무선 응용 프로토콜 보안 기술

경북대학교 이동근 · 김기조 · 임경식*
한국전자통신연구원 이석준 · 정병호

1. 서 론

시장조사업체인 양키그룹은 2003년 무선 인터넷 기기 보급이 10억대 이상이며 무선 인터넷을 통한 전자상거래 규모는 500억 달러 이상이 될 것으로 전망했다[1]. 국내 무선 인터넷 시장은 1999년 말부터 본격적으로 성장하여 현재 사용자가 이천만을 넘고 있는 상태이며 제공되는 콘텐츠의 종류도 멀티미디어 다운로드, 메일 송수신에서 인터넷 뱅킹, 증권, 전자상거래까지 다양하다. 특히 인터넷 뱅킹, 증권, 전자상거래 서비스는 사용자의 중요한 정보가 처리되기 때문에 보안이 중요하다. 그러나 무선 인터넷에서 보안 서비스를 제공하기에는 제약 사항이 많다. 무선망은 상대적으로 낮은 데이터 전송률과 높은 오류 발생률을 가지고 있으며, 무선 단말기들은 낮은 컴퓨팅 능력과 부족한 저장공간을 가지고 있기 때문에 많은 연산과 그에 따르는 많은 전력 소비를 필요로 하는 유선망의 보안 방식을 무선 단말기에 그대로 적용한다는 것은 어렵다. 무선 인터넷의 이러한 제약요소들을 해결하고 공통 플랫폼을 구축하기 위하여 다양한 무선 인터넷 표준이 제시되어 왔었다. 대표적인 예로 Wireless Application Protocol(WAP)[2], Microsoft Mobile Explorer(ME)[3], Lightweight Efficient Application Protocol(LEAP)[4], i-mode[5]를 들 수 있다.

WAP은 에릭슨, 모토롤라, 노키아, 폰닷컴등 4개사가 포럼을 결성하여 만든 소형 무선 단말기를 위한 무선 인터넷 프로토콜이다. 현재 포럼에는 전세계 100여개의 주요 제조업체, 네트워크 사업자들이 가입된 상태이다. WAP은 무선 단말기(client)와 인터넷 서버 사이에 프록시(proxy)역할을 하는 WAP 게이트웨이(gateway)를 두도록 하고 있다. 게이트웨이의 주요 역할은 WAP 프로토콜과 인터넷 TCP/IP 프로토콜을 중간에서 변환해 주는 것이다. 보안 기능은 무선 단말기와 게이트웨이 사이에서 제공되며 Wireless Transport Layer Security(WTLS)를 사용한다. 현재는 1.x 모델의 단점을 보완한 2.0 모델이 나와 있는 상태이다.

ME는 마이크로소프트사가 퀄컴과 제휴하여 제시한 방식으로 WAP 1.x 모델 방식과는 달리 유선 인터넷의 프로토콜 스택을 그대로 사용하는 방법이다. 하지만 무선 환경의 제약 때문에 기존 유선망의 콘텐츠를 수용할 수가 없어 m-HTML[6]이라는 새로운 마크업언어를 사용하여 서비스를 하고 있다. ME의 경우 기존 유선망 프로토콜 스택인 TCP/IP, HTTP를 그대로 사용하면서 보안방식도 기존 유선망 방식을 사용하려고 하였다. 하지만 ME 1.0에서는 SSL이 지원되지 않았으며, 최근 발표된 ME 3.0에서는 WTLS와 SSL 3.0을 지원한다.

LEAP은 WAP처럼 무선 응용에 적합하게 설계된 프로토콜이다. LEAP은 Efficient Short Remote Operation(ESRO), Efficient Mail Submission and Delivery(EMSD), Efficient Hyper Text Delivery (EHTD), Efficient Dictionary(E-DICT)로 구성되어 있다. ESRO는 신뢰적 비연결형 전송 계층이며, EMSD는 ESRO 상위 계층으로 메일전송 프로토콜인 SMTP를 최적화시킨 계층이다. EHTD는 EMSD와 함께 하이퍼텍스트 문서 전송에 최적화된 계층이며, E-DICT는 디렉토리 서비스를 위한 프로토콜 계층이다. 보안 기능은 ESRO 계층 위에 Secure Short Remote Operations(SSRO) 계층을 두어 제공한다.

* 종신회원

i-mode는 1999년 일본의 NTT DoCoMo가 개발한 패킷 기반의 이동전화 서비스이다. Compact HTML (c-HTML)[7]을 사용하며, 유선망과 무선망 사이에 게이트웨이를 두고 서비스를 한다. 초창기에는 게이트웨이와 웹서버사이에서만 SSL방식의 보안을 적용하고 무선단말기와 게이트웨이 사이에는 보안을 적용하지 않았으나, 최근에는 무선 구간까지 적용범위가 확대되었다. 이외에 최근 미들웨어 기반의 보안방식으로 JAVA VM[8]을 이용하는 방법도 사용되고 있다.

본 고에서는 위와 같은 다양한 표준들 가운데 2001년 8월에 2.0 모델을 발표한 WAP에서의 보안기술에 대하여 살펴보기로 한다. 기존의 1.x 모델에서 나타난 보안상의 문제점에 대한 분석과 WAP 2.0에서는 어떠한 해결책이 제시되었는지 알아보며, 응용계층에서의 end-to-end 보안을 유지하기 위한 방법과 무선 인터넷 인증 서비스를 위한 WPKI에 대해서 이야기하고자 한다.

2. WAP 1.x에서의 보안

2.1 WAP 1.x의 구조

초창기 무선망은 낮은 대역폭, 높은 오류 발생률, 데이터 전송 지연, 불안전한 접속 등의 문제점들을 가지고 있었으며, 단말기 또한 낮은 컴퓨팅 능력, 제한된 입·출력 장치로 인한 제약들을 가지고 있었다. WAP 1.x는 이러한 문제점들과 제약들을 완화하면서 안정적인 무선 인터넷 서비스를 제공하기 위하여 설계되었다. 이러한 설계에는 상호운용성(interoperability), 확장성(scalability), 효율성(efficiency), 신뢰성(reliability) 그리고 보안성(security) 등의 요구사항이 반영되었다.

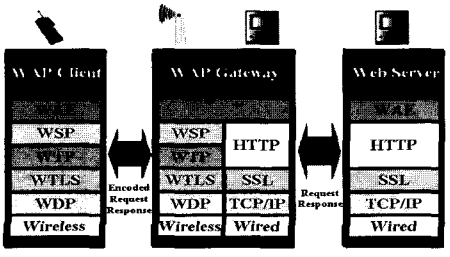


그림 1 WAP 1.x 구조

그림 1은 WAP 1.x 모델의 구조를 나타낸 것이다. 1.x 모델은 무선망에 특성화된 프로토콜을 구성하고 연결 분리(split connection)기법을 사용하여 무선 단말기를 인터넷에 연결시켰다. 이러한 방식은 유선망에서 사용하는 기존 프로토콜의 변화 없이 무선망과 연동할 수 있다. 연결 분리 기법에는 무선 단말기로부터 수신한 인코딩된 요구 메시지를 HTTP[9] 요구 메시지로 전환하고 웹 서버로부터 수신한 응답 메시지를 인코딩된 응답메시지로 변환하여 무선 단말기로 전달하는 게이트웨이가 필요하다. 게이트웨이에서는 텍스트 기반의 Wireless Markup Language (WML)[10]와 WMLScript[11] 문서를 바이너리 문서로 인코딩하는 기능을 제공한다[12]. WAP 1.x에서 프로토콜 스택은 Wireless Application Enviorment(WAE)[13], Wireless Session Protocol (WSP)[14], Wireless Transaction Protocol(WTP) [15], Wireless Transport Layer Security(WTLS) [16], Wireless Datagram Protocol(WDP)[17] 및 무선망으로 구성되어 있다.

2.2 WAP 1.x의 보안

WAP 1.x 모델은 보안 서비스를 위하여 WTLS를 사용한다. WTLS는 Transport Layer Security (TLS)[18]를 무선 환경에 맞게 변형한 프로토콜이다. TLS는 Secure Sockets Layer(SSL)[19]를 IETF가 수정하여 표준화한 것이다.

그림 2는 WTLS의 구조를 나타낸 것이다. WTLS에서 Handshake 프로토콜은 보안 세션을 설정하기 위하여 필요한 정보를 교환하는데 사용한다. 이때, 서버와 클라이언트는 프로토콜 버전과 대칭키 암호 알고리즘을 결정하고, 대칭키를 생성한다. 그리고 인증서 교환을 통하여 상호 인증을 수행한다. Record 프로토콜은 핸드쉐이크 이후 서버와 클라이언트가 합의한 보안 설정 값을 바탕으로 데이터를 압축, 인증코드를 첨가한 후 암호화하여 전송하는 작업과 수신한 데이터를 복호화하고 인증코드값을 검사하고 데이터를 해제하는 기능을 한다. Alert 프로토콜은 통신 중에 발생한 문제에 대하여 알려주는 기능을 한다. Change Cipher Spec 프로토콜은 데이터가 보안 설정 값을 바탕으로 통신을 시작함을 알려주는 기능을 한다[20].

WAP 1.x 모델은 end-to-end 보안에 있어 커다란

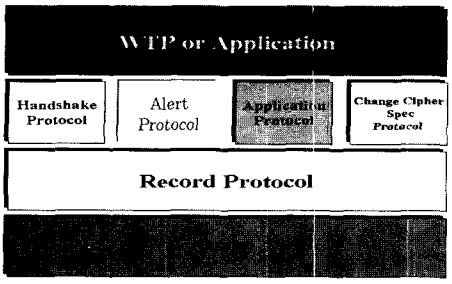


그림 2 WTLS 프로토콜 구조

문제점을 가지고 있다. 그림 1에서 보았듯이 1.x 모델은 연결 분리기법을 사용하여 유선망과 무선망을 연결한다. 연결 부분에는 유선망 프로토콜을 무선망 프로토콜로 변환시켜주는 역할을 하는 WAP 게이트웨이가 있다. 보안상의 문제점은 프로토콜 변환이 발생하는 게이트웨이에 있다. 유선망에서 SSL 방식으로 암호화된 데이터를 WTLS 방식으로 암호화하기 위해서는 SSL 방식으로 암호화된 내용이 복호화 되어야하기 때문에 데이터의 원본이 노출되어 버리는 것이다. 게이트웨이가 해킹을 당하는 경우에는 아무리 사용자의 데이터를 암호화해서 보내더라도 결국 목적지에 도착하기 전에 게이트웨이에서 원본 데이터가 노출되는 것이다. 이러한 보안 문제를 해결하기 위하여 WAP 게이트웨이를 콘텐츠 제공자의 보안 영역 안으로 포함시켜 게이트웨이의 신뢰성을 높이는 방법과 응용 계층 보안 기능을 강화하는 WMLScript Crypto Library[21]를 이용하는 방법이 있다.

3. WAP 2.0에서의 보안

3.1 WAP 2.0의 구조

WAP 1.x 모델에서는 독자적인 프로토콜을 무선망에 적용시켰으나 2.0 모델[22]에서는 기존 유선망과의 연동성을 고려하여 표준 인터넷 기술인 TCP/IP를 도입하였다. 마크업 언어는 1.x 모델에서 사용하던 WML을 Extensible Hypertext Markup Language(XHTML)[23]를 채택하여 변형시킨 WML2를 사용한다. XHTML은 HTML에 확장성과 이식성을 부여한 것으로 HTML을 XML형태로 재구성한 것이다. 유선 인터넷용 콘텐츠 개발뿐만 아니라 무선 인터넷용 콘텐츠 개발에도 사용할 수 있는 장점이 있어 무선 인터넷 서비스의 개발 속도를 매우 빠르게 할 수 있고 멀티미디어 서비스를 포함한 다양한 서비스의 제공이 가능하게 되었다. 이밖에도 데이터 동기화를 위한 SyncML과 Multimedia Messaging Service(MMS)가 포함되었고, 온라인에 저장된 데이터를 무선으로 액세스할 수 있는 Persistent Storage Interface(PSI)가 지원된다.

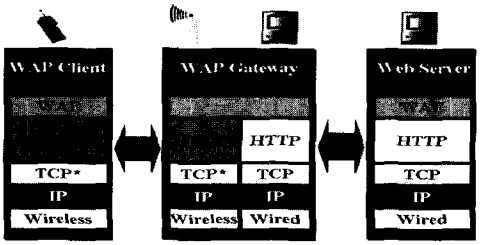


그림 3 WAP HTTP Proxy 구조

WAP 2.0은 유선망에서 일반적으로 사용하는 인터넷 프로토콜인 TCP/IP, HTTP를 도입함으로써 게이트웨이에서 프로토콜 변환으로 발생하는 문제를 해결하였다. 2.0에서는 세 가지 형태의 모델을 제시하고 있다. 그림 3은 프로파일 TCP와 프로파일 HTTP를 이용한 WAP HTTP Proxy 형태의 모델이다. 프록시를 배치하는 형태의 모델은 일반적으로 인터넷에서 널리 사용된다. 이 모델은 프록시를 무선과 유선 구간 사이에 위치시킨 경우이다. 프로파일 TCP는 기존의 TCP와 상호동작하면서 무선환경에 최적화된 프로토콜이며, 프로파일 HTTP는 CONNECT 메소드를 통한 보안채널 설정기능과 응답메시지 바디 압축 기능을 제공한다. 그림 4는 Web Server와 WAP Client사이에 TLS 터널링을 설정하여 end-to-end 보안을 제공하는 모델이다.

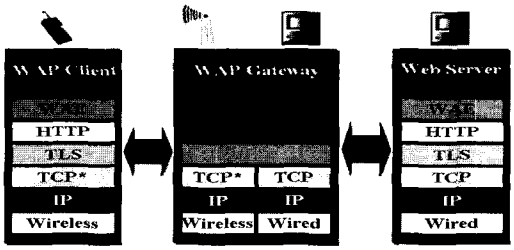


그림 4 TLS 터널링 구조

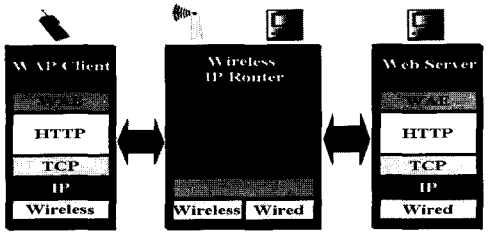


그림 5 Direct Access 구조

그림 5는 WAP Client와 Web Server가 직접 연결되는 모델이다. 게이트웨이는 무선망과 유선망 사이에서 IP 패킷을 전달하는 무선 IP 라우터 역할을 하게된다.

3.2 WAP 2.0의 보안

WAP 2.0은 TLS 터널링을 통하여 end-to-end 보안을 제공한다. 그림 6은 WAP HTTP Proxy를 통해서 TLS 터널링을 하기 위한 동작 과정을 나타낸 것이다. 터널링은 HTTP Upgrade to TLS[24]방식을 사용한다. HTTP의 Upgrade 헤더 필드는 클라이언트가 서버에게 원하는 프로토콜을 요청할 때 사용된다. 먼저 WAP Client가 CONNECT 메소드를 사용하여 Web Server에게 터널링을 위한 채널을 요청한다. 서버로부터 응답이 오면 클라이언트는 OPTION 메소드를 사용하여 보안채널 설정 요구를 보내고 Web Server는 Switching Protocols를 통해서 연결 설정을 TLS 보안 설정으로 한다는 메시지를 보낸다. 설정 확인 메시지를 수신한 클라이언트가 Client Hello 메시지를 보내면서 TLS 핸드쉐이크가 수행된다.

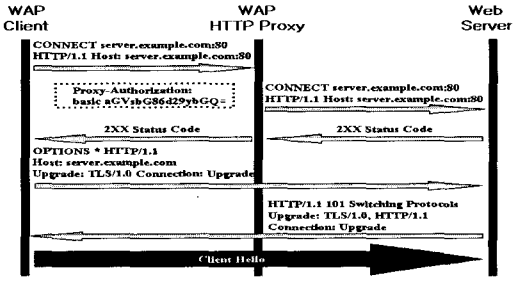


그림 6 HTTP Upgrade to TLS 동작 과정

TLS를 통해서 end-to-end 보안 서비스는 제공하지만, 전자서명 관련 서비스는 1.x처럼 응용 계층에서 처리를 하게 된다. 응용 계층 보안 서비스는 WML Script Crypto 라이브러리를 사용하여 제공한다.

4. WMLScript기반 응용 계층의 End-to-End 보안

1.x에서 2.0 모델로 바뀌면서 TLS를 통하여

표 1 WMLScript Crypto 라이브러리 확장 API

응용 계층에서의 암호화 기능	*encryptText*	wapEnvelopedString = Crypto.encryptText(flag, inputParams, prompt, defaultText, recipientCertificateChain, keyManagementAlgorithm, keyEncryptionAlorithms)
	encrypt	wapEnvelopedString = Crypto.encrypt(flag, dataToEncrypt, recipientPublicKey, keyManagementAlgorithm, keyEncryptionAlorithms, ridType, rid)
응용 계층에서의 복호화 기능	*decrypt*	decryptedString = Crypto.decrypt(wapEnvelopedString)
응용 계층에서의 보안 채널 형성 기능	*initContext*	returnData = Crypto.initContext(mechanismId, mechanismData)
	initContextFinal	status = Crypto.initContextFinal(contextHandle, mechanismDataFinal)
응용 계층에서의 보안 채널 종료 기능	*closeContext*	status = Crypto.closeContext(contextHandle)

end-to-end 보안을 제공하게 되었다. TLS를 사용하는 경우 전송계층 전체에 대한 보안이 설정된다. 이러한 경우에 전송되는 모든 데이터가 암호화되기 때문에 보안 적용이 필요하지 않은 데이터까지 암호화되어 불필요한 오버헤드(overhead)가 발생할 수 있다. 따라서 데이터에 대한 선택적인 보안 적용을 통한 오버헤드 감소가 필요하다. 최근 WAP 포럼에서는 전자서명과 관련된 signText 응용 프로그램 인터페이스(API)만이 표준화 되어있던 WMLScript Crypto 라이브러리를 확장하여 사용하는 방법이 논의되고 있다. 그 중에는 보다폰, 텔스타, 썰티콤이 제안한 encryptText, encrypt[25], 엔트러스트에서 제안한 decrypt[26], 그리고 베리사인에서 제안한 initContext, initContextFinal, closeContext[27] API가 있다. 표 1은 확장 API들의 기능과 형태를 나타내고 있다.

현재 이러한 방법들은 계속적으로 변경 요구(Change Request)가 되고 있는 상태이며 아직 정식으로 WMLScript Crypto 라이브러리에 포함되어 있지는 않다.

4.1 E2E Crypto 라이브러리

위에서 언급한 스크립트 기반의 응용 계층 보안 모델이 WAP 포럼에서 정식으로 표준화한 방법은 아니지만 무선 인터넷에서 end-to-end 보안을 유지하는데 효과적으로 이용될 수가 있다. 따라서 WAP 포럼에서 제안된 방법들을 기반으로 보안 라이브러리를 구현하였다.

구현된 모델은 그림 7과 같다. 클라이언트는 기본 WMLScript 라이브러리와 확장된 Crypto 라이브러리를 사용하는 인터프리터를 사용한다. 기본 라이브러리에는 WAP 1.x에서 사용하던 Lang, Float, String, Url, Wmlbrowser, Dialogs 등이 있다. 확장된 Crypto 라이브러리는 handshake, signText, encrypt, decrypt API 등을 가진다. 웹서버는 윈도우즈 운영체제의 인터넷 서비스인 IIS를 기반으로 handshake, verifysigntext, encrypt, decrypt API 등을 가지는 Crypto ASP 컴포넌트를 생성하여 사용한다. 표 2는 E2E Crypto 라이브러리에 구현된 API를 나타낸 것이다. handshake는 WAP 포럼에서 베리사인이 제안한 initContext처럼 보안 세션을 설정하는 기능을 하며, 서버의 verifysigntext는 클라이언트가 signText를 사용하여 서명한 내용을 검증하는 기능을 하고, encrypt와 decrypt는 특정 문자열을 암복호화 할 때 사용한다. 구현된 API들은 WTLS 보안 라이브러리와 연동하여 동작한다. 전체적으로 포럼에서 제안한 표준 API들 보다 좀더 간단한 인자 설정을 통하여 사용자가 쉽게 이용할 수 있도록 하였다.

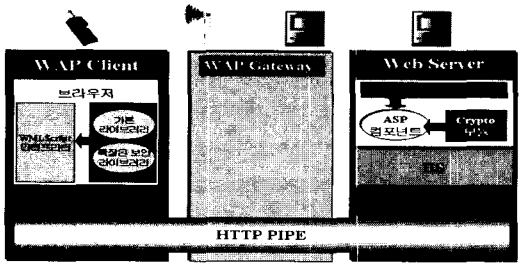


그림 7 E2E Crypto 라이브러리 보안 모델

표 2 E2E Crypto 라이브러리 API

클라이언트	Bool handshake(String handshakeURL, String nextpageURL)
	String signText(string ToSign, int options, int KeyidType, String Keyid)
	String encrypt(string ToEncrypt)
	String decrypt(string ToDecrypt)
서버	HRESULT handshake()
	HRESULT verifysigntext(BSTR signin, BSTR indata, BSTR url)
	HRESULT encrypt(BSTR plaintext, BSTR *rciphertext)
	HRESULT decrypt(BSTR ciphertext, BSTR *rplaintext)

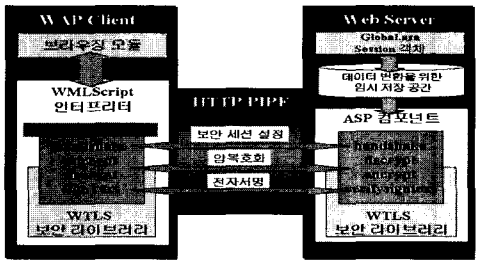


그림 8 E2E Crypto 라이브러리 동작 모델

E2E Crypto 라이브러리의 동작 모델은 그림 8과 같다. 웹서버의 컨텐츠는 ASP 페이지로 제공이 된

다. 클라이언트가 handshake 스크립트 API를 통하여 웹서버의 ASP를 참조할 때, 서버에서는 Crypto ASP 컴포넌트가 생성되고, Crypto 객체의 hanshake API가 호출되면서 핸드쉐이크가 수행된다. 이 과정에서 서버의 Global.asa에 설정된 Session 객체는 WTLS 라이브러리를 사용하면서 생성되는 보안 세션 정보를 임시 저장 공간을 통하여 객체 내부에 저장한다. 클라이언트 경우는 프로세스가 생성하는 메모리 영역에 보안 세션 정보가 저장된다. 세션 관리 및 유지는 IIS 서버가 제공하는 ASP 세션 매커니즘을 그대로 사용한다. 클라이언트는 서버가 설정한 세션 ID 사용이 가능하도록 쿠키값을 유지 할 수 있어야 한다. 암복호화, 전자서명 과정은 보안 세션 정보를 바탕으로 이루어지며, 전자 서명의 경우 인증서를 자기 자신이 가지고 있는 셀프 인증서 방식과 인증서버로부터 인증서를 받아서 사용하는 방식을 사용할 수 있다.

5. WPKI

무선 인터넷 환경에서 공개키 기반 구조(PKI)를 구축하는데 있어 일반적인 요구사항으로는 유선 PKI와 구조를 동일하게 유지함으로써 변화되는 내용을 최소화하는 것과, 인증서 취소 목록을 유선 환경에서와 같이 X.509 v.3를 사용하는 것과 대역폭과 CPU 성능 및 기억 장치를 고려하여 인증서 검증 메커니즘을 경량화 하는 것 등이 있다[28]. WAP 포럼은 무선 환경에 적합한 WAP Public Key Infrastructure(WPKI)[29]를 표준화하였다. WAP 1.x에서 PKI는 기본적으로 WTLS 보안을 전제로 하였으나 2.0에서는 TLS를 기반으로 표준화 작업이 진행중이다. PKI에서 가장 기본이 되는 인증서 검증은 큰 컴퓨팅 능력을 필요로 한다. 때문에 현재의 무선 단말기에는 부담이 된다. 이를 해결하기 위하여 WAP 1.x 모델에서는 인증서의 유효기간을 짧게 하여 사용하는 Short Lived Certificate(SLC)와 제 3자를 통한 인증서 확인 방법인 Online Certificate Status Protocol(OCSP)를 도입하였다. WAP 2.0 모델에서는 TLS를 지원할 수 있는 외부 장치로써 무선 단말기의 인증서 검증 부하를 분담하여 처리하는 방법이 검토 중에 있다. 보안 알고리즘으로는 RSA가 많이 사용되고 있다. 하지만, 안전성 확보를 위해선 1024비트 이상의 공개키를 사용해야 하므로 무선 단말기에는 상당한 부담이 되는 알고리즘이다. 현재는 512비트 이하의 키를 사용하여 부담을 줄이고자 하지만 보안상 위험을 감수해야하는 문제가 있다. 최근에는 무선 인터넷 보안에 적합한 새로운 공개키 암호기술인 타원곡선암호(ECC:Elliptic Curve Cryptosystem)를 적용하는 방법이 검토 중에 있다. ECC는 160비트의 키로 1024비트 키를 이용하는 RSA방식과 동등한 보안성을 제공하는 것이 가능하기 때문에 작은 메모리와 처리능력이 제한된 무선 단말기와 스마트카드 등에 적합하다[30].

6. 결 론

본 고에서는 여러 가지 무선 인터넷 표준들 가운데 최근 2.0버전을 발표한 WAP의 보안 기술 동향에 대하여 살펴보았다. 1.x 모델에서 무선 단말기와 웹서버사이에 프로토콜 변환 게이트웨이를 두는 연결 분리기법 때문에 end-to-end 보안을 제공하지 못한 문제점을 2.0 모델에서는 표준 인터넷 프로토콜을 수용하여 TLS 터널링 방법을 통해 해결하였다. 최근 WAP 포럼에서 논의되고 있는 응용 계층 end-to-end 보안과 관련하여 표준화 작업중인 WMLScript 기반의 Crypto 라이브러리 확장 방식이 있다. 이를 기반으로 실제로 E2E Crypto 라이브러리 모델을 구현하였다.

IMT-2000 상용화를 앞둔 지금 무선 인터넷 환경은 초기의 문제점들인 낮은 대역폭, 높은 오류 발생률, 데이터 전송 지연, 불안전한 접속 등을 개선하고 있으며, 무선 단말기 또한 과거와는 비교가 되지 않을 정도로 높아진 컴퓨팅 능력, 사용자 중심의 입·출력 장치 개발 등으로 발전해 가고 있다. 이러한 환경 개선은 무선 인터넷 사용률을 더욱 높여 줄 것이며 무선 인터넷을 통한 전자상거래는 신뢰성 있는 보안 솔루션을 제공받으면서 그 규모가 더욱 커질 것이다. 결국 무선 인터넷과 유선 인터넷은 서로 분리된 환경이 아니라 하나의 통합된 모습으로 이루어질 것이며 무선 인터넷 보안 솔루션 역시 유선 인터넷과의 상호 운용성에 초점을 맞추어 사용자 중심의 유·무선 통합 환경에 최적화된 방향으로 나아갈 것이다.

참고문헌

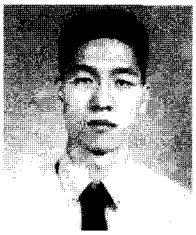
[1] The Yankee Group, "Yankee Projects Over 1 Billion Wireless Devices Worldwide By 2003," NEWS RELEASE November 15, 2000 http://

www.yankeegroup.com
[2] WAP, "Wireless Application Protocol Arichtecture Sepcification," WAP Forum, November 8, 1999 http://www.wapforum.org/WAP
[3] Microsoft, "Mobile Phones," http://www.microsoft.com/mobile/phones/default.asp
[4] LEAP, "Overview of the LEAP Protocols," LEAP Forum, August 4, 2000 http://www.leapforum.org/
[5] NTT, "What is i-mode," http://www.nttdocomo.com
[6] Microsoft, "Microsoft Mobile Explorer 1.0 Specification," May 1999.
[7] NTT, "DoCoMo i-mode," November 1999.
[8] Tim Lindholm Frank Yellin, Sun Microsystems, "The Java(TM) Virtual Machine Specification," http://java.sun.com/docs/books/vmspec/
[9] R.Fielding, et al.m, "Hypertext Transfer Protocol-HTTP 1.1," January 1997
[10] WML, "Wireless Markup Language," WAP Forum, November 8, 1999 http://www.wapforum.org/
[11] WMLScript, "Wireless Markup Language Script," WAP Forum, November 8, 1999 http://www.wapforum.org/
[12] 김기조, 최윤석, 최은정, 임경식 "무선 응용 프로토콜 기술," 정보처리학회지 pp. 44-55 2000년 5월.
[13] WAE, "Wireless Application Environment Protocol Specification," WAP Forum, November 8, 1999 http://www.wapforum.org/
[14] WSP, "Wireless Session Protocol Specification," WAP Forum, November 8, 1999 http://www.wapforum.org/
[15] WTP, "Wireless Transaction Protocol Specification," WAP Forum, November 8, 1999 http://www.wapforum.org/
[16] WTLS, "Wireless Transport Layer Security Protocol Specification," WAP Forum, November 8, 1999 http://www.wapforum.org/
[17] WDP, "Wireless Datagram Protocol Specification," WAP Forum, November 8, 1999 http://www.wapforum.org/
[18] T. Dierks, C. Allen, "The TLS Protocol," January, 1999 http://www.ietf.org/
[19] Alan O, Freier, Philip Karlton, Paul C. Kocher, "The SSL Protocol version 3.0, Internet Draft," 1996, http://home.netscape,com/eng/ ssl3/
[20] 원유재, "무선 응용 프로토콜 보안 기술," 정보통신기술, 14권, pp. 34-35, 2000년 5월.
[21] WMLScript Crypto, "WMLScript Crypto API Library," WAP Forum, November 1999, http://www.wapforum.org/
[22] WAP 2.0, "Wireless Application Protocol Architecture Specification," WAP Forum, July 15, 2001 http://www.wapforum.org/
[23] S. Pemberton et al., "XHTML 1.0: The Extensible HyperText Markup Language," W3C Aanbeveling, January 2000.
[24] S. Lawrence, "Upgrading to TLS Within HTTP/1.1," May 2000, http://www.ietf.org/rfc/rfc2817.txt
[25] Vodafone, Telstar, Certicom, "Change Request WMLScript Crypto API," June 2001, http://www.wapforum.org/
[26] Entrust, "Change Request WMLScript Crypto Specification," June 2001, http://www.wapforum.org/
[27] VeriSign, "Change Request WMLScript Crypto Library Specification," Aug 2001, http://www.wapforum.org/
[28] 원유재, "무선 응용 프로토콜 보안 기술," 정보통신기술, 14권, p.41, 2000년 5월.
[29] WPKI, "Wireless Application Protocol Public Key Infrastructure Definition," WAP Forum, April 2001, http://www.wapforum.org/
[30] 유병수, "무선통신 암호 알고리듬 어디까지 왔나," 전자신문, 2001.8.14.

이 동 근

2001 경북대학교 컴퓨터과학과(이학사)
2001~현재 경북대학교 컴퓨터과학과 (석사과정)
관심분야:무선 인터넷, 네트워크 보안, 컴퓨터통신
E-mail:leedg@ccmc.knu.ac.kr

김 기 조



1999 경북대학교 컴퓨터과학과(이학사).
2002 경북대학교 컴퓨터과학과(이학석사).
2002~현재 경북대학교 컴퓨터과학과 박사과정.
관심분야:이동 컴퓨팅, 무선 세션 프로토콜, HTTP, 윈도우즈CE 커널, 디바이스 프로그래밍
E-mail:kijo@ccmc.knu.ac.kr

임 경 식

1982 경북대학교 전자공학과(공학사)
1985 한국과학기술원 전산학과(공학석사)
1994 University of Florida 전산학과 (공학박사)
1985~1998 한국전자통신연구원 책임연구원, 실장
1998~현재 경북대학교 컴퓨터과학과 조교수
관심분야:이동 컴퓨팅, 무선 인터넷, 홈 네트워킹, 컴퓨터통신
E-mail:kslim@ccmc.knu.ac.kr

이 석 준

1998 서울대학교 컴퓨터공학과(공학사)
2000 서울대학교 컴퓨터공학과(공학석사)
2000~현재 한국전자통신연구원 무선 인터넷보안연구팀
관심분야:암호이론, 전자지불, 무선인터넷 보안, AAA
E-mail:junny@etri.re.kr

정 병 호

1988 전남대학교 컴퓨터과학과(이학사)
2001~현재 충남대학교 컴퓨터과학과 (박사과정)
1988~2001 국방과학연구소, 선임연구원
2001~현재 한국전자통신연구원 무선 인터넷 보안연구 팀장
관심분야:무선 인터넷 보안, 이동통신, 네트워크 보안
E-mail:cbh@etri.re.kr