 관한 연구”, 경영컨설팅 리뷰, pp.127-144, Feb 2011.
[2] 이강찬, “모바일 클라우드 표준화 동향 및 전략”, 한국통신학회지, pp.44-49, Sep 2011.
[3] IBM : SSL on ISC, Part 1, www.ibm.com/developerworks/kr/library/ac-iscssl1, Jun 2012.
[4] Vipul Gupta, “Performance analysis of elliptic curve cryptography for SSL”, pp.87-94, Feb 2002
[5] 이진우, “SSL/TLS, WTLS의 현재와 미래”, 한국정보보호학회, pp.27-39, Aug 2004.
[6] M S.Bhiogade “Secure Socket Layer”, Informing Science, pp.85-90, Jun 2002
[7] Behrouz A. Forouzan, “TCP/IP PROTOCOL SUITE Fourth Edition”, 2010.
[8] OpenSSL Project, www.openssl.org, Jun 2012.