WIPI 환경에서 XML 문서 암호화 시스템

홍현우*·이재승** · 이성현** · 정회경*

XML Encryption System on WIPI Environment

Hyeon-Woo Hong* · Jae-Seung Lee** · Seoung-Hyeon** · Hoe-Kyung Jung*

요 약

현재 국내에서는 이동통신사마다 별도 무선인터넷 플랫폼을 사용하고 있어 콘텐츠 제공업체와 단말기 제조사가 개발시간이나 비용이 많이 소요되고 있다. 이에 국내 이동통신 3사와 전자통신연구원 등을 주체로 무선인터넷 표준 플랫폼인 WIPI(Wireless Internet Platform for Interoperability)를 제정하고 표준화작업을 국내외로 추진하고 있다. 그러나 국내환경에서도 WIPI는 표준화 작업이 진행 중이며 WIPI 환경을 기반으로 한 콘텐츠가 미비한 현실이다. 특히 무선인터넷 플랫폼이 하나로 통합되면서 기존의 여러 플랫폼이 공존하던 환경에 비해 해킹이나 악성코드의 집중적인 공격이 예상되며 이를 대비해 모바일 환경에서 데이터를 효과적으로 보호할 필요성이 대두하였다.

이에 본 논문에서는 모바일 환경에서 XML 문서로 데이터를 교환하는 상황을 고려하여 PC 환경에서 사용되던 여러 암호화 기법을 적용하여 WIPI 환경에서 XML 문서 암호화 시스템을 설계 및 구현하였다.

ABSTRACT

Recently, The biggest three mobile telecommunication companies of our country still using independence wireless internet platform. And, It carry so many difficulties to the phone company and content provider company. Such as the timing of the development or the fee of the development. Because even they develop one product and they must make it prepare for some platform of every mobile telecommunication companies. And this make the development more longer and more expensive. For this reason, SKT, LG telecom and KTF develop the new wireless internet platform named WIPI with ETRI. and the working is still go on and go ahead with propulsion. And if it come to reality, the WIPI will attached from much of attack such as hacking or virus. But some data exchange between mobile phone is so important as to flow.

Thus, in this paper, we consideration the XML using in the wireless environment and we are design and implementation the XML encryption system working at the WIPI in order to protect the data, we want to protect.

키워드

WIPI, XML 문서, 모바일, 암호화

Ⅰ. 서 론

최근 무선 인터넷 기술과 단말기 제조기술의 비약적인 발전으로 현재 휴대용 단말은 기존의 단일한 이동전화에서 MP3, 영화, 동영상, 디지털 방송 수신 등 여러 가지 디지털 콘텐츠를 소비하고 모바일 서비스도 와이브로, 금융결제, 금융조회 등 영역으로 용도가 확장되어 개인정보처리 중심으로 그 역할을 발휘하고 있다.

하지만 아직까지도 국내의 휴대용 단말에서 사용하고 있는 무선 인터넷 플랫폼은 통합되지 않고 있다. 이러

* 배재대학교 컴퓨터공학과(교신저자 : 정회경)
** 한국전자통신연구원 정보보호연구단

접수일자 : 2007. 6. 11

한 현실은 단말기 제조업체들로 하여금 더욱 긴 개발시간을 소요하게 할 뿐만 아니라 디지털 콘텐츠를 제공하는 업체들은 동일한 콘텐츠를 여러 플랫폼에 접목시켜야 하기 때문에 개발시간이나 개발경비가 많이 소모되고 있으며 국외의 플랫폼을 사용하는 경우에는 상당한 액수의 로열티를 지불해야 하기 때문에 국내 업체의 부담을 증가하고 있다.

이러한 현실을 배경으로 무선인터넷 표준 플랫폼인 WIPI가 TTA 단체 표준인 모바일 표준 플랫폼 규격으로 채택되었으며 WIPI 3.0이 출시 될 예정이다[1].

휴대용 단말기에서 데이터 교환은 현재 XML 문서를 교환하는 방식으로 사용되고 있다. 하지만 XML 문서는 유통될 때 모든 데이터가 노출되기 때문에 XML 문서 암호화의 필요성이 제기되었으며, 현재 W3C에서 XML 문서의 암호화에 대한 표준화 작업을 진행하고 있다.

이에 본 논문에서는 모바일 환경에서 WIPI 플랫폼을 기반으로 현재 PC 환경에서 사용되고 있는 여러 암호화 기법을 적용하여, WIPI 환경에 맞는 XML 문서 암호화 시스템을 설계 및 구현하였다.

Ⅱ. 관련연구

2.1 WIPI

WIPI는 각 이동통신사의 서비스 요구와 상호호환성을 고려하고 디지털 콘텐츠의 보안기능을 강화하여 플랫폼의 안정성을 고려하였다. WIPI는 단말기 OS나 Air Interface에 독립적이며 기존의 WAP나 J2ME와도 호환된다. 그리고 콘텐츠 개발을 지원하는 API를 지원하고 있으며 기존 플랫폼의 단점을 보완하고 차세대 서비스 기술을 반영하고 있다.

WIPI의 구성도인 그림 1을 살펴보면 주요하게 HAL, Basic API, APP Manager, OEM SPEC Extended API , 기타 서비스 등으로 구성되었다.

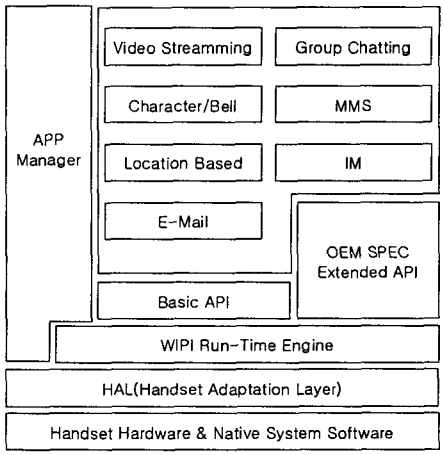

그림 1. WIPI 구성도
Fig. 1. WIPI Architecture

1) HAL(Handset Adaptation Layer)은 하드웨어에 대한 추상화를 실현하여 하드웨어 독립성을 지원하는 계층이다.

2) Basic API는 Java 및 C언어로 구성되어 있는 응용프로그램 개발을 지원하고 있는 API 들이며, 기능면에서 동등한 API를 제공한다.

3) App Manager는 응용프로그램의 다운로드, 설치, 삭제를 관리한다.

4) OEM SPEC Extended API는 App Manager을 통하여 추가/갱신된 API 및 컴포넌트를 저장 및 관리한다.

5) 기타 서비스로는 위의 HAL, Basic API, App Manager, OEM SPEC Extended API 등에 기초하여 실행되고 있는 문자메시지, 비디오 스트리밍 등 서비스를 포함한다[1,2,3].

2.2 XML 문서 암호화

시스템사이 데이터 교환이 기존 텍스트 문서로부터 현재의 XML 문서로 변화하면서 텍스트 문서에 사용되던 암호화방식이 XML 문서 암호화에 적합하지 않게 되었다. XML 문서는 구조적인 특징을 가지고 있기 때문에 기존 방식대로 전체 문서를 암호화하면 XML 문서의 구조적인 특징이 파괴된다. 따라서 W3C는 XML 문서 암호화에 대한 권고안을 발표하였다. 국내에서는 이 권고안을 보완하여 일부 내용을 추가하여 표준으로 채택하였다[4].

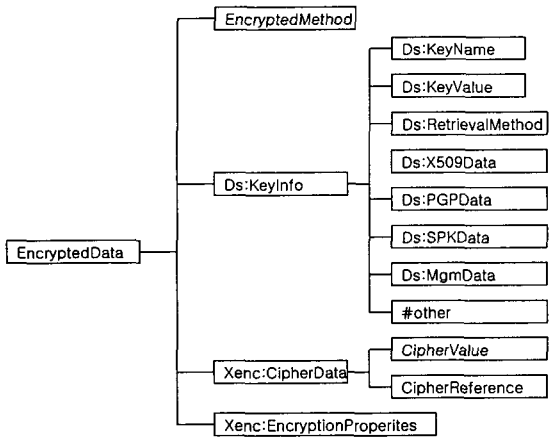

그림 2. XML 문서 암호화 스키마 구조
Fig. 2. XML Encryption schema

XML 문서 암호화는 그림 2의 스키마 구조로이루어진다[4].

1) EncryptionMethod 엘리먼트

데이터를 암호화할 때 사용한 암호화 알고리즘을 기술하는 선택적인 엘리먼트이다. 만약 EncryptionMethod 엘리먼트가 없다면 암호 알고리즘은 수신자에게 별도로 알려져야 하거나 아니면 복호화가 실패하게 된다.

2) CipherData 엘리먼트

암호화된 데이터를 제공하는 필수 엘리먼트이다. CipherValue 엘리먼트의 base64로 인코딩된 텍스트처럼 암호화된 octet 시퀀스나 CipherReference 엘리먼트를 통해 암호화된 octet 시퀀스를 포함한 외부 위치의 참조값을 제공해야 한다.

3) CipherValue 엘리먼트

선택적인 엘리먼트로 실제로 암호화된 데이터값을 base64로 인코딩하여 가진다.

4) CipherReference 엘리먼트

암호화한 엘리먼트로 만약 CipherValue가 제공되지 않는다면 CipherReference는 암호화된 데이터 위치를 참조하고 있다.

5) EncryptedData 엘리먼트

암호화 구문에서 핵심 엘리먼트로 암호화된 데이터를 포함하는 자식 CipherData 뿐만 아니라 암호화된 엘리먼트를 대체할 수 있으며 XML 문서 전체가 암호화될 때는 새로운 문서 루트(root)처럼 사용된다.

6) ReferenceList 엘리먼트

EncryptedData나 EncryptedKey elements에 의해서 암호화된 데이터까지의 포인터를 포함하는 엘리먼트이다. XML 문서 암호화는 원 문서의 구조적 형식을 유지하며, XML 문서내 임의 데이터, 임의 부분에 대해 암호화가 가능하다.

2.3 데이터 암호화

현재 보안 분야에서 사용하고 있는 데이터 암호화 알고리즘은 처리되는 데이터 블록의 크기에 따라 블록 암호화와 스트링 암호화 방식으로 구분할 수 있고, 키의 유형에 따라 공개키 방식과 대칭키(비밀키) 방식으로 구분할 수 있다.

스트링 암호화는 일반적으로 스트링을 입력받아 비트단위로 암호화를 진행하는 방식이고 블록 암호화는 일정한 크기의 고정된 블록으로 데이터를 입력받아 암호화를 진행하는 방식이다. 스트링 암호화는 블록 암호화에 비해 실행속도가 빠른 특징이 있지만 일반적으로 블록 암호화의 안전성이 스트링 암호화에 비해 강력하다.

대칭키 암호화는 암 · 복호화 할 때 같은 키를 사용하며 실행속도가 스트링 암호화에 비할 때 좀 느리다. 특히 스트링 암호화와 대칭 키 암호화는 암 · 복호화에 사용하는 키가 동일하기 때문에 송신자와 발신자사이에서 암호화키를 안전하게 전송해야 하고 키 관리가 복잡한 단점이 있다. 공개 키는 이러한 단점을 극복하기 위하여 개발되었으며 암 · 복호화 과정에 사용하는 키가 다르기 때문에 키 관리가 상대적으로 용이한 이점이 있다.

Ⅲ. 시스템 설계

3.1 암호화 스키마 설계

그림 3은 W3C에서 제안한 XML 문서 암호화 스키마를 참조하여 본 본문에서 설계한 XML 문서 암호화의 스키마 구조이다. 이 구조에서 EncryptedData 엘리먼트는 핵심적인 요소로서 XML 문서내의 암호화되는 엘리먼트 또는 엘리먼트 데이터를 대체한다. 암호화된 데이터는 CipherValue 엘리먼트내에 직접 포함되거나 또는 CipherReference 엘리먼트로 위치가 표기된다. 그리고 EncryptionMethod 엘리먼트에 암호화에 사용한 알고리즘의 내용을 명시한다.

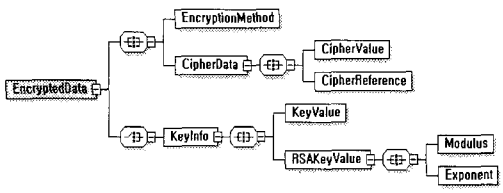

그림 3. XML 암호화를 위한 스키마
Fig. 3. XML Encryption schema

암호화 키 부분은 선택사항으로 키를 포함하거나 포함하지 않을 수 있다. 대칭키를 포함하는 경우에는 대칭키를 RSA 알고리즘으로 암호화하여 KeyValue 엘리먼트에 기술한다. 만약 RSA 공개키 쌍을 기술해야 할 경우에는 KeyValue 엘리먼트의 자식엘리먼트인 Modulus와 Exponent에 배포한다[5].

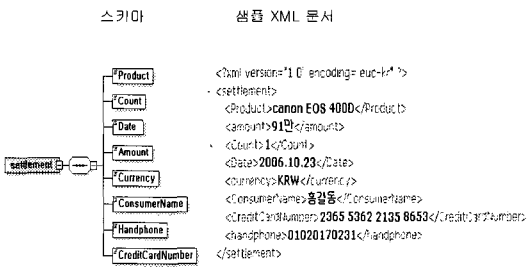

그림 4. 스키마와 샘플 XML
Fig. 4. Schema and Sample XML

슈퍼 암호화 또는 다중 암호화를 진행할 경우에는 EncryptedData에 스트링형의 ID 식별자를 추가해야 한다.

그림 4는 암호화할 XML 문서 스키마와 샘플 XML 문서이다. 이 XML 문서는 상품명, 상품가격, 상품수량 등의 상품정보와 신용카드번호, 주민등록번호, 구매자 명함, 구매자 휴대폰번호 등의 결재정보를 포함하고 있으며 암호화시 상품정보와 결재정보를 각각 암호화 한다.

3.2 암호화 모듈

그림 5에서는 본 논문에서의 암호화 구성도를 보여주고 있으며, 그림 6에서는 시스템의 암호화를 진행하는 시퀀스 다이어그램이다.

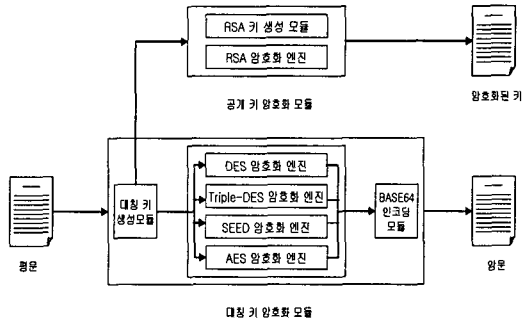

그림 5. 암호화 구성도
Fig. 5 Encryption Organization Chart

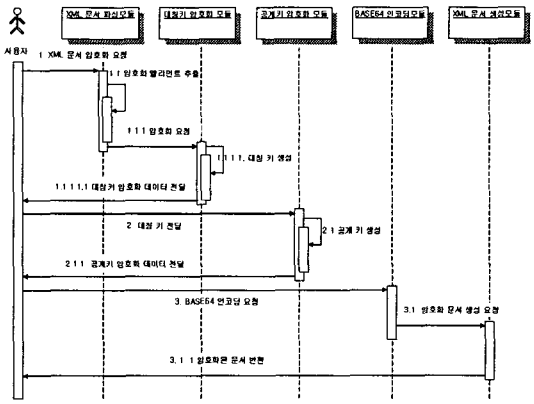

그림 6. 암호화 시퀀스다이어그램
Fig. 6. Encryption Sequence Diagram

암호화 되는 문서는 먼저 XML 문서파싱 모듈에서 파싱되어 문서내의 각 엘리먼트와 포함한 데이터를 추출한다. 사용자는 이 데이터들 중에서 암호화하려는 데이터를 선택하여 대칭키 암호화 모듈로 이 데이터들을 전달한다.

대칭키 암호화 모듈에서는 암호화하려는 데이터를 수신한 다음 먼저 암호화에 사용될 알고리즘을 선택하고 다시 해당하는 대칭키를 생성하여 암호화를 진행한다.

대칭키 암호화가 진행된 다음에 사용한 대칭키는 공개키 암호화 모듈로 전송된다. 공개키 암호화 모듈에서는 공개키를 생성하여 수신 받은 대칭키를 RSA 알고리즘으로 암호화한다.

암호화과정에서 생성된 바이너리 데이터는 BASE64 인코딩모듈에서 XML 문서에 표현이 가능하게 인코딩 되어 암호화문서를 생성한다.

3.3 복호화 모듈

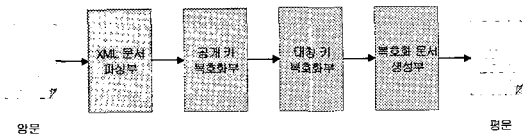

그림 7. 복호화 모듈 구성도
Fig. 7. Decryption Model

복호화 모듈 구성도와 복호화 모듈의 시퀀스다이어그램은 각 그림7과 8에서 보인다.

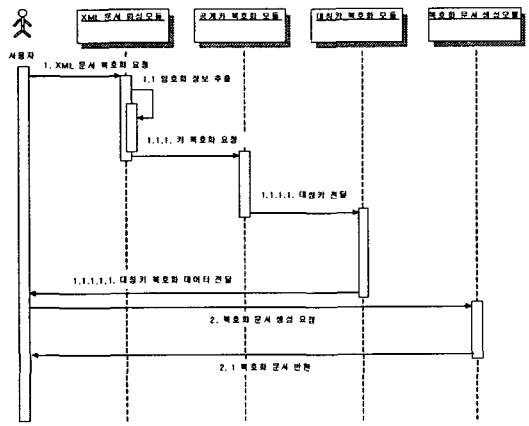

그림 8. 복호화 시퀀스다이어그램
Fig. 8. Decryption Sequence Diagram

먼저 복호화되는 문서는 파싱모듈로 전송되어 EncryptedData 엘리먼트로부터 암호화 정보를 추출한다. 공개키 복호화 모듈에서는 복호화를 진행하여 KeyInfo 엘리먼트로부터 암호화에 사용된 대칭키를 추출한다. 이 대칭키는 다시 대칭키 복호화 모듈로 전송되어 대칭키 복호화에 사용된다. 공개키와 대칭키 복호화를 진행한 뒤 데이터들은 복호화 문서 생성모듈로 전송되어 복호화된 XML 문서를 생성한다.

3.4 암호화 라이브러리

본 시스템의 실행 환경이 모바일이라는 제한적인 환경임을 고려하여 본 시스템에서 사용한 알고리즘 정보는 표 1과 같다.

표 1. 사용한 암호화 알고리즘
Table. 1. Encryption algorithm

	블록의 크기	키 크기	운영 모드
DES	64 비트	64 비트	CBC
Triple-DES	64 비트	128 비트	CBC
AES	128 비트	128 비트	CBC
SEED	128 비트	128 비트	CBC
RSA	DES/Triple-DES/AES/SEED의 키	512/1024/2048 비트	-

표 1과 같이 본 논문에서는 DES, Triple-DES, AES, SEED에 대하여 모두 PC환경에서 사용한 경우에 비해 최소의 블록길이와 키 길이로 설계하였다. RSA알고리즘에 대해서는 512비트부터 2048비트까지의 키를 사용 가능하며 본 시스템에서 RSA 키를 생성하거나 다른 시스템에서 생성한 RSA 키를 사용할 수 있다[5-9].

대칭 키 알고리즘의 운영모드 선택에서는 모바일 환경에서 전송되는 데이터의 크기가 한정되어 단일의 전송목적으로만 사용 될 것을 고려하여 모두 CBC 운영방식으로 설계되어 있다.

Ⅳ. 시스템 구현 및 고찰

4.1 시스템 구현

WIPI 환경에서의 데이터 암호화 시스템은 IBM-PC 호환 컴퓨터의 Windows XP SP2 운영체제에서 개발되었다. 개발환경은 SKT IDE이고 테스트 환경은 SKT 에뮬레이터이다. WIPI 환경의 특징상 실행속도를 고려하여 C 언어를 사용하고 유저 인터페이스는 WIPI C API로 구축하였다. 실제로 시스템을 테스트하기 위하여 SKT 이동통신사의 SCH-W210 폰에서 테스트를 진행하였다. 본 시스템을 사용하여 모바일 환경에서 데이터를 암호화하는 시나리오는 다음과 같다.

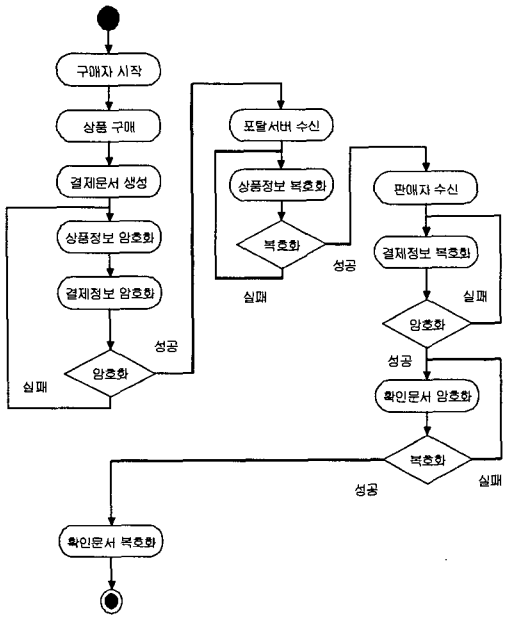

그림 9. 시스템 시나리오
Fig. 9. System Scenario

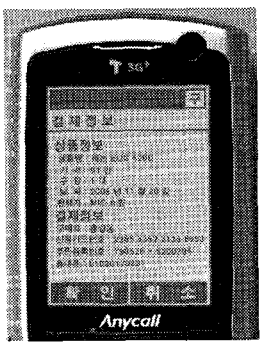

그림 10a. 결제 정보
Fig. 10a information

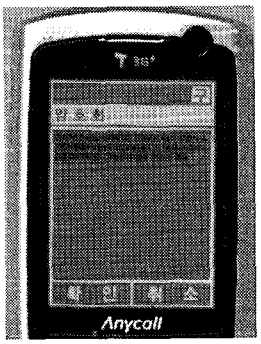

그림 10b. 암호화
Fig. 10b Encryption

그림 10a는 상품 구매자가 상품을 결제한 결제정보를 확인하고 있는 장면을 보여주고, 그림 10b는 결제 정보에 대해 AES 알고리즘으로 암호화 된 결제문서를 보여준다. 그림에서 암호화 된 바이너리 데이터가 다시 BASE64 인코딩으로 변환되어 사용자에게 보임을 확인할 수 있다.

4.2 시스템 고찰

본 시스템의 특징은 현재까지 PC 환경에서 많이 사용되는 대칭 키 암호화 알고리즘과 공개키 암호화 알고리즘을 적용하여 WIPI 환경에서 데이터의 보안을 강화한데 있다.

본 논문의 구현에서 보여준 바와 같이 포탈서버 측과 판매자 측은 모두 동일한 XML 문서를 수신하였으나 자신의 공개키로 암호화 된 데이터만 개인키로 복호화 가능할 뿐 문서내의 다른 암호화 된 정보에 대해서는 접근할 수 없다.

기존의 텍스트 암호화는 전체 데이터를 암호화하기 때문에 사용자마다 각각 다른 데이터를 전송해야 하는 불편이 있다. 그러나 본 논문에서는 XML 문서를 다중 암호화하여 유무선 통신망으로 수신자들에게 전송될 때 각 수신자들은 자신의 개인키로 해당하는 데이터만 복호화하여 이용이 가능하며 이런 특징은 상품유통 사슬이 복잡한 현재의 비즈니스 망에서 광범위하게 응용될 수 있다.

그러나 RSA 알고리즘 같은 경우는 보안성이 높고, 대칭키 암호화에 비해 키 관리가 용이하지만 실행속도가 느리다. 실제 폰 상에서 테스트 한 결과 DES, Triple-DES, AES, SEED 등 대칭 키 알고리즘은 소모되는 시간이 실제 응용이 가능한 범위에 속하며 RSA 알고리즘, 특히 RSA 알고리즘에서도 키의 생성과정은 512 키를 사용함에도 불구하고 비교적 긴 시간이 소모된다.

표 2는 본 시스템에서 암호화키에 RSA 알고리즘을 적용할 때 소모되는 시간을 20회 측정한 평균시간이다.

현재 PC 환경에서는 512 비트의 RSA 키는 보안성이 떨어져 실제 응용에서는 1024 비트 이상의 키를 사용할 것을 제안하고 있다. 표 2를 보면 생성하는 RSA 알고리즘의 키 길이가 2048 비트일 때 시스템에서의 시간소모가 너무 많다는 단점이 있다. 현재 시스템의 실행환경, 실행속도 및 보안성을 모두 고려할 때 모바일 같은 자원이 제한적인 환경에서 RSA 키 생성과 같은 시간소모가 많은 부분을 폰에서 실행하면 효율적이지 않기 때문에 폰 상에서는 RSA 키를 저장하고 있거나 또는 서버로부터 수신하는 방식으로 사용할 것을 추천한다.

표 2. RSA 알고리즘의 실행속도
Table. 2. RSA's speed

	DES	Triple-DES	AES	SEED
RSA 512 비트	2 초	4 초	3 초	3 초
RSA 1024 비트	4 초	6 초	5 초	5 초
RSA 2048 비트	17 초	19 초	18 초	18 초

현재까지 모바일 환경을 목적으로 한 XML 문서 암호화 시스템은 JAVA 언어로 개발된 경우가 대부분이다. 이러한 시스템의 경우에는 확장이 용이한 특징이 있지만 시스템의 성능이 본 논문에서 제안한 시스템에 비교할 때 실행속도에서 성능이 떨어지게 된다. 아래 표는 본 논문에서 설계한 시스템과 JAVA로 구축된 시스템을 비교하여 보여주고 있다.

표 3에서 보여주는 바와 같이 본 논문에서 설계한 XML 문서 암호화 시스템은 현재 모바일 환경에서 많이 사용되고 있는 JAVA기반의 시스템에 비하여 실행속도, 크기면에서 모두 향상된 성능을 갖고 있다. 라이센스 문제에 관해서는 JAVA가 오픈된 소스가 많은 현실이지만 모바일 환경에서 썬 마이크로시스템에서 로열티 지불을 요구한적 있다. 본 논문의 시스템은 표준 C언어로 구축되었기 때문에 비용문제가 없다.

또한 라이브러리의 크기나 실행속도를 고려하면 자원이 제한적인 모바일환경을 고려할 때 본 논문에서 설계한 시스템은 JAVA로 구축된 시스템에 비해 더욱 향상된 성능을 보여줄 수 있다.

다만 JAVA가 현재까지 구축된 클래스나 오픈소스가 많기 때문에 시스템의 확장성에서는 강력한 성능을 보여주고 있는데 이 부분에서는 향후 연구해야할 과제로 남는다.

표 3. 본 논문에서 제안한 시스템과 JAVA로 구축된 시스템 비교

	본 논문에서 제안한 시스템	JAVA로 구축 된 시스템
라이센스	무료	없음
실행속도	빠름	VM성능의 향상으로 속도가 향상했지만 C언어로 구축된 시스템에 비해 느림
보안성	암호화 알고리즘 표준을 따름	암호화 알고리즘 표준을 따름
크기	1M 이하	1M 이상
확장성	어려움	용이함

V. 고찰 및 결 론

본 논문에서는 WIPI 환경에서 데이터 교환에 주로 사용되는 XML 문서에서 필요한 데이터를 암호화하여 데이터 유출에 의한 피해를 대비하기 위한 연구를 하였다. XML 문서 암호화에 관해서는 데이터 보안에서 많이 사용되고 있는 DES, Triple-DES, AES, SEED, RSA 등의 알고리즘으로 암호화 시스템을 설계 및 구현하였다.

본 논문은 WIPI 플랫폼을 대상으로 개발되었으며 WIPI 플랫폼에서 처음으로 개발된 보안 시스템으로 원천기술의 확보에 의의가 있다. 또한 공개키와 대칭키 알고리즘을 결합하여 XML 문서를 다중 암호화함으로써 데이터를 효율적으로 보호할 뿐만 아니라 각 수신자는 문서 복호화를 독립적으로 진행하여 상호간에 영향을 끼치지 않는다.

응용분야로는 현재 모바일 금융서비스, DMB, Wibro, 전자상거래 등이 있으며, 모바일 환경에서 XML 문서가 광범위하게 사용됨에 따라 점차 필요성이 높아질 것으로 사료된다.

향후 연구과제로는 시스템의 공개키 암호화에서 DSA 알고리즘을 추가하여 XML 서명과 인증처리가 가능하도록 시스템을 확장하는 것이다.

참고문헌

[1] TTA, WIPI V 2.0.1. Sep 2004
[2] 한국 무선 인터넷 표준화 포럼, 모바일 표준 플랫폼 규격 2.0.1, 2004
[3] 이상윤, 김선자, 김홍남, 무선인터넷 표준 플랫폼 WIPI 2.0, 2004
[4] W3C, XML Encryption Syntax and Processing, http://www.w3.org/TR/xmlenc-core/
[5] NIST FIPS 46-3: Data Encryption Standard
[6] ANSI X9.52: Triple Data Encryption Algrithm Modes of Operation. 1998
[7] H.X.Mel & Doris Baker, Cryptography Decrypted, 2001
[8] AES page available via http://www.nist.gov/CryptoToolkit.
[9] Daemen and V.Rijmen, AES Proposal: Rijndael, AES Algorithm Submission, 1999

저자소개

홍 현 우(Hyeon-Woo Hong)

2003년 동북사범대학교 컴퓨터 공학과 (공학사)
2007년 배재대학교 컴퓨터 공학과 (공학석사)
2007년~현재 배재대학교 컴퓨터공학과 박사과정
※관심분야 : XML 암호화 및 전자서명, 웹서비스, 유비쿼터스 센서 네트워크

이 재 승(Jae-Seung Lee)

1993년 서강대학교 수학과 (이학사)
1997년 포항공과대학교 정보통신학과 (공학석사)
1997년~1999년 데이콤 정보통신연구소 연구원
1999년~현재 한국전자통신연구원 정보보호연구단 선임연구원
※관심분야 : 웹서비스 정보보호, 유비쿼터스 정보보호, 전자상거래 정보보호

이 성 현(Seoung-Hyeon Lee)

2003년 한남대학교 컴퓨터공학 (공학석사)
2006년 한남대학교 컴퓨터공학 (공학박사)
2006년~ 한국전자통신연구원 홈네트워크보안연구팀 Post-Doc
※관심분야 : 웹서비스 및 그리드 정보보호

정 회 경(Hoe-Kyung Jung)

1985년 광운대학교 컴퓨터공학과 (공학사)
1987년 광운대학교 컴퓨터공학과 (공학석사)
1993년 광운대학교 컴퓨터공학과(공학박사)
1994년~현재 배재대학교 컴퓨터공학과 교수
※관심분야 : 멀티미디어 문서정보처리, XML, SVG, Web Services, Semantic Web, MPEG-21, 정보보호, 유비쿼터스 센서 네트워크