# DAA 자바 실험모듈 구현을 통한 모바일 DAA 모델 설계

## (Design of a Mobile DAA Model through Java Test Module for the DAA Protocol)

양 석 환 † 이 기 열 ††
(Seokhwan Yang) (Kiyeal Lee)

정 목 동 †††
(Mokdong Chung)

**요 약** 오늘날의 모바일 장비들은 무작위로 움직이는 이동성을 가지고 있어서 보안에 대한 다양한 요구사항을 가지고 있으며 이러한 요구사항을 충족시키기 위해 모바일 장비에 대한 보안기술과 인증기술에 대한 많은 연구가 진행 중이다. 이를 위하여 TCG(Trusted Computing Group)에서는 사용자의 프라이버시를 보호하는 동시에 강력한 인증방법을 제공하기 위하여 보안칩에 해당하는 TPM(Trusted Platform Module)을 설계하고 영지식 증명을 이용한 DAA(Direct Anonymous Attestation) 프로토콜을 제시하고 있다. 본 논문에서는 자바를 이용하여 DAA 프로토콜을 소프트웨어로 구현하고 이를 바탕으로 해서 모바일 환경에 적합한 모바일 DAA 모델을 제시한다.

**키워드** : DAA, TPM, 그룹서명, PKI

***Abstract*** Today's mobile devices have characteristic of random mobility in the heterogeneous networks. Thus, they should have various kinds of security requirements. To satisfy these requirements, there are many researches on security and authentication for mobile devices. TCG (Trusted Computing Group) designed TPM(Trusted Platform Module) for providing privacy and authentication to users. Also TCG suggests a protocol, called DAA(Direct Anonymous Attestation) which uses zero knowledge proof theory. In this paper, we will implement DAA protocol using Java and show the efficiency and the problems in the DAA protocol. Finally, we will suggest an efficient mobile DAA model through Java test module for the DAA protocol.

**Key words** : DAA, TPM, Group signature, PKI

· 이 논문은 정부(교육인적자원부)의 재원으로 한국학술진흥재단의 지원을 받아 수행된 연구임(R05-2004-000-12606-0)
· 이 논문은 2008 한국컴퓨터종합학술대회에서 'DAA 자바 실험모듈 구현을 통한 모바일 DAA 모델 설계'의 제목으로 발표된 논문을 확장한 것임

† 학생회원 : 부경대학교 정보보호학 협동과정
tigergal@chol.com
†† 학생회원 : 부경대학교 컴퓨터공학과
zestgame@daum.net
††† 종신회원 : 부경대학교 컴퓨터공학과 교수
mdchung@pknu.ac.kr

논문접수 : 2008년 2월 4일
심사완료 : 2008년 9월 1일



정보과학회논문지: 컴퓨팅의 실제 및 레터 제14권 제8호(2008.11)

## 1. 서 론

오늘날의 모바일 장비들은 무작위로 움직이는 이동성을 가지고 있어서 보안에 대한 다양한 요구사항을 가지고 있으며 이를 위해 모바일 장비에 대한 보안과 인증에 대한 많은 연구가 진행 중이다. 그러나 제안되고 있는 모바일 장비들에 대한 인증기술은 PKI(Public Key Infrastructure)나 ID 또는 해시기법을 이용한 인증기법들이 주류를 이루고 있어서, 모바일 장비가 가진 자원한계에 대한 문제점 및 약한 보안 안전성에 대한 문제들이 지적되고 있다.

이에 TCG[1]에서는 사용자의 프라이버시를 보호하는 동시에 강한 인증방법을 제공하기 위하여 TPM을 설계하고 영지식 증명을 이용한 DAA[2] 프로토콜을 제시하고 있다. 그러나 DAA 프로토콜은 모바일 환경에 부적합한 특징도 있으므로 본 논문에서는 DAA 프로토콜을 소프트웨어로 구현하고 기존의 실험 결과와 비교하여 DAA 프로토콜의 효율성과 문제점을 제시하고 모바일 환경에 적합한 효율적인 모바일 DAA 모델을 제시한다.

논문의 구성은 다음과 같다. 2절에서는 관련 연구를 다루고, 3절에서는 자바를 이용하여 DAA 프로토콜 구현을 제시한다. 4절에서는 시스템 구현 결과분석을 다루고 5절에서는 모바일 DAA 모델을 제안한다. 마지막으로 6절에서는 결론 및 향후 연구 방향을 논한다.

## 2. 관련연구

### 2.1 TPM

TCG는 이종 컴퓨터 플랫폼에서 컴퓨팅환경의 보안 강화를 목적으로 설립된 산업표준화 기관으로, TPM 모듈을 정의하고 있다[1]. TPM은 플랫폼에 장착되어 보안 정보를 저장하며, 무결성을 보장해주는 플랫폼 모듈이다. TPM은 외부 소프트웨어 공격이나 물리적 공격에 대해 저장된 정보를 보호한다(그림 1).

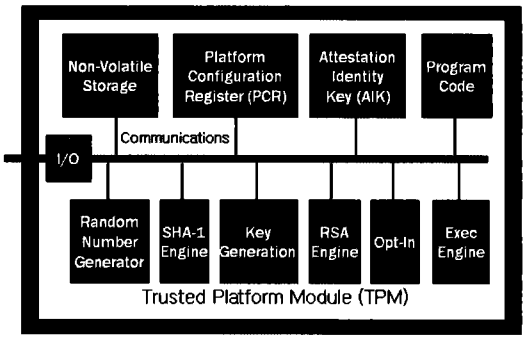


그림 1 TPM의 구성

DAA는 TPM이 가지는 인증방식으로 TPM은 플랫폼의 인증과 신뢰 사슬을 이용하여 검증하는 인증 방식을 수행하며 사용자 및 디바이스에 관련된 정보를 제공하지 않으면서 자신의 신뢰성을 인증할 수 있는 영지식 증명 기법을 이용한다.

### 2.2 영지식 증명

영지식 증명은 검증자에게 비밀 정보는 알려주지 않고 증명의 타당성만을 제공하여 비밀 정보를 소유하고 있다는 것을 확인하는 과정을 말한다[3].

영지식 증명은 주로 NP문제의 어려움에 수학적 기반을 두고 있으며 NP 증명 방식은 대화형 통신을 사용하여 증명하기가 용이하기 때문에 통상적으로 대화형 프로토콜을 이용하여 증명하는 방식을 사용한다.

### 2.3 DAA 프로토콜

DAA는 플랫폼 사용자에 대한 프라이버시를 보호하고 TPM 하드웨어의 원격 인증을 제공하는 서명 기법으로 특징은 다음과 같다.

① TTP(Trusted Third Party) 없이 인증
② 익명성 제공
③ 불법 사용자(rogue TPM) 탐지 가능
④ Strong RSA 및 Decisional Diffie-Hellman Assumption을 사용하여 랜덤 오라클 모델에서 안전[4]

DAA 프로토콜의 구성요소는 인증을 받기 원하는 TPM 사용자, 인증서 발급 기능을 하는 Issuer, TPM 사용자를 검증하는 Verifier로 구성되며 인증 과정은 Setup, Join, Sign, Verify 과정을 다음과 같이 거친다.

① Setup: Fiat-Shamir 휴리스틱을 사용하여 Issuer의 공개키와 개인키를 작성[5]
② Join: TPM은 issuer에게 $N_1^f$ 형태의 정보를 전달하여 자신이 비밀정보 f를 가지고 있음을 증명하고 인증서를 생성할 수 있는 비밀정보($f_0$, $f_1$, v)를 발급받음
③ Sign: TPM은 Join 프로토콜에서 발급받은 인증서와 verifier로부터 받은 $N_2^f$를 사용하여 메시지 서명
④ Verify: verifier 이용 서명유효성 검증

### 2.4 TPM 에뮬레이터 및 DAA 툴

TPM 에뮬레이터 프로젝트는 리눅스를 위한 소프트웨어 기반의 TPM 에뮬레이터를 개발하는 프로젝트이다[6]. DAA 툴은 IBM과 TCG가 함께 개발한 것으로 DAA의 기능을 점검할 수 있는 테스트 기능을 구현하였다[7].

## 3. 자바 이용 DAA프로토콜 구현

본 절에서는 자바를 이용하여 DAA 프로토콜을 구현한다. DAA 모듈에 필요한 실험적인 TPM을 구현하고, 각 TPM사이에서 DAA 프로토콜 인증 과정을 시뮬레이션 한다.

TPM은 DAA 모듈과 공개키 및 비밀키, 서명구조 클래스 그리고 각 TPM들과 통신을 하기 위한 통신 모듈로 구성되며 DAA 모듈은 DAA의 기본 프로토콜인 Setup 프로토콜, Join 프로토콜, Sign 프로토콜, Verify 프로토콜로 이루어진다(그림 2).

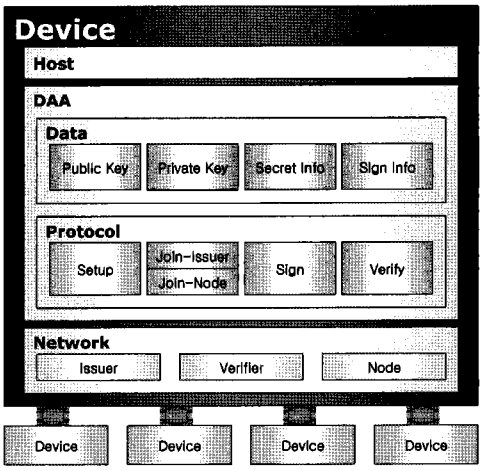


그림 2 DAA모듈 포함 Device 구조

### 3.1 Setup 프로토콜

Setup 프로토콜은 Issuer의 공개키, 비밀키 쌍을 생성하고 공개키를 하위 노드에게 전송하는 과정이다. Issuer는 n=p*q를 만족하는 p, q와 이차잉여 그룹에 대한 랜덤 생성자 g'를 선택하여 공개키를 생성하고, p'=2p+1, q'=2q+1를 만족하는 p'*q'를 비밀키로 저장한 후 공개키를 하위 노드에게 전송한다. 2048bit로 이루어진 공개키, 비밀키를 비롯하여 BigInteger 형의 파라미터와 연산을 사용한다.

### 3.2 Join 프로토콜

Join 프로토콜은 노드들이 DAA 프로토콜을 이용하기

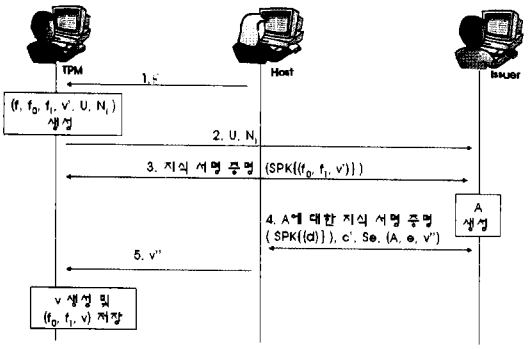


그림 3 Join 프로토콜 처리과정

위한 인증서를 발급 받는 과정이다. 인증서를 발급 받기 위해서 노드는 TPM에서 생성한 비밀 정보를 가지고 있다는 것을 영지식 증명을 통해서 증명한다(그림 3).

### 3.3 Sign 프로토콜

Sign 프로토콜은 TPM를 가진 노드가 Join 프로토콜에서 얻은 비밀정보와 Issuer의 공개키를 이용하여 전자서명하는 과정이다. Sign 과정에서는 인증서, Issuer의 공개키, 메시지를 이용하여 서명을 만든다(그림 4).

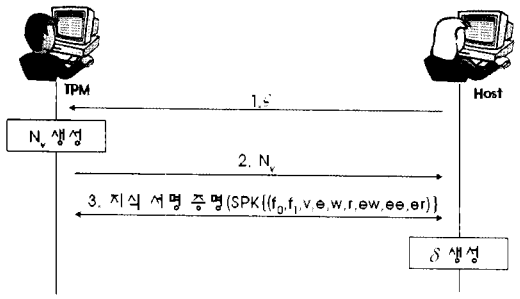


그림 4 Sign 프로토콜 처리과정

### 3.4 Verify Protocol

Verify Protocol은 전자서명에서 공개된 값들과 Issuer의 공개키를 이용하여 영지식 증명과 자신의 유효성을 판단하기 위한 정보를 생성 및 계산하여 전송받은 전자서명의 값과 일치하는지 비교한다.

## 4. 시스템 구현 결과 분석

실제 모바일 환경에 비하여 PC기반의 소프트웨어 DAA는 사용가능한 자원 환경면에서 큰 차이를 가지고 있으나 특정 환경간의 성능 및 자원사용량의 비교를 통해 실제 모바일 환경에 적용했을 경우를 예측해 볼 수 있다.

DAA프로토콜의 구현환경은 표 1,2와 같다.

표 1 PC 구현환경

| |
|---|
| - CPU: Intel Pentium Dual Core 3.0GHz |
| - RAM: 1 GByte |
| - OS: Windows XP, Service Pack 2 |
| - Network: TCP/IP, LAN, Wireless LAN |

표 2 PDA 구현환경

| |
|---|
| - CPU: Intel PXA 2790 624Mhz |
| - RAM: 256 MByte |
| - OS: MS Windows CE 5.0 |
| - Network: Wi-Fi [802.11b], Bluetooth |
| - JVM: IBM J9 |

표 3은 2048 bit의 키에 대한 각 프로토콜 수행시간 비교결과를 나타내며 Join, Sign, Verify 보다 Setup 프로토콜의 소요시간이 약 200배 정도 많이 걸린다는 것을 알 수 있다. Setup 프로토콜에서는 다양한 파라미터를 생성하고 유효성을 검증하는 역할을 수행하므로 긴 수행시간을 필요로 한다.

표 4는 DAA의 수행 시간 중 한 번만 수행되는 Setup 프로토콜을 제외한 Join, Sign, Verify 프로토콜의 총 수행시간과 PKI를 동일한 환경에서 구현한 (CA(Certification Authority)와 CRL(Certificate Revocation Lists) 조회작업 제외) 프로토콜의 전체 수행시간을 각각 1,000회의 반복수행을 통하여 비교한 결과를 보여준다. 표 4는 표 3과의 비교를 위하여 2048bit의 키를 사용하였다.

표 4에서 평균 수행시간은 PKI가 짧지만 편차가 큰 것을 알 수 있다. 반면 DAA는 고른 분포를 보인다. PKI의 경우 CRL의 크기가 커질수록 긴 수행시간을 필요로 하지만, DAA는 평준화된 수행시간을 가지며, CA가 필요 없기 때문에 수행시간 손실이 거의 없다. 따라서 DAA는 PKI보다 더욱 안정적이고 효율적이므로, 모바일 환경에서의 보안은 PKI에 의한 방법보다는 CA의 부담에서 자유로운 DAA 기술을 이용하는 것이 효율적이다.

표 3 PC에서의 수행시간 (단위: msec)

| 프로토콜 | 최소시간 | 최대시간 | 평균시간 |
|---|---|---|---|
| Setup | 478,844 | 522,188 | 489,257 |
| Join | 2,469 | 3,062 | 2,542 |
| Sign | 2,390 | 3,031 | 2,409 |
| Verify | 2,234 | 2,688 | 2,319 |

표 4 PC에서 PKI와 DAA 수행시간 (msec)

| 구분 | 최소시간 | 최대시간 | 평균시간 |
|---|---|---|---|
| PKI | 468 | 24,610 | 4,215 |
| DAA | 7,093 | 8,781 | 7,270 |

표 5 PC에서의 검증과정 포함여부에 따른 Setup 수행 시간 (단위: msec)

| | 검증 포함 | 검증 미포함 |
|---|---|---|
| 최소시간 | 10,656 | 987 |
| 최대시간 | 52,141 | 32,875 |
| 평균시간 | 19,376 | 9,307 |

표 6 PC와 PDA환경에서의 Setup 프로토콜 수행시간 비교 (단위: msec)

| | PC | PDA |
|---|---|---|
| 최소시간 | 987 | 913,487 |
| 최대시간 | 32,875 | 4,141,472 |
| 평균시간 | 9,307 | 1,882,388 |

표 7 FFT의 계산시간 (단위: msec)

| Maker | DSP | 기수2실수 | 기수2복소수 |
|---|---|---|---|
| Analog Devices | ADSP2100 (8MHz) | 66.1 | |
| Texas Instruments | TMS320C20 (40MHz) | 31.82 | 45.0 |
| Texas Instruments | TMS320C30/31 (33.3MHz) | 1.67 | 3.75 |
| Motorola | DSP56001 (27MHz) | | 2.45 |
| Motorola | DSP96002 (33.3MHz) | 0.8 | |

표 5는 Setup에서 검증과정 포함 여부에 따른 수행시간 비교결과이다.

표 5에서 보면 Setup 프로토콜에서 검증과정이 가장 많은 시간을 사용하고 있음을 알 수 있다. 즉 검증과정을 개선한다면 더욱 효율적인 결과를 예측할 수 있다.

표 6은 Setup 프로토콜에 대한 PC 및 PDA환경에서의 수행시간 비교 결과를 나타낸다. PDA 장비의 제한성을 고려하여 키의 크기는 512bit를 사용하였다.

표 7은 소프트웨어 환경과 하드웨어 칩 환경의 속도 차이를 비교하기 위해서 Real-Time FFT(Fast Fourier Transform) Analyzer를 DSP(Digital Signal Processing)를 이용하여 구현하였을 때의 1024 Point FFT에 대한 계산 시간을 보여준다[8]. 소프트웨어에 의한 계산이 수초에 이르는 것을 고려하면 DSP 칩을 이용한 계산은 PC에서의 계산시간 보다 수십~100배의 속도 향상이 있음을 알 수 있다. 이결과로 부터 DAA를 보안 칩으로 구현하면 소프트웨어 구현 보다 최소 몇 십 배의 속도 향상을 가져올 수 있을 것으로 예측된다.

## 5. 모바일 DAA 모델

4절에서 제시한 DAA와 PKI 그리고 하드웨어 칩과 소프트웨어에 의한 구현을 비교함으로써 모바일 환경에 적합한 효율적인 모바일 DAA 모델을 제시한다. 지금까지의 실험결과를 종합하면 DAA 기술은 실제 모바일 장비에 적용할 때 Setup 프로토콜의 많은 계산량이 문제가 될 것으로 판단할 수 있지만, Setup 프로토콜은 Issuer에 대하여 처음에 단 한번만 수행됨을 고려하면 DAA 프로토콜을 모바일 환경에도 적용할 수 있다고 판단된다. 표 8은 PKI와 DAA의 비교를, 표 9는 DAA를 소프트웨어와 하드웨어 구현 결과를 보인다.

표 8 PKI와 DAA 비교

| | PKI | DAA |
|---|---|---|
| 수행시간 | 편차가 큼 | 편차가 작음 |
| CA사용 | CA사용<br>많은 부하 | CA 미사용 |
| 계산량 | 복잡한 계산 수행<br>모바일 장비 적용 어려움 | Setup외의 프로토콜은<br>계산량이 적고 단순함 |

표 9 S/W와 H/W 칩에서의 성능 비교

| | 소프트웨어 | 하드웨어 칩 |
|---|---|---|
| 연산응용력 | 강함 | 제한됨 |
| 수행속도 | 저속 | 고속. 소프트웨어 구현 보다<br>수십~100배 이상 빠름 |

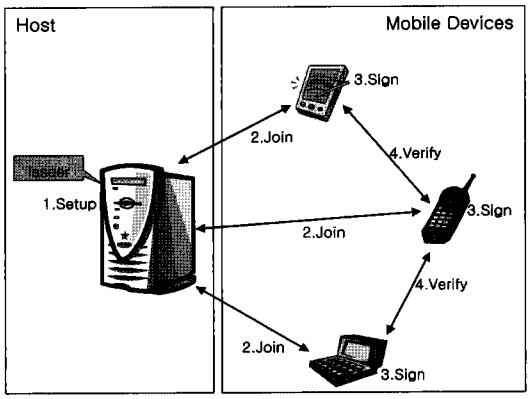


그림 5 모바일 환경에 사용 가능한 모바일 DAA 프로토콜 모델

그러나 실제 모바일 장비에 적용하기 위해서는 Setup 프로토콜에서의 개선이 여전히 필요하다. 가령 Issuer 장비 로드 밸런싱(load balancing) 연구나 Setup의 부하를 줄이는 네트워크 모델에 대한 연구도 필요하다.

그림 5는 본 논문에서 제안하는 모바일 DAA 모델이며, 아래와 같은 작업을 수행함으로써 모바일 환경 보안을 담당하게 된다.

① Setup프로토콜 (호스트에서 수행)

- 나머지 프로토콜보다 200배 이상의 시간이 걸리므

로 모바일에서 수행 어려움
- Issuer의 공개키, 개인키 및 전체 프로토콜의 파라미터를 생성, 검증함

② Join프로토콜(모바일과 Issuer사이 수행)
- 비밀 정보를 Issuer로부터 발급 받음

③ Sign프로토콜 (모바일 장비에서 수행)
- Join 프로토콜을 통해 발급받은 인증정보를 이용하여 메시지 서명

④ Verify프로토콜(모바일 장비에서 수행)
- 서명의 유효성 검증

## 6. 결론 및 향후 연구

본 논문에서는 자바를 이용하여 DAA 프로토콜을 구현하였고 실험적인 TPM 모듈을 정의하고, DAA 프로토콜을 통한 각 노드들의 인증 및 통신을 테스트 하였다. 또한 모바일 환경에 사용가능한 모바일 DAA 모델을 제시하였다. DAA 프로토콜을 이용한 인증 모델은 CA 없이 인증이 가능하므로 CA를 필요로 하는 PKI 기반 인증시스템보다 모바일 환경에 적합하다고 판단되며 Strong RSA와 Decisional Diffie-Hellman 가정을 이용하여 강력한 안전성을 가지므로 기존의 인증모델을 대체하기 위한 한 방법이 될 수 있다.

그러나 영지식 증명을 이용한 인증으로 인한 많은 계산량은 아직 개선의 여지를 가지고 있다고 판단되며, 모바일 환경에 적용 가능한 수준까지 계산량을 줄이기 위한 연구가 필요할 것이다. 향후 자바 기반을 C언어를 이용한 모바일 기반으로 확장해서 제안한 모바일 DAA 모델의 효율을 높일 계획이다.

## 참 고 문 헌

[1] TCG, TCG Specification Architecture Overview Specification Rev. 1.3, 2007.
[2] E. Brickell, et al., "Direct anonymous attestation," In Proc. of the 11th ACM Conf on Computer and Communications Security, ACM Press, 2004.
[3] Shafi Goldwasser, et al., "The knowledge complexity of interactive proof-systems," SIAM Journal on Computing Archive, Vol.18, pp. 186-208, 1989.
[4] Ronald Cramer, "Signature Schemes based on the Strong RSA Assumption," ACM TISSEC Vol.3, No.3, pp. 161-185, 2000.
[5] A. Shamir, "How to Share a Secret," Communication of the ACM, Vol.22, No.11, pp. 612-613, 1979.
[6] Mario Strasser, "Software-based TPM Emulator for Linux," 2004.
[7] Roger Zimmerann, et al., "IBM Direct Anonymous Attestation Tools - TPM Test Suite" Release 1.2.20, 2005.
[8] 박선호, 디지털 신호처리의 기초와 DSP 응용실무, 국제테크노정보연구소, 2002.