모바일 환경에서 XML 기반 전자 서명 시스템의 성능 평가

정회원 김 남 윤*, 황 기 태**

The Performance Evaluation of XML-based Digital Signature System on Mobile Environment

Namyun Kim*, Kitae Hwang** *Regular Members*

요 약

최근 인터넷과 모바일 통신의 발전으로 인해 전자 서명을 이용한 전자 상거래(M-Commerce)가 널리 사용되고 있다. 모바일 환경은 유선 환경에 비해 단말기의 하드웨어와 네트워크가 취약하다. 따라서 효율적인 시스템 구축을 위해서는 전자 문서의 처리, 전자 서명 생성, 모바일 네트워크 대역폭이 성능에 미치는 영향에 대한 체계적인 분석이 요구된다. 본 논문에서는 이를 위해 전자 계약 시스템을 모델로 하여 XML을 기반으로 하는 전자 계약서와 RSA 전자 서명 알고리즘을 사용한 테스트 시스템을 구축하였다. 그리고 전자 서명 서비스의 수행 시간에 영향을 주는 요소를 분석한 후, 현재 상용화되고 있는 Compaq iPAQ 3850 단말기와 CDMA 2000 1x 네트워크를 이용하여 모바일 전자 서명 시스템의 성능을 체계적으로 분석하였다.

Key Words: 전자 서명(digital signature), 모바일 통신(mobile communication), XML, 성능 분석(performance analysis), RSA

ABSTRACT

Due to development of internet and mobile communication, M-commerce applications that utilize the digital signature are widely used these days. The mobile environment is weaker than the wired internet environment. Thus, it is highly required to systematically analyze the effect of processing the digital document and the digital signature algorithm, and low bandwidth of the mobile network on the system performance. In this paper, we have constructed the mobile contract system which has the XML based digital contract and the RSA signature algorithm as the test system. And we have derived the performance parameters and then measured in detail the runtime performance of the mobile digital signature system with the Compaq iPAQ 3850 PDA and CDMA 2000 1x mobile network.

Ⅰ. 서 론

최근 인터넷과 모바일 통신의 발전으로 인해 모바일 단말기를 이용한 전자 상거래(M-Commerce)가 널리 사용되고 있다. 모바일 전자 계약(M-Contract), 모바일 뱅킹(M-Banking), 모바일 금융 거래 (M-Payment) 등은 M-Commerce의 대표적인 예라고 할 수 있다[1, 2]. 이러한 M-Commerce 시스템은

* 한성대학교 정보공학부 조교수(nykim@hansung.ac.kr), ** 한성대학교 컴퓨터공학부 부교수 (calafk@hansung.ac.kr),
논문번호 : 030502-1113, 접수일자 : 2003년 11월 13일
※본 연구는 2004학년도 한성대학교 교내 연구비 지원 과제임.

서비스의 안전을 보장하기 위해 보안 서비스를 제공하고 있다. 일반적으로 기밀성이 요구되지 않는 환경에서는 전자 서명을 수행함으로써 무결성, 인증, 부인 봉쇄 서비스를 제공한다. 전자 서명 기법은 모바일 금융 거래나 모바일 전자 계약 시스템 등에 활발히 적용되고 있다. 예를 들어 자동차 보험이나 방문 카드 가입 시 계약서에 전자 서명을 추가하여 보안 기능을 제공하고 있다.

한편 이러한 모바일 전자 서명 시스템은 유선 기반 환경과는 달리 모바일 단말기의 제한된 메모리 및 CPU 처리 능력, 모바일 네트워크의 낮은 대역폭 등의 제약 조건을 가진다. 최근에 모바일 환경의 성능 제약 요소들을 해결하기 위해 인증서 형식의 간소화[3], 인증서의 효율적 검증 기법[4], 효율적인 서명 알고리즘 개발[5] 등의 노력이 진행되어 오고 있다.

그러나 모바일 전자 서명 시스템의 성능을 개선하고자 하는 많은 노력들에도 불구하고, 모바일 전자 서명 시스템의 성능 요소들에 대한 분석이나 실측에 대한 심도 있는 연구가 부족하다. 즉, 모바일 전자 서명 시스템에서 나타나는 성능 저하의 원인을 설명하고 해석할 수 있는 분석 도구의 개발이 거의 이루어지지 않았으며, 현재의 모바일 단말기 하드웨어 수준에서 성능 개선이 시급한 요소에 대한 정량적, 정성적인 데이터가 부족한 현실이다.

본 논문에서는 모바일 환경에서 전자 서명 시스템의 성능 요소를 분석하고 실측을 통해 각 요소가 성능에 미치는 영향을 제시하고자 한다. 성능 요소는 크게 전자 문서를 해석하고 단말기 화면에 출력하는 성능, 전자 서명을 생성하거나 서명을 검증하는 성능, 단말기에서 서명 값 및 전자 문서를 서버로 전송하는데 소요되는 시간 등으로 분류한다. 본 논문에서는 XML 기반의 전자 계약서를 대상으로 RSA 전자 서명을 실행하는 전자 서명 시스템을 테스트 시스템으로 구현하고 상용 모바일 환경에서 성능 요소들의 수행 시간을 측정하였다. 본 논문의 분석 및 실측 결과는 M-Commerce 시스템에 대한 설계, 하드웨어 및 알고리즘 연구 시 기초 데이터로 중요하게 이용될 수 있다.

본 논문의 구성은 다음과 같다. 2 절에서 전자 서명 시스템 평가 모델에 대해 설명하고 3 절에서는 성능 요소를 분석한다. 그리고 4 절에서는 테스트 시스템에서 실측한 결과를 제시 분석하고 5 절에서 결론을 맺는다.

Ⅱ. 전자 서명 시스템 평가 모델

본 절에서는 전자 서명 시스템의 전체적인 구성에 대해 설명한 후, 성능 요소 및 전자 문서와 소프트웨어의 구성에 기술한다.

1. 테스트 시스템 구성

M-Commerce의 일반적인 응용 모델은 클라이언트-서버 구조를 이루며 이들 사이의 네트워크는 클라이언트 단에 모바일 네트워크로 연결되고 서버 쪽에서는 유선으로 연결된다. 이 사이에는 WAP (Wireless Application Protocol) 게이트웨이가 존재하여 모바일과 유선 네트워크의 접속 변환 및 프로토콜 변환을 돕는다. 본 논문은 XML 형태의 전자 계약서를 바탕으로 전자 서명을 수행한 후 서버로 전송하는 테스트 시스템을 구축하였다(그림 1).

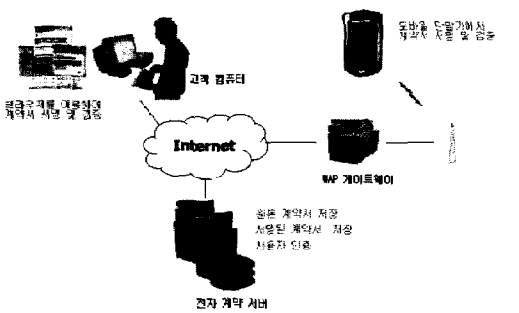


그림 1. 테스트 시스템 모델(전자 계약 시스템)

모바일 단말기로는 WinCE 기반의 Compaq iPAQ 3850 모델을 사용하였으며 전자 계약 서버는 Window 2000 서버에서 IIS 웹 서버를 구동하고 JSP엔진으로 Tomcat을 사용하였다. 그리고 SQL 서버를 데이터베이스로 사용하였다.

테스트 시스템은 다음과 같이 동작한다. 모바일 단말기를 가진 고객이나 영업 사원이 전자 계약 서버에 접속하여 계약서를 다운 받고 계약서에 필요한 필드 값을 채운 뒤 전자 서명하여 서명 값을 자신의 계약서와 함께 서버에 전송한다. 고객은 필요 시 서버에 접속하여 자신의 계약서를 확인할 수 있다.

2. 성능 요소

전자 서명 시스템에서 모바일 단말기와 전자 계약 서버간의 작업은 그림 2와 같이 다운 스트림

(down stream), 업 스트림(up stream)으로 분류된다. 본 논문은 성능 평가를 위해 다음의 성능 요소들을 정의한다.

P_{view} : 단말기에서 전자 문서의 화면 출력 성능
P_{comp} : 단말기에서 새로운 전자 문서 완성 성능
P_{sign}: 전자 서명 생성 성능
P_{verify} : 전자 서명 검증 성능
P_{up} : 단말기에서 네트워크 전송 성능
P_{down} : 서버에서 네트워크 전송 성능
P_{save} : 서버에서 전자 문서를 저장하는 성능
P_{read} : 서버에서 전자 문서를 찾고 준비하는 성능

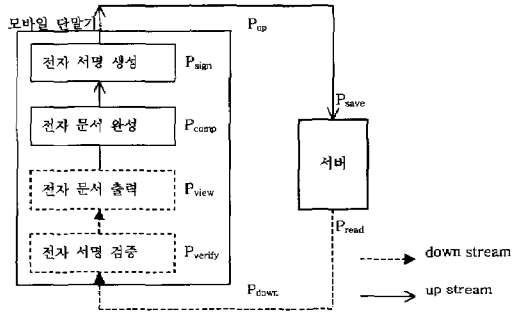


그림 2. 전자 서명 시스템의 작업 과정 및 성능 요소

다운 스트림은 단말기에서 전자 문서를 다운받고 이를 해석하여 화면에 출력하는 과정을 통칭하며, 성능은 <P_{read}, P_{down}, P_{verify}, P_{view}> 와 같이 표현한다. 업 스트림은 단말기에서 사용자의 입력을 바탕으로 전자 문서를 완성하고 전자 서명을 한 후, 전자 문서와 서명 값을 서버에 전송하는 과정이다. 이 과정은 <P_{comp}, P_{sign}, P_{up}, P_{save}> 로 표현된다. 다운 스트림에서 사용자가 자신의 계약서를 확인하는 작업의 경우에만 P_{verify} 의 요소가 존재하며 계약서를 새로 작성하는 경우에는 필요치 않다.

본 논문에서는 P_{save}, P_{read} 의 성능 요소를 분석 대상에서 제외한다. 그 이유는 서버의 처리율이 동시 접속한 클라이언트의 개수에 의존적이고 일반적인 웹 서버와 데이터베이스 서버의 성능 패턴을 크게 벗어나지 않을 것으로 판단하기 때문이다. 또한 P_{down} 와 P_{up} 는 유사한 특징을 나타내기 때문에 P_{down} 는 분석 대상에서 제외하였다.

3. 전자 문서

전자 계약서 등과 같은 전자 문서를 범용 문서 편집기를 이용하여 작성하면, 모바일 단말기 상에서 문서의 크기, 호환성, 사용자 인터페이스 구현 등의 문제점이 존재한다. 예를 들어 데스크탑 환경에서 사용되는 마이크로소프트사의 Word 7.0은 한 글자만 입력하고 저장해도 19 KB나 차지한다. 따라서 범용 문서 편집기로 저작된 전자 문서를 사용하면 단말기의 메모리를 낭비하고 서버와의 통신 시간이 길어지는 단점이 존재한다. 또한 모바일 단말기는 WinCE, Palm, Symbian 등 다양한 운영 체제를 사용하므로 범용 문서 편집기로 작성된 전자 문서는 호환성에 문제가 있다. 한편, PDA에 수행 가능한 Word 편집기는 문서의 크기가 작지만 전자 계약서에 필요한 입력 박스나 버튼과 같은 사용자 인터페이스를 표현할 수 없는 단점이 있다. 또한 HTML은 문서의 표현에 초점을 두었기 때문에 문서의 구조나 데이터를 표현할 수 없는 단점이 있다.

본 논문에서는 이러한 단점을 극복한 XML(eXtensible Markup Language) 기반 전자 문서를 테스트 시스템에 도입하고 XML 태그 셋을 정의하였다. XML은 구조화된 문서와 데이터를 표현하기 위한 마크업 언어이다[6]. 현재 많은 응용 프로그램들이 자신의 데이터 구조를 표현하기 위해 XML을 사용하고 있다. 본 논문에서 정의된 XML 기반 전자 문서는 윈도우 컨트롤을 이용한 다양한 인터페이스를 지원한다.

4. 소프트웨어 구현

그림 1의 테스트 시스템에 구현된 소프트웨어 모듈은 단말기에서 실행되는 모듈과 서버에서 실행되는 모듈로 구분되며 그림 3과 같다. iPAQ 단말기에는 ME(Mobile Explorer)가 실행되며 ME는 서버에 접속하여 웹 페이지를 가지고 온다. 단말기의 전자 서명 프로그램은 ActiveX 코드로 작성되어 웹 페이지에 내장되며, 표 1과 같이 7 개의 모듈로 구성된다. 서버 상에는 HTML 페이지들과 전자 문서 입출력을 위한 JSP 프로그램이 작성되었다.

표 1. 모바일 단말기상의 소프트웨어 (ActiveX Control)

모듈	기능
DocDown	서버에 전자 계약서 요청 패킷 전송, 서버로부터 전자 계약서 수신
DocUp	HTTP를 이용하여 서버로 서명 값과 전자 계약서 전송
DocView	1) XMLParser 를 이용하여 전자 계약서를 트리 형태로 변환 후, 윈도우 콘트롤을 이용하여 전자 계약서를 단말기에 출력 2) 사용자가 윈도우 콘트롤에 입력한 내용을 XML 노드에 반영
XMLParser	XML 형태의 전자 계약서 파싱
XMLBuilder	사용자가 입력한 계약서 내용이 저장된 노드를 재구성하여 XML 문서 생성
DocSign	XMLBuilder를 이용하여 재구성된 계약서에 전자 서명 생성
DocVerify	전자 서명 검증

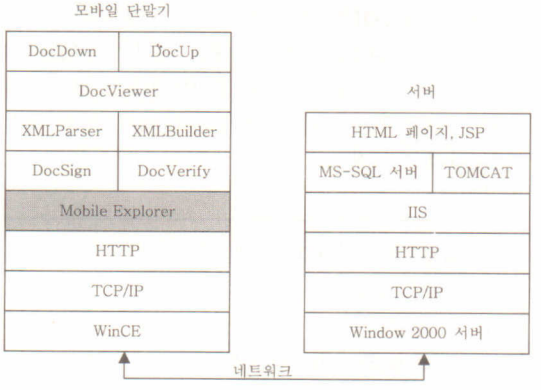


그림 3. 모바일 단말기와 서버상의 소프트웨어 계층

Ⅲ. 성능 요소 분석

1. 전자 문서 처리 성능 : P_{view}, P_{comp}

P_{view} 는 XMLParser와 DocViewer 모듈의 성능을 표현한 것으로서, 전자 계약 서버로부터 받은 XML 전자 계약서를 DOM API[7]를 이용하여 XML 문서의 각 노드의 값을 분석하고 윈도우 컨트롤(Edit Box, Radio Button, Combo Box)을 이용하여 화면에 출력하는 성능이다. 그리고 P_{comp} 는 XMLBuilder와 DocViewer 의 성능을 표현한 것으로서, 사용자가 입력한 정보를 윈도우 컨트롤에서 읽은 후 XML 형태의 전자 계약서를 재구성하는 성능이다.

일반적으로 모바일 단말기는 액정 화면이 작아서 한번에 모든 XML 문서를 표현할 수 없다. 예를 들면, Compaq iPAQ의 경우 액정 화면이 320x240 이고 다른 PDA의 경우도 이와 유사한 크기로 작은 편이다 따라서 본 논문에서는 단말기 화면을 고려하여 하나의 화면을 나타내는 XML 태그를 정의하였다. 결국 P_{view} 는 다음의 두 가지 요소에 의존적이다.

① dSize : 하나의 화면을 나타내는데 필요한 XML 문서의 크기
② dTime : 하나의 화면에 윈도우 컨트롤을 이용하여 XML을 출력하는데 소요되는 시간

XML 문서의 전체 크기를 n이라고 할 때 P_{view} = n/dSize * dTime 로 표현할 수 있다. P_{comp} 성능도 위 두 가지 요소에 의존적이며 dTime 은 하나의 화면에 있는 컨트롤에서 데이터를 읽은 후 XML을 재구성하는데 소요되는 시간으로 정의된다.

2. 전자 서명 처리 성능 : P_{sign}, P_{verify}

1) 전자 서명 알고리즘

전자 서명은 전자 계약서나 세금 계산서 등 전자 문서에 작성자의 서명을 생성, 부착함으로써 인증, 무결성, 부인 봉쇄 기능을 제공한다[8]. 일반적으로 전자 서명은 암호화 및 복호화 시에 서로 다른 키인 공개 키(Public Key)와 개인 키(Private Key)를 사용하는 PKI(Public Key Infrastructure) 기반의 공개 키 암호 시스템을 이용한다[8]. 사용자는 두 키를 쌍으로 가지며 공개 키는 모든 사용자가 접근하도록 공개된 장소에 등록하며 개인 키는 사용자는 안전한 장소인 USB 메모리 스틱, 하드 디스크 등에 보관한다. 전자 서명 생성 시에는 비밀 정보인 개인 키를 이용하고 서명 검증 시에는 공개 키를 이용하여 누구나 검증할 수 있다. 공개 키는 임의 사용자가 수정하는 것을 방지하기 위해 인증서(Certificate)에 저장된다. 인증서는 사용자의 신원에 관한 정보도 함께 포함하며 인증 기관(Certificate Authority)에서 발급된다.

일반적으로 전자 서명을 생성하는 과정은 그림 4 (a)와 같다. 첫번째 단계로서 전자 계약서 등과 같은 메시지에 대해 해쉬 알고리즘을 적용하여 일정

크기의 해쉬 값을 생성한다. 해쉬 값의 크기는 MD5[9]의 경우 128 비트, SHA-1[10]은 160 비트이다. 두 번째 단계에서는 해쉬 값을 서명자의 개인키로 암호화하여 전자 서명 값을 생성한다. 그리고 원본 메시지, 해쉬 및 암호 알고리즘 식별자, 전자 서명 값, 서명자의 인증서 등을 인코딩하여 서명된 데이터(Signed Data)[11]를 생성한다. 서명된 데이터는 수신자에게 전송된다. 대표적인 서명 알고리즘으로는 RSA, DSA, ECDSA 등이 존재한다[8]. RSA 알고리즘은 유선 환경에서 널리 사용되고 있기 때문에 본 논문에서는 유무선 통합 환경의 호환성을 위해서 RSA를 서명 알고리즘으로 채택하였다.

수신자는 그림 4(b)와 같이 서명된 데이터로부터 전자 서명을 검증한다. 서명 데이터 내의 원본 메시지를 압축하여 새로운 해쉬 값을 계산한다. 한편 서명자의 인증서에 저장된 공개 키를 사용하여 전자 서명 값을 복호화하여 원래의 해쉬 값을 얻는다. 그리고 두개의 해쉬 값을 비교하여 일치하면 서명이 정확하다고 판단한다.

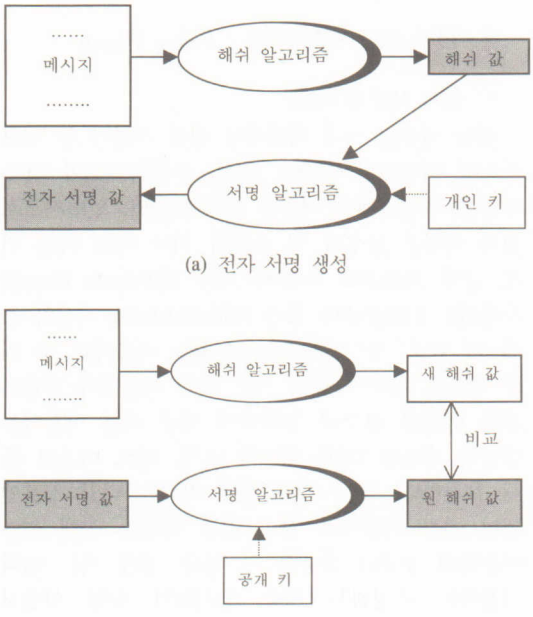


그림 4. 전자 서명 생성 및 검증 과정

2) RSA 알고리즘의 성능

RSA 알고리즘을 이용한 전자 서명 과정을 분석하여 전자 서명 생성 및 검증에 영향을 미치는 요소들을 분석한다. 서명자의 개인 키 값은 d로, 공개키는 (n, e)로 정의한다[8].

① 전자 서명 생성: 전자 서명을 생성하는 함수 SIGN(d, n, M)을 정의한다. SIGN 함수는 전자 계약서의 텍스트 M에 대해 전자 서명 값을 반환하는 함수이다. 전자 서명은 M에 대한 해쉬 값(h(M))을 계산한 후, 멱승 연산(exponential operation)과 모듈러 연산을 수행한다. 따라서 전자 서명 시간 분석 시 해쉬 연산과 d에 대한 멱승 연산을 고려해야 한다. 서명 값 S는 n에 대한 나머지이므로 n의 크기에 의존적이다.

$$P_{sign} : SIGN(d, n, M) = h(M)^d \bmod n = S$$

② 전자 서명 검증: 전자 서명을 검증하는 함수 VERIFY(n, e, M, S)를 정의한다. VERIFY 함수는 계약서 M과 전자 서명 값 S를 바탕으로 공개키를 구성하는 n, e 값을 이용하여 서명을 검증하는 함수이다. VERIFY 함수는 서명 값 S을 복호화한 원래의 해쉬 값(h1)과 M에 대한 해쉬 값(h2)을 비교하여 불린 값 {true, false}을 반환한다. SIGN 함수와 마찬가지로 해쉬 연산과 e에 대한 멱승 연산을 포함하고 있다.

$$P_{verify} : VERIFY(n, e, M, S)$$
$$S^e \bmod n = h_1,$$
$$h(M) = h_2$$
$$\text{compare } h_1 \text{ with } h_2$$

RSA 서명 생성 및 검증 시간에 영향을 주는 파라미터를 요약하면 다음과 같다.

① 전자계약서 M 의 크기 : 해쉬 알고리즘은 그림 3와 같이 메시지 블록에 대해 반복 압축 함수를 수행하기 때문에 메시지의 크기는 해쉬 값 생성 시간과 밀접한 연관을 가진다.

② 공개 키 n, e의 크기: 서명 검증 시에는 공개 키 e의 멱승 연산을 수행하기 때문에 e의 크기에 따라 서명 검증 시간이 다르게 나타난다. 그리고 공개 키 n의 크기에 따라 서명 값 S의 크기가 다르게 나타난다.

③ 개인 키 d의 크기: 서명 생성 시에는 개인 키 d

의 멱승 연산을 수행하기 때문에 개인 키 d의 크기에 따라 서명 생성 시간이 다르게 나타난다.

3. 단말기에서 네트워크 전송 성능: P_{up}

1) 서버로의 전송 시간

단말기에서 서명된 데이터를 전자 계약 서버에 전송하기 위해 HTTP와 하부 TCP/IP 프로토콜을 이용한다. 그림 5는 HTTP 1.1에서 TCP 연결 설정 후 문서를 전송하는 과정을 보여주고 있는데, 전체 전송 시간은 TCP 연결 설정 시간인 T_{setup}과 데이터 전송시간 T_{data}로 나눌 수 있다.

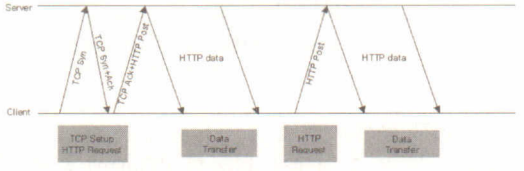


그림 5. HTTP 1.1에서 문서 전송 과정

HTTP 1.0과는 달리 HTTP 1.1에서는 서버에 대한 요청마다 연결 설정이 필요 없다. HTTP 1.1에서 T_{setup} 시간은 처음 HTTP 요청 시에만 TCP 연결을 설정하므로 다음과 같이 계산될 수 있다. 수식에서 RTT는 단말기와 서버간의 round-trip time을 의미한다.

$T_{setup} = 1.5 \times RTT$: 처음 요청 시
$T_{setup} = 0.5 \times RTT$: 두 번째 및 그 이후 요청 시 (수식 1)

한편 서명 데이터를 전송하는 T_{data} 시간은 크게 두 가지 요소로 구성된다. 즉, 모바일 단말기와 서버간에 첫 바이트가 도착하기까지의 시간인 전파 지연 시간 $T_{propagation}$ 과 순수 데이터 전송 시간 $T_{transfer}$ 로 나눌 수 있다. $T_{propagation}$ 은 "ping" 명령어를 통하여 알 수 있으며, $T_{transfer}$ 은 네트워크 대역폭과 TCP 알고리즘에 의해 영향을 받는다. Windows 운영체제에서 구현된 TCP 알고리즘은 다음과 같은 특징이 있다.

① Nagle 알고리즘 사용

Microsoft TCP 스택은 패킷 전송 시, Nagle 알고리즘을 사용한다. Nagle 알고리즘은 작은 양의 데이터가 네트워크로 전송되는 것을 방지하기 위해, ACK을 수신하기 이전에 전송 데이터를 통합하여 전송한다. 그러나 예외 사항으로 이전 패킷에 대한 ACK을 수신하기 전에, 전송할 데이터가 MTU(Maximum Transmission Unit)보다 클 경우에는 바로 전송한다.

② Delayed ACK

Microsoft TCP 스택은 패킷 수신시, ACK을 보내기 위해 200 ms 타이머를 구동한다. 그 이유는 수신자측에서 전송할 데이터가 있을 경우 ACK과 같이 전송하기 위해서이다. 그러나 200 ms 이내에 새로운 패킷을 수신했을 때에는 타이머 종료 전에 ACK을 전송한다.

만약 모바일 단말기에서 서명된 데이터를 연속적으로 전송할 경우, ACK을 수신하지 않더라도 연속적으로 MTU 단위로 전송하게 된다. 그리고 전자계약 서버에서는 첫 패킷을 수신한 후 200ms 이전에 새로운 패킷을 수신하게 된다면 타이머 종료 전에 ACK을 전송할 수 있게 된다. 결국 두 개의 패킷 단위로 ACK을 전송하게 된다. 실제로 CDMA 2000 1x의 최대속도는 144 kbps이므로 MTU 크기인 1500 바이트 전송 시 83 ms가 소요되기 때문에 200 ms보다 작다. 따라서 타이머 종료 전에 ACK을 전송한다.

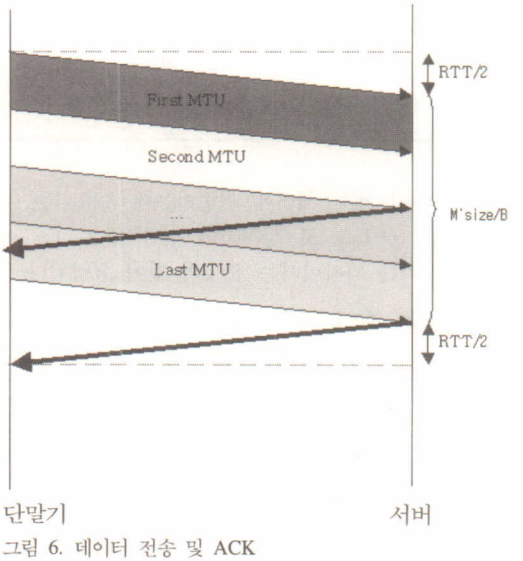


그림 6. 데이터 전송 및 ACK

그림 6은 모바일 단말기에서 서버로 전송하는 과정을 보여주고 있다. 모바일 단말기에서 서명된 데

이터를 MTU 단위로 연속적으로 보낼 경우 서버에서는 200 ms 이전에 새로운 패킷을 수신할 수 있다. 따라서 서버에서는 두 개의 패킷에 대한 ACK을 보내게 된다. 결국 데이터 전송 시간 T_{data} 는 다음과 같이 표현할 수 있다. 여기서 M'_{size} 는 서명된 데이터 M'의 크기를, B는 네트워크 대역폭을 의미한다.

$$T_{data} = T_{propagation} + T_{transfer} = RTT + M'_{size}/B \quad \text{(수식 2)}$$

2) 윈도우 크기

TCP는 흐름 제어를 위해 윈도우를 사용한다. 윈도우는 수신자로부터 ACK을 수신하기 전에 송신자가 전송할 수 있는 최대 데이터의 양으로서 T_{data}에 중요한 영향을 준다. 그림 6에서 송신자인 모바일 단말기에서 첫 번째 ACK을 수신하기까지의 시간은 RTT + (2 * MTU/B) 시간이 소요된다. 모바일 단말기에서 흐름 제어로 인한 대기 시간을 없애기 위해서는 이 시간동안 전송할 수 있는 데이터의 크기가 수신자의 윈도우 크기보다 커야 한다. 따라서 최대 성능을 얻기 위한 윈도우 크기 W는 다음과 같이 계산될 수 있다.

$$W \geq (RTT + 2 \times MTU/B) \times B = (RTT \times B) + (2 \times MTU) \quad \text{(수식 3)}$$

M'을 전자 계약 서버에 전송하는데 소요되는 시간은 $T_{setup} + T_{data}$ 로 정의되며 결론적으로 P_{up} 에 영향을 미치는 파라미터는 표 2와 같이 요약된다.

표 2. P_{up}에 영향을 미치는 파라미터

요소	성능 영향
M'_{size}	서명된 데이터의 크기로서 T_{data}에 영향
RTT	전파 지연 시간의 두 배의 크기
B	모바일 네트워크의 대역폭
MTU	한 번에 전송할 수 있는 최대 패킷 크기, 네트워크의 종류에 따라 다름
W	모바일 단말기와 서버의 윈도우 크기

Ⅳ. 실험 결과

본 절에서는 3 절에서 정의한 성능 요소들을 기반으로 모바일 단말기 상의 성능을 측정하였다. 실험 환경은 표 3과 같이 구성되었다.

표 3. 실험 환경

실험 요소	사양
모바일 단말기	Compaq iPAQ 3850(203MHz, 64MB, 320x240 화면)
전자계약 서버	Windows 2000, IIS, Tomcat, MS-SQL 서버
네트워크	CDMA 2000 1x, 최대 144Kb

1. 전자 문서 처리 성능 : P_{view}, P_{comp}

P_{view} 성능은 3 절에서 dSize와 dTime에 의해 영향을 받는 것으로 분석되었으며 본 절에서는 이 값들을 실측한 결과를 보인다. 이를 위해 한 화면에 출력될 규모의 전자 계약서들의 크기와 디스플레이 시간을 측정하였다. 본 논문에서는 하나의 PDA 화면에 출력될 문서는 표 4와 같은 컨트롤을 포함하도록 하였다. P_{comp} 성능은 P_{view} 와 유사하게 나타나서 자세한 설명은 생략하였다.

표 4. PDA 한 화면에 출력되는 윈도우 컨트롤

윈도우 컨트롤	개수
Text Box	9
Edit Box	5
Combo Box	2
Radio Button	1
Push Button	1

① dSize

320x240 화면에 필요한 XML 전자 계약서의 크기(dSize)를 측정해 본 결과, 평균적으로 3KB로 조사되었으며 이는 기존의 워드 문서에 비해 매우 작은 크기로 평가된다. XML 전자 문서를 사용함으로써 모바일 단말기의 메모리를 낭비를 줄이고 서버와의 전송 시간이 줄일 수 있는 장점을 확인할 수 있었다.

② dTime

한 화면에 출력될 XML 전자 계약서를 파싱하고 윈도우 컨트롤을 이용하여 화면에 출력하는데 걸리는 dTime은 약 245 ms가 소요되었다. 비록 윈도우 컨트롤의 수와 종류에 따라 약간의 차이는 존재하지만 대체로 200~300 ms가 소요되는 것으로 평가되었으며 이 시간은 사용자가 인내할 수 있는 지연 시간이라고 판단된다.

2. 전자 서명 처리 성능 : P_{sign}, P_{verify}

P_{sign} 과 P_{verify} 를 보기 위해 단말기에서 RSA 서명의 생성 및 검증 시간을 측정하였다. 실험 파라미터는 표 5와 같으며 512, 1024 비트의 공개 키 n을 가지는 두 개의 인증서를 사용하였으며 각 인증서에는 동일한 크기의 공개 키 e(0x010001)가 저장되어있다.

표 5. 전자 서명 모듈의 실험 파라미터

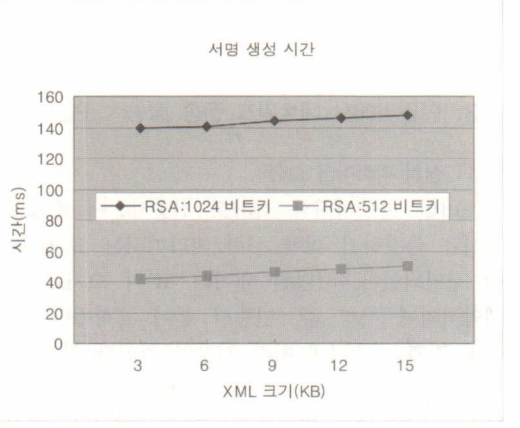
실험 파라미터	설정
해쉬 알고리즘	SHA-1
서명 알고리즘	RSA
공개키 n	512, 1024 비트
공개키 e	17 비트
개인키 d	512, 1024 비트
전자 계약서 크기 M	3, 6, 9, 12, 15 KB

1) 전자 서명 생성 시간, SIGN(d, n, M)

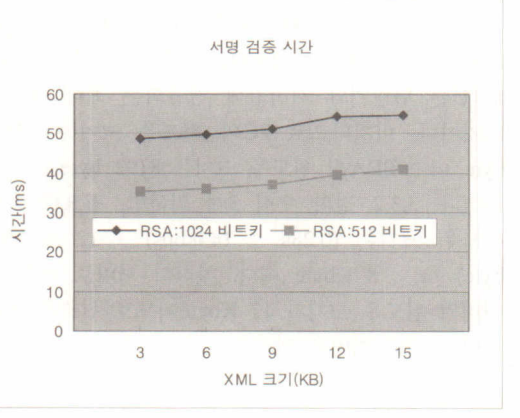
그림 7(a)는 모바일 단말기에서 전자 서명을 생성하는데 걸리는 시간을 보여주고 있다. 실험 결과는 다음과 같다.

① M의 크기에 따른 서명 생성 시간: 문서의 크기가 증가할수록 서명 생성 시간이 증가하는 것을 볼 수 있다. 서명 생성 시간은 해쉬 단계와 서명 단계로 나눌 수 있는데 해쉬 단계가 문서의 크기에 따라 시간이 증가하기 때문이다. 즉, 해쉬 알고리즘의 특성상 문서의 크기가 증가할수록 압축 함수를 수행하는 횟수가 증가하기 때문이다. 반면 서명 단계는 일정한 크기의 해쉬 값에 따라 수행되므로 실제 값에 따라 다소 차이는 있지만 대체로 일정하다고 볼 수 있다. 따라서 문서의 크기가 증가할수록 완만하게 증가함을 볼 수 있다.

② 키 n, d의 크기에 따른 서명 생성 시간: n, d의 크기가 512 비트인 경우에는 약 42 ~ 50 ms 가 소요되며 키의 크기가 1024 비트인 경우에는 140 ms ~ 147 ms가 소요되는 것을 확인할 수 있다. 일반적으로 RSA 서명 생성은 개인 키 d의 멱승을 포함하는데 개인 키가 클수록 서명 시간이 증가하게 된다. 결국, 키의 크기는 서명 생성 시간에 많은 영향을 끼침을 알 수 있다.



(a) M의 크기에 따른 서명 생성 시간



(b) M의 크기에 따른 서명 검증시간

그림 7. 전자 서명 생성 및 검증 시간

2) 전자 서명 검증 시간, VERIFY(n, e, M, S)

그림 7(b)는 모바일 단말기에서 전자 서명을 검증하는데 걸리는 시간을 보여주고 있다. 실험 결과는 다음과 같다.

① M의 크기에 따른 서명 검증 시간 : 문서의 크기가 증가할수록 서명 검증 시간이 증가하는 것을 볼 수 있다. 그 이유는 서명 생성과 마찬가지로 해

쉬 값 생성 시간이 증가하기 때문이다.

② 공개 키 n 의 크기에 따른 서명 검증 시간: 공개 키 n이 512 비트 키의 경우 약 35 ~ 40 ms 가 소요되며 1024 비트 키인 경우에는 48 ms ~ 54 ms가 소요되었다. 전자 서명 검증 시에는 공개 키 e에 대한 멱승을 포함하고 있다. 일반적으로 공개 키 e는 개인 키 d 보다 작은 비트를 사용하기 때문에 서명 검증 시간은 공개 키 n의 크기에 덜 의존적임을 알 수 있다.

3. 단말기에서 네트워크 전송 성능 : P_{up}

1) 실험 파라미터 설정

모바일 단말기에서 서버로 전송하는데 소요되는 시간을 측정하기 위한 실험 파라미터는 표 6과 같이 설정되었다. 서명된 데이터 M'의 크기는 원본 계약서외에 서명 값, 인증서 등이 포함된다. 실제 측정 결과 인증서에 따라 약간의 차이가 있지만 대체로 약 1K 바이트 정도였다. 따라서 M'의 크기는 M의 크기에 1K 바이트가 증가된 양이다. 그리고 RTT는 Ping 명령어를 사용하여 측정된 값의 최소 값으로서 280 ms가 측정되었다.

한편, CDMA 2000 1x의 최대 대역폭은 144 Kbps이며, MTU는 1500 바이트가 사용되었다. MTU는 단말기와 서버간에 전송되는 최대 패킷의 크기이다. 마지막으로 PC의 윈도우 크기는 17520 byte 이고 PDA의 윈도우 크기는 8192 byte로 설정하였다. 3.3.2 절의 수식 3에 의하면, 필요한 최소 윈도우 크기는 (0.28s * 144kbps) + (2 * 1500 byte) ≅ 8 Kbyte 이다. 따라서 서버로 전송 시 서버의 윈도우 크기가 17 Kbyte이므로 윈도우 흐름 제어로 인한 지연은 없다.

표 6. 네트워크 전송 실험 파라미터

실험 파라미터	설정
M'_{size}	4, 7, 10, 13, 16 KB
RTT	280 ms
최대 대역폭 B_{max}	144 Kbps
MTU	1500 바이트
PDA 윈도우 크기 W_{PDA}	8192 바이트
서버 윈도우 크기 W_{server}	17520 바이트

2) 전송 시간

그림 8은 M'_{size} 의 크기에 따른 전송 시간을 보여주고 있다. 본 실험치는 20번의 반복 테스트를 수행하여 얻은 값이다. 그림 8(a)는 최소 전송 시간을, 8 (b)는 평균전송 시간을 보여주고 있다.

① 최소 전송 시간

그림 8(a)는 이론적인 전송시간과 최소 전송 시간을 보여주고 있는데, 이론적인 전송 시간은 수식 1, 2를 이용하여 계산된 값이다. 예를 들어 서명 데이터의 크기 M'_{size} 가 4KB라고 가정하면,

T_{setup} = 0.5 RTT = 0.14 sec

T_{data} = RTT + M'_{size} / B_{max} = 0.28 + (4K * 8)/144Kbps = 0.507 sec

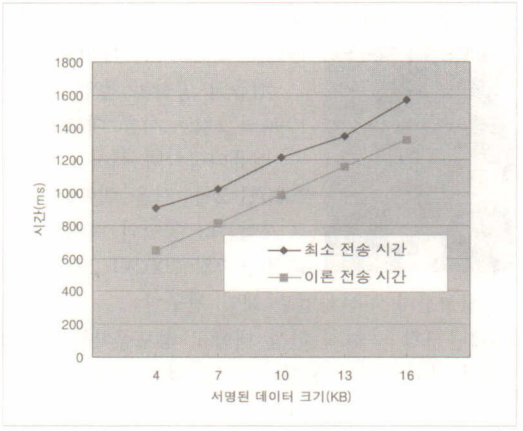
이론적 전송시간 = T_{setup} + T_{data} = 0.647 sec

M'size 가 3KB씩 증가함에 따라 T_{data} 값은 (3K * 8bit)/144 kbps = 170 ms 씩 증가한다. 최소 전송 시간은 이론적인 전송 시간에 비해 대체로 200 ms 차이가 발생하며 M'_{size} 가 클수록 증가한다. 일반적으로 모바일 네트워크의 대역폭은 유선 LAN 대역폭보다 매우 낮기 때문에 T_{data} 는 전파 지연 시간 $T_{propagation}$ 보다는 순수 데이터 전송 시간 $T_{transfer}$ 에 더욱 의존적이라고 할 수 있다. 따라서 M'_{size} 의 크기와 밀접한 관련이 있다.

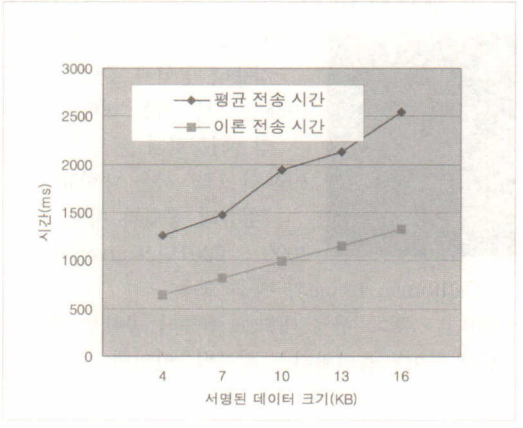
② 평균 전송 시간

그림 8(b)는 평균적인 전송 시간을 나타내고 있는데, M'_{size} 가 클수록 증가한다. 그러나 실험 결과는 평균적인 전송 시간이 이론적인 전송 시간에 비해 약 2배가 됨을 보여 준다. 그 이유는 다음과 같다. 첫째, CDMA 2000 1x의 최대 속도는 144 Kbps이지만 실제 속도는 그 이하이다. 일반적으로 정지 혹은 도보 이동 중에서는 80~100 Kbps의 전송 속도 밖에 되지 않는다[12]. 둘째, 데이터 전송 도중 라우터 및 서버에서의 지연 시간 때문이다.

결론적으로 모바일 네트워크의 대역폭은 유선 LAN 대역폭보다 매우 낮기 때문에 문서의 크기 M을 작게 유지함으로써 전송 시간을 줄이는 것이 무엇보다 중요하다고 할 수 있다.



(a) 최소 전송 시간



(b) 평균적인 전송 시간

그림 8. 모바일 단말기에서 서버로의 전송 지연 시간

V. 결 론

M-Commerce는 모바일 환경의 이동성과 편리성으로 인해 그 사용이 활성화되고 있지만 유선 환경에 비해 모바일 단말기의 하드웨어가 열악하며 네트워크가 취약하다는 단점을 가지고 있다. M-Commerce 응용의 많은 경우, 상거래의 안전을 확보하기 위해 전자 서명을 도입하고 있기 때문에 전자 서명에 따르는 성능 부하를 감내하여야 한다. 이처럼 모바일 전자 서명 시스템은 성능에 매우 민감한 요인을 내재하고 있으므로 성능에 영향을 미치는 요소와 그 정도에 대한 체계적인 분석이 매우 필요하다.

본 논문은 현재의 상용 모바일 단말기를 대상으로 모바일 전자 서명 시스템의 성능에 영향을 미치는 요소를 이론적으로 분석하고 실측을 통해 전자 문서의 크기, 전자 서명 및 검증, 네트워크 전송 등에 따른 정성적, 정량적인 성능 평가를 수행하였다. Compaq iPAQ 3850을 단말기로 하고 펜티엄 IV급 PC를 전자 계약 서버로 하는 전자 계약 시스템을 테스트 시스템으로 구성하고 XML을 기반으로 하는 전자 계약서를 정의하여 사용하였으며, 단말기에 탑재된 Mobile Explorer 상에 실행될 ActiveX 코드를 작성하여 실험을 수행하였다.

본 논문의 실험 결과는 다음과 같이 요약된다. 첫째, XML로 작성된 전자 문서는 크기가 작아 데이터 전송 시간을 줄이는데 기여하였으며, XML 문서의 파싱을 비롯하여 전자 계약서가 한 화면에 출력되는데 소요된 시간이 약 200 ms 로서 고객이 충분히 감내할 수 있는 시간이다. 둘째, 전자 서명 시간은 전자 문서의 크기와 키의 크기에 비례하며 키의 크기에 더욱 의존적이다. RSA 전자 서명 생성 알고리즘은 개인 키 d의 멱승 연산을 수행하기 때문에 개인 키의 크기에 많은 영향을 받는다. 유선 환경에서 많이 사용되는 1024 비트 RSA 전자 서명이 iPAQ 3850에서 약 150 ms 소요되기 때문에 모바일 환경에 큰 무리가 없는 것으로 평가되었다. 한편, 전자 서명 검증 시간은 공개 키 e의 크기가 작기 때문에 서명 생성 시간에 비해 작은 것으로 평가되었다. 셋째, 서버로의 전자 문서 전송 시간은 모바일 네트워크의 낮은 대역폭으로 인해 문서의 크기에 매우 의존적이었다. 현재 상용화되고 있는 CDMA 2000 1x에서 10 KB 서명된 데이터의 평균 전송 시간은 약 2000 ms로서 매우 크다고 할 수 있다. 따라서 전자 서명 시간에 비해 전송 시간이 모바일 환경의 성능에 매우 의존적임을 알 수 있다.

현재 상용화되고 있는 단말기 및 네트워크를 바탕으로 본 논문에서 실측된 결과들은 M-Commerce 시스템 연구에 중요한 데이터를 제공할 수 있다고 평가된다. 향후에는 ECDSA와 같은 전자 서명 알고리즘, OCSP, CRL을 이용한 인증서 검증 알고리즘을 분석할 예정이며, 시스템 병목 현상을 해결하기 위한 효율적인 방안들을 제시할 예정이다.

참 고 문 헌

[1] U. Varshney and R. Vetter, "A Framework for the Emerging Mobile Commerce Applications," *Proceedings of the 34th Hawaii International Conference on System Sciences*, 2001.

[2] James A. Senn, "The Emergence of M-Commerce," *IEEE Computer*, December, 2000.

[3] M. Heijden and M. Taylor, *Understanding WAP: Wireless Application, Devices, and Services*, Artech House, 2000.

[4] M. Myers, R. Ankney, etc., Internet X.509 Public Key Infrastructure: Online Certificate Status Protocol-OCSP, RFC 2560, IETF, 1999.

[5] A. Jurisic and A. J. Menezes, "Elliptic Curves and Cryptography," *Dobb's Journal*, April, 1997.

[6] W3C, Extensible Markup Language (XML) 1.0, Technical Report, WWW Consortium, http://www.w3.org/TR/1998/REC-xml-19980210, 1998.

[7] W3C, Document Object Model, WWW Consortium, http://www.w3.org/DOM, 2002.

[8] S. Burnett and S. Paine, *RSA Security's Official Guide to Cryptography*, RSA Press, 2001.

[9] R. Rivest, The MD5 Message-Digest Algorithm, RFC 1321, 1992.

[10] FIPS 180-1, "Secure Hash Standards," Federal Information Processing Standards Publication, U.S. Department of Commerce/NIST, 1995.

[11] R. Housley, Cryptography Message Syntax, RFC 2630, Internet Society, 1999.

[12] 디지털 타임즈, "cdma 2000-1x 속도," http://www.dt.co.kr/dt_srcview.html?gisaid=2001040202010663587007, 2001.

김 남 윤 (Namyun Kim) 정회원

1992년 2월: 서울 대학교 컴퓨터 공학과 졸업
1994년 2월: 서울 대학교 컴퓨터공학과 석사
2000년 2월: 서울 대학교 컴퓨터 공학과 박사
1999년 9월~2002년 2월 삼성전자 무선사업부 책임 연구원
2002년 3월 ~ 현재 한성 대학교 정보공학부 조교수
<관심 분야> 이동 통신 시스템, 정보 보안, 실시간 시스템 등

황 기 태(Kitae Hwang) 정회원

1986년 2월: 서울 대학교 컴퓨터 공학과 졸업
1988년 2월: 서울 대학교 컴퓨터공학과 석사
1994년 2월: 서울 대학교 컴퓨터 공학과 박사
2000 ~ 2001년 University of California, Irvine의 방문 교수
1994년 ~ 현재 한성 대학교 컴퓨터 공학부 부교수
<관심 분야> 유비쿼터스 컴퓨팅, 인터넷 시스템, 모바일 보안 등