웹에 기반한 전자 서명 제공 멀티미디어 메일러

이 성 민† · 김 태 우†† · 김 태 윤†††

요 약

전자 메일과 월드 와이드 웹은 가장 보편적으로 사용되는 인터넷 응용 서비스이다. SMTP 프로토콜을 사용하여 멀티미디어 데이터를 전송하는데는 몇 가지 문제점이 있다. 이러한 문제점을 해결하기 위해서 MIME 프로토콜이 제안되었다. 전자 메일을 통해서 중요한 자료를 전송하는 사용자의 수가 증가함에 따라 전자 메일은 전자 서명 제공을 필요로하게 되었다. 제안한 시스템은 MIME 프로토콜을 지원하며 RSA와 DES-CRC 알고리즘을 사용해서 설계되었다. CGI를 이용하여 메시지에 서명, 변환 과정을 거친 후에 전송한다. 본 시스템은 메시지 위조를 방지하여 사용자에게 안전성을 제공한다.

A Web-based Multimedia Mailer Supporting Digital Signature Scheme

Sung-Min Lee† · Tai-Woo Kim†† · Tai-Yun Kim†††

ABSTRACT

Electronic mail and World Wide Web are the most widely used Internet application services. There are some problems that multimedia data can not be transferred using simple mail transfer protocol(SMTP). In order to solve the problem, multipurpose internet mail extensions(MIME) is proposed. When the number of people who transfer important data via E-mail is increased, it is needed that E-mail provides digital signature scheme. The proposed system is designed according to the MIME protocol, RSA and DES-CRC algorithm. Using CGI, a message is signed, translated and sent. The system provides safety with users as it prohibits E-mail message from forge.

1. 서 론

전자 메일과 월드 와이드 웹은 인터넷 서비스중 가장 보편적으로 사용된다. 오늘날과 같이 인터넷이 보편화 될 수 있도록 사용자에게 편리성을 제공한 것이 월드 와이드 웹이다. 사용자들은 하이퍼링크를 통해서 쉽게 원하는 정보에 접근할 수 있다. 전자 메일의 전송규약은 1982년에 발표된 RFC 821/822에 명시되어 있다[1,2]. RFC 821에서는 SMTP(Simple Mail Transfer Protocol)의 전송 규격을 명시하고 있고, RFC 822에는 헤더와 바디부분의 메시지 형식이 명시되어 있다. 전자 메일의 전송은 SMTP를 통하여 이루어지는데, 7비트의 ASCII 형태의 메시지만 전송할 수 있고 메시지 한 라인의 길이가 1000바이트를 넘지 못한다는 규정이 있다[3,4].

컴퓨터 산업과 멀티미디어 관련 기술의 급격한 발전은 사용자들로 하여금 멀티미디어 메일러에 대한 요구를 증대시켰다. 그러나 SMTP가 지닌 제약 때문에 멀티미디어 데이터를 전송하기에는 어려움이 있었다. 이러한 제약을 극복하여 멀티미디어 메시지를 전달할 수

† 준 회 원 : 고려대학교 컴퓨터학과 대학원
†† 정 회 원 : 성공회대학교 정보통신학과 조교수
††† 종신회원 : 고려대학교 컴퓨터학과 교수

논문접수 : 1998년 3월 24일, 심사완료 : 1998년 6월 1일

있도록 제안된 프로토콜이 MIME(Multipurpose Internet Mail Extensions)이다. 1992년에 RFC 1341/1342에서 처음으로 발표되었고, 1993년 RFC 1521/1522에서 개정된 버전의 MIME이 발표되었다 [5,6].

오늘날과 같은 정보화 사회에서 전자 메일은 통신 수단으로써 전화나 편지 이상의 중요한 역할을 담당하고 있다. 전자 메일 사용의 보편화는 메시지 전송에 있어서 수신인이 그 메시지를 보낸 사람이 실제로 처음 메시지를 작성한 사람인지 혹은 중간에 메시지가 위조되지 않았는지에 대한 검증이 필요하게 되었다. 하지만 기존의 멀티미디어 메일러들은 보안성이 취약하다는 단점을 지닌다.

본 연구에서 제시한 멀티미디어 메일러는 UA(User Agent)로서 웹브라우저를 사용하였고, CGI를 통해서 전자 서명, 메시지의 전송, 변환, 관리하도록 설계하였다. 메타메일, 넷스케이프 메일 등의 기존의 멀티미디어 메일러가 이미 사용되고 있지만 문제점을 지니고 있다. 메타메일은 GUI를 제공하지 않으며, 넷스케이프 메일은 현재 많이 사용되고 있지만 자신의 컴퓨터가 아닌 다른 사람의 컴퓨터에서 메시지를 확인할 경우는 웹브라우저 소유자가 설정해 놓은 전자 메일 환경설정을 바꾸어야 하는 불편함을 가지고 있다. 또한 기존의 멀티미디어 메일러의 대부분이 메시지의 진위성 확인과 기밀성을 보장하는 방법을 제공하고 있지 않다. 본 논문에서는 이러한 문제점을 해결하기 위해서 웹에서 CGI를 기반으로 하고, 메시지 수취인이 메시지가 위조되지 않았고, 전자 서명을 확인하여 송신자를 신뢰할 수 있는 멀티미디어 메일러를 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 전자 서명 제공 멀티미디어 메일러의 필요성과 관련 암호화 기술을 살펴보고, 3장에서는 웹 기반 전자 서명 제공 멀티미디어 메일러를 설계하며, 4장에서는 전자 서명 제공 멀티미디어 메일러를 구현한다. 5장에서는 결론 및 향후 연구 방향을 제시한다.

2. 전자 서명 제공 멀티미디어 메일러의 필요성과 관련 암호화 기술

2.1 기존의 멀티미디어 전자 메일 시스템의 문제점

현재 많이 사용되고 있는 멀티미디어 메일 시스템으로는 메타메일, 넷스케이프 메일, 유도라 등을 들 수 있다. 메타메일의 플랫폼은 유닉스를 기반으로 하고 넷스케이프 메일은 유닉스와 윈도우즈에서 사용된다. 메타메일은 텍스트를 기반으로 하여 그래픽칼 유저 인터페이스를 제공하지 못한다. 넷스케이프 메일은 편리하여 많이 사용되고 있지만 브라우저의 소유자가 아닌 다른 사람이 메일을 확인하거나 보내기 위해서는 소유자의 전자 메일 환경설정을 옵션메뉴에서 변경해야 하는 불편함을 지니고 있다.

오늘날 전자 메일이 통신 수단으로서 보편화되어 사용되므로 메시지의 기밀성 보장과 송신자에 대한 인증이 중요하게 되었다. 그러나 멀티미디어 메일러들 중 대부분은 전자 서명 기능을 제공하고 있지 않아서 메시지 전송에 있어서 기밀성이 보장되기 어렵다는 문제가 있다. 제안한 전자 서명을 제공하는 멀티미디어 메일러는 웹브라우저가 설치되어 있는 곳이면 어디서든 접속하여 메시지를 확인하고 전송할 수 있으며, 전자 서명을 통해서 메시지의 기밀성과 송신자에 대한 신뢰성을 보장한다.

2.2 전자 서명(Digital Signature)과 RSA

전자 서명은 어떤 문서에 대한 작성자를 확인할 수 있도록 해 주는 방법이다. 대칭키를 이용한 전자 서명과 공개키를 이용한 전자 서명이 있다. 대칭키를 이용한 방식에서는 중재자의 역할이 크게 작용하고 공개키를 이용한 방식에서는 중재자가 서명을 증명해 줄 필요가 없으므로 더 효율적이다. 공개키를 이용한 전자 서명 방법은 (그림 1)과 같다.

공개키 방법의 대표적인 암호화 알고리즘으로 RSA를 들 수 있다. 이것은 큰 수를 인수분해하기 어려움에 근거를 둔 암호화 방법이다. 메시지를 숫자형태로 변환한 후, 블록단위로 암호화를 한다.

1. Alice가 자신의 비밀키로 문서를 암호화하여 문서에 서명을 한다.
2. Alice는 서명한 문서를 Bob에게 보낸다.
3. Bob은 Alice의 공개키로 문서를 복호화하여 Alice의 서명을 증명할 수 있다

(그림 1) 공개키 방식의 전자 서명 단계
(Fig. 1) Step of digital signature using public key method

키 생성 단계는 다음과 같다. 먼저 두 개의 소수 p,q를 생성한 후 p와 q의 곱으로 n을 생성한다. 그리고 오일러의 파이 함수를 사용하면 $\Phi(n)=\Phi(p)\Phi(q)=(p-1)(q-1)$와 같이 나타낼 수 있다[11,12]. 먼저 $\Phi(n)$과 서로소인 충분히 큰 숫자 d를 생성한다. modulo $\Phi(n)$에서 $\gcd(d,\Phi(n))=1$을 만족하는 d와 곱셈의 역원인, 즉 $e * d \equiv 1 \pmod{\Phi(n)}$을 만족하는 정수 e를 확장된 유클리디안 알고리즘을 이용하여 구하면 공개키 e와 비밀키 d가 얻어진다[9,10].

C를 암호문이라고 하고 M을 평문, E를 암호화 함수, D를 복호화 함수, V를 인증 함수, S를 서명 함수라고 가정한다. 그러면 C와 M은 각각 $C\equiv E(M)\equiv M^e \pmod n$, $M\equiv D(C)\equiv C^d \pmod n$와 같이 암호화와 복호화를 할 수 있다. 이것을 전자 서명에 적용하면, 서명 $s=D(M)$을 인증 함수 V(M,s)로 확인했을 때 $E(s)=M$이면 서명이 참이고 그렇지 않은 경우는 거짓 서명으로 판별한다[21].

2.3 메일 메시지 전송에 있어서 전자 서명의 필요성

인터넷이 보편화된 오늘날 전자 메일을 통해서 중요한 일들이 많이 처리되고 있다. 이때 메시지 수취인은 그 메시지를 작성한 사람이 실제로 메시지를 전달했는지와 메시지가 중간에 변경되지 않았는지를 확인할 필요가 있다.

전자 메일은 SMTP 프로토콜을 사용하는데, 이것은 25번 포트를 사용하여 접속한다[3,4]. 만약 사용자가 거짓 메일을 보내기 위해서 SMTP의 기본적인 5개의 명령어를 사용하면 (그림 2)와 같은 문제가 야기될 수 있다.

(그림 2)의 시나리오를 보면 smlee@netlab.korea.ac.kr라는 사용자가 메일서버에 접속해서 SMTP 명령어를 이용해서 마치 Alice@somewhere가 Bob@edilab.korea.ac.kr에게 메시지를 보내는 것처럼 하고 조작하고 있다. 물론 이러한 사실은 Alice와 Bob은 알 수 없을 것이다. 따라서 Bob이 메시지를 확인했을 때 실제로는 smlee가 보낸 메시지이지만, Alice로부터 수신한 메시지로 인식하게 된다. 만약 이 메시지의 내용이 사업을 위한 계약서나 중요한 약속 등이면 사태가 심각해질 수 있다. 또한 adversary가 패킷을 가로채서 메시지를 위조할 수도 있다. 이러한 문제를 해결하기 위해서 본 논문에서는 RSA 알고리즘을 사용하여 전자 서명을 제공하는 멀티미디어 메일러를 제안한다.

```
prompt) telnet netlab.korea.ac.kr 25
Escape character is '^]'.
220 netlab.korea.ac.kr Sendmail SMI 8.6/SMI-SVR4
ready at Sat, 7 Feb 1998 15:53:49 +0900
HELO
250 response
MAIL FROM:<Alice@somewhere>
250 response
RCPT TO:<Bob@edilab.korea.ac.kr>
250 response
DATA
354 response
오늘 약속을 취소합시다.
.
250 response
QUIT
221 response
```

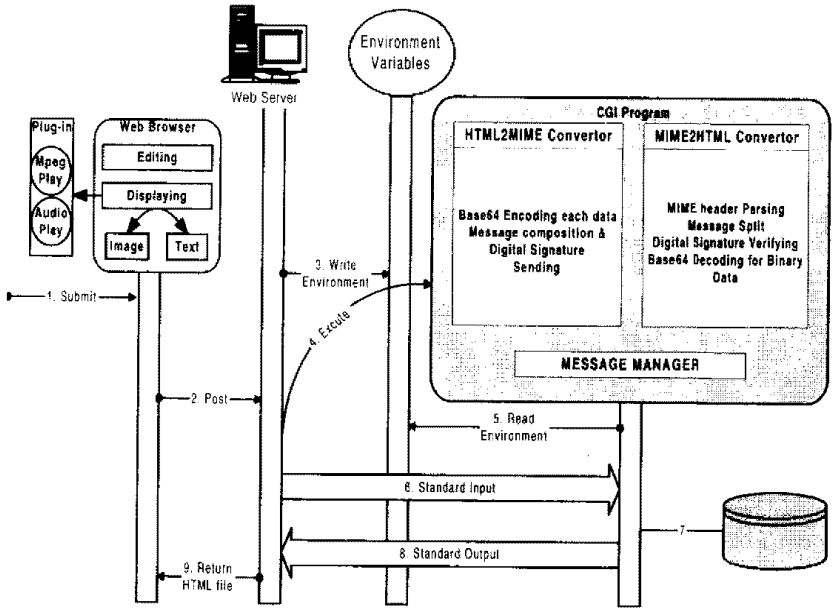
(그림 2) 송신자 조작 시나리오
(Fig. 2) Scenario of a fake sender

3. 웹 기반 전자 서명 제공 멀티미디어 메일러 시스템 설계

3.1 전체적인 시스템 구조

본 논문에서 제안하는 시스템의 전체적인 구조는 (그림 3)와 같다. 웹 브라우저를 UA(user agent)로 사용하기 때문에 웹 브라우저에서 이메일 메시지를 저작하고 확인할 수 있다. CGI 프로그램을 실행하여 결과를 얻기 위해서는 웹서버로 포스트를 한 후 필요한 환경 변수를 요청하고 CGI 프로그램을 실행한다. 이때 웹서버와 CGI 프로그램 사이의 입출력은 STDIN과 STDOUT으로 입출력 된다[16,17].

본 시스템에서 메시지의 MIME 형태로의 변환, 전자 서명, 메시지 전송, 메시지 관리는 CGI 프로그램을 통해서 수행 된다. (그림 3)에서 CGI 프로그램은 HTML2MIME 모듈과 MIME2HTML 모듈, 메시지 관리 모듈로 구성된다. HTML2MIME 모듈에서는 UA에서 저작된 HTML을 이용하여 멀티미디어 데이터를 포함한 메시지를 SMTP 프로토콜을 통해 전송 가능한 MIME 형태로 변환한다. 메시지의 멀티파트 중 텍스트 메시지 부분에 전자 서명을 하고 Content-Type에 application/ X-DigitalSignature라고 표시한후 메시지를 전송한다. MIME2HTML 모듈에서는 메시지를



(그림 3) 웹 기반 전자 서명 제공 멀티미디어 메일러 시스템의 구조
(Fig. 3) Architecture of a web-based multimedia mailer supporting digital signature scheme

디스플레이 할 때, MIME 헤더의 Content-Type의 서브필드에 X-Digital Signature라고 표시되어 있으면 전자 서명을 확인하고 송신전의 형태대로 HTML과 포함된 멀티미디어 데이터들을 복원한다. 메시지 관리 모듈에서는 수신한 메시지의 저장과 삭제를 담당한다.

3.2 멀티미디어 데이터들의 통합과 메시지 변환을 통한 전송

제안된 멀티미디어 메일러는 HTML 태그를 이용하여 멀티미디어 데이터들을 포함하여 하나의 메시지로 통합하였다[13]. 이미지 데이터인 경우 IMG 태그를 사용하였고, 오디오나 비디오 데이터인 경우는 EMBED 태그와 Anchor 태그를 사용하였다. 통합된 메시지는 예는 (그림 4)과 같다.

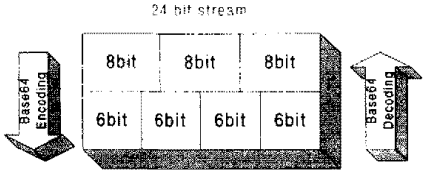
```
<img src="bee.gif">
<a href="sound.au">My voice</a>
<embed src="motion.mpg" height=144 width=160>
<embed src="voice.mid" autostart=TRUE hidden=TRUE >
```

(그림 4) HTML을 이용한 멀티미디어 데이터 통합
(Fig. 4) Integration of multimedia data using HTML

메일러에서 HTML 형태로 저작된 메시지는 멀티미디어 데이터 전송을 위해 MIME 형태로 변환된다. 이때 메시지에 포함된 멀티미디어 데이터들을 SMTP를 통해 전송 가능하도록 한라인 당 크기가 1000바이트를 넘지 않는 ASCII 형태의 텍스트로의 변환이 필요한데 Base64 알고리즘이 이용된다.

MIME에서는 MIME-Version등의 헤더와 〈표 1〉과 같은 Content-Type 및 서브 타입으로 데이터를 정의한다. MIME에서 표준으로 제시한 인코딩 방법은 Base64 방법인데, 이것은 65개의 US-ASCII 테이블을 사용하여 인코딩을 한다[5].

Base 64 알고리즘은 하나의 문자를 6비트로 만든 다음, 인코딩 테이블에서 그 숫자에 해당하는 문자로 대체한다. 인코딩은 3개의 문자로 이루어진 24비트의 입력을 받아서 4개의 문자를 생성해낸다. 디코딩은 4개의 문자로 이루어진 24비트를 다시 3개의 8비트 문자로 변환해 준다. (그림 5)은 Base64 방법을 사용하여 변환하는 구조를 나타낸다. 제안된 시스템은 각각의 멀티미디어 데이터를 Base64 알고리즘을 이용해서 인코딩하고 MIME 형식으로 변환하여 하나의 메시지로 통합하여 전송하도록 설계하였다.



(그림 5) Base 64 변환구조
(Fig. 5) Base 64 architecture

〈표 1〉 MIME의 주요 자료형
〈Table 1〉 Data type of MIME

Type	Sub Type	자료형
Text	Plain Richtext	일반 텍스트 다양한 폰트를 사용하는 텍스트
Audio	Basic	Sun Sparc 워크스테이션에서만 가능한 오디오 데이타
Image	Gif Jpeg	각각의 포맷을 가지는 이미지 데이타
Video	Mpeg	MPEG 압축 / 복원 방식의 동화상 데이타
Multipart	Mixed Alternative Parallel	복수개의 바디를 지정하여 다양한 미디어를 포함하는 메시지 구성
Application	Octet stream Postscript	사용자 응용 프로그램 정의
Message	RFC-822 Partial External-body	전자메일 형태로 길이가 긴 경우나 외부 파일을 참조하는 경우

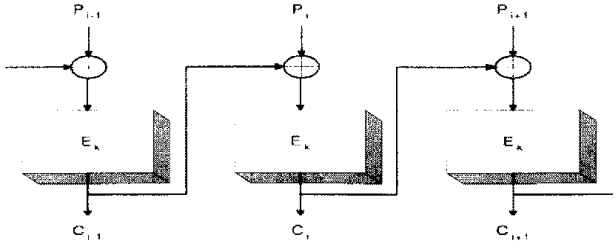
3.3 제안한 시스템에서 사용한 전자 서명 방법

전자 메일 메시지 전송시 송신자의 이메일 어드레스와 그것을 전자 서명한 내용을 MIME 헤더에 덧붙여 전송하고 수신측에서는 서명 내용이 송신자의 이메일 어드레스와 같은지 비교함으로써 간단하게 인증을 할 수 있다. 그러나 이와 같은 방법을 사용하면 송신자의 서명이 있는 메시지에서 서명과 고정된 이메일 어드레스를 복사하여 다른 메시지에 붙여서 거짓 서명을 할 수 있다. 또한 메시지의 위조도 막을 수 없다. 이러한 문제를 해결할 수 있는 한가지 방법으로 (그림 6)와 같이 Alice가 Bob에게 전자 메일 메시지를 전달하는 방법이 있다[9]. 하지만 공개키 방식을 이용해서 두 번 암호화와 복호화를 해야하므로 메시지의 크기가 클 경우는 속도나 메시지 크기면에서 효율성이 크게 떨어진다는 단점이 있다. 만약 Alice가 Bob에게 전달할 경우 (그림 6)의 첫단계와 같이 RSA 알고리즘을 이용하여 전체 메시지에 Alice의 비밀키를 이용하여 서명한다. 이때 서명된 메시지의 크기는 약 두배로 증가된다. 두 번째 단계에서 서명과 메시지를 합하여 다시 Bob의 공개키로 암호화한다. 따라서 메시지 전송을 위해서 RSA로 두 번 암호화 과정을 거치게 되므로 메시지의 크기와 서명 및 암호화 속도면에서 비효율적이다. 또한 RSA는 하드웨어에서 구현했을때 DES보다 약 100배 정도 더 느리다[9].

(1) Alice가 메시지에 자신의 비밀키로 서명한다.
$S_A(M)$
(2) Alice가 서명한 메시지와 원래의 메시지를 합하여 Bob의 공개키로 암호화하고 Bob에게 전송한다.
$E_B(S_A(M)+M)$
(3) Bob은 자신의 비밀키로 복호화한다.
$D_B(E_B(S_A(M)+M)) = S_A(M)+M$
(4) Bob은 서명과 메시지 분리후 Alice의 공개키로 메시지를 M을 복구하여 비교 한다.
$V_A(S_A(M)) = M$

(그림 6) 암호화한 전자서명
(Fig. 6) Digital signature with public key

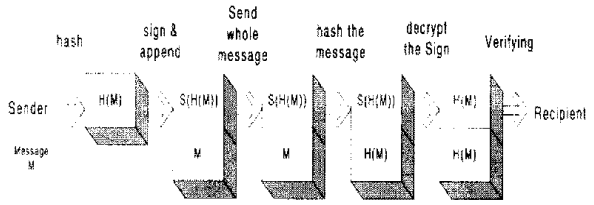
이러한 문제의 해결을 위해서 본 논문에서는 DES-CBC(cipher block chaining) 알고리즘을 해쉬함수로 이용하여 메시지로부터 생성된 64비트의 해쉬값을 RSA 알고리즘으로 전자 서명을 하였다. CBC 모드는 (그림 7)과 같이 암호화하기 전에 바로 전 단계의 암호문과 평문을 XOR하는 방법으로 $C_i = E_k(P_i \oplus C_{i-1})$과 같은 식으로 암호화 할 수 있다[9,10]. 여기서 마지막으로 생성된 64비트의 암호문을 해쉬값으로 사용한다. CBC 모드를 사용해서 생성된 64비트의 해쉬값은 이전 단계의 암호문들을 이용하여 생성되었으므로 메시지 전송 중 비트 손실이나 삽입 등의 메시지 위조가 발생했는지를 확인할 수 있다[21].



(그림 7) CBC(Cipher block chaining)모드 암호화 방법
(Fig. 7) DES-CBC mode encryption

일방향 해쉬와 RSA 알고리즘을 이용한 서명 방법은 메시지 전체에 전자 서명을 하는 대신에 메시지의 해쉬값에 서명을 함으로써 서명에 걸리는 시간이나 암호문 크기면에서 효율을 높일 수 있다.

(그림 3)의 CGI 프로그램에서의 전자 서명은 다음과 같은 순서로 이루어진다. 송신자는 DES-CRC를 사용하여 일방향 해쉬값을 생성하고 그 값을 RSA 알고리즘으로 생성한 자신의 비밀키로 암호화하여 메시지에 전자 서명을 한다. 그리고 원래의 메시지와 서명한 해쉬값을 수신자에게 전송한다. 그러면 수신자는 송신자가 보낸 메시지의 일방향 해쉬값을 생성하고 송신자의 공개키로 전자 서명된 해쉬값을 복호화한다. 이때 송신자가 생성한 해쉬값과 수신자가 생성한 해쉬값이 같다면 수신자는 송신자와 수신한 메시지 내용을 신뢰할 수 있다. 일련의 단계를 도시하면 (그림 8)과 같다.



(그림 8) 메시지 서명과 전송 및 인증 구조
(Fig. 8) Stage of digital signature and authentication

* (그림 8)에서 H(M)은 메일 메시지를 DES-CRC 알고리즘에 의해 64비트로 생성한 해쉬값이다. S(H(M))은 생성된 해쉬값에 서명하는 함수이다. RSA 알고리즘을 이용해서 서명하므로 송신자가 생성한 공개키 e와 n은 공개된다[7,8]. S(H(M))은 $s=H(M)^d \pmod n$와 같이 서명한 것이고 수신자는 송신자가 공개한 키를 사용하여 $S(H(M))^e \pmod n$와 같은 식으로 복호화하여 해쉬값 H(M)을 구한다. 이때 메시지의 해쉬값과 전자 서명을 복호화한 값이 같다면 송신자와 메시지를 신뢰할 수 있다. 이러한 메시지와 메시지의 해쉬값을 전자 서명한 값은 MIME 형태로 변환시 포함하도록 설계하였다.

4. 전자 서명 제공 멀티미디어 메일러의 구현

본 논문에서 제안된 시스템은 Sun Sparc-20에서 JAVA, C언어를 사용하여 구현되었고, 아파치 웹서버와 넷스케이프 네비게이터가 사용되었다.

(그림 9)은 본 시스템의 전자 서명 부분 중 키 생성에 대한 코드의 일부분이다. 키는 JAVA의 BigInteger 클래스를 이용하여 생성했다[19]. 먼저 난수를 생성하고 소수 p와 q를 생성하여 1024비트의 n을 구한다. 생성된 p와 q를 이용해서 Pi를 구하고 공개키 e를 생성한 후 $ed \equiv 1 \pmod{Pi}$를 만족하는 비밀키 d를 생성한다.

```
class DigitalSig {
public void KeyGeneration() {
        Random Seed = new Random();
        p = new BigInteger(512,100,Seed);
                .....
        q = new BigInteger(512,100,Seed);
                .....
        n = p.multiply(q);
        Pi= p.subtract(One).multiply(q.subtract(One));
                .....
        if (e.gcd(Pi).compareTo(One) == 0 )
                d = e.modInverse(Pi);
        }
}
```

(그림 9) 키 생성 코드
(Fig. 9) Key generation code

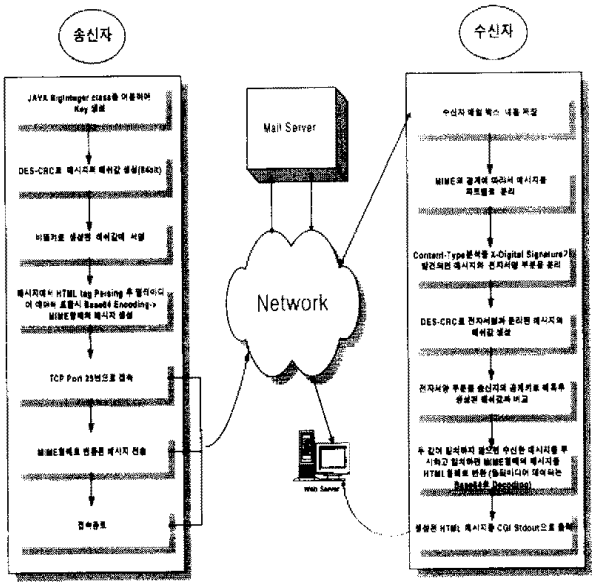
(그림 10)는 전자 서명 후 멀티미디어 데이터를 포함한 메시지를 MIME 형식으로 변환하고 TCP 포트 25번으로 메일 서버에 접속하여 수신자에게 메시지를 전송하는 코드의 일부분을 나타낸다. 먼저 MIME 형식으로 변환된 메시지를 입력 받는다. 메일 서버에 접속한 후 SMTP 명령 중 HELO, MAIL FROM, RCPT TO, DATA, QUIT의 5가지 명령을 이용해서 서버와 통신하면서 메시지를 전송한다.

본 시스템에서 메시지가 전송되는 전체적인 과정은 (그림 11)와 같다. 송신측에서는 메시지의 해쉬값을 구한후 전자 서명을 한다. 또한 MIME 형식으로의 변환 그리고 메시지 전송을 담당한다 수신측에서는 메일 박스의 내용을 읽어서 MIME의 각 부분별로 분리한 후 전자 서명 확인과 멀티미디어 데이터의 디코딩을 한 후 생성된 HTML 형태의 메시지를 STDOUT으로 웹서버로 출력한다.

(그림 12)은 메일러에서 HTML 형식으로 저작된 오디오와 비디오 데이터를 포함한 메시지의 예이다. (그림 13)는 넷스케이프에서 저작된 메시지가 CGI 프로그램에 의해서 MIME 형태로 변환된 예를 나타낸다. (그림

```
class SendMultimediaMail {
public void SendMail(String MimeMessage) {
        smtp = new Socket(MailServer, 25);
        .....
        try {
        String loc =

InetAddress.getLocalHost().getHostName();
        send(HELO + loc);
        receive();
        send(MAIL_FROM);
        receive();
        send(RCPT_TO);
        receive();
        send(DATA);
        receive();
        send(MIME_VER);
        send(MimeMessage);
        send(TERMINATOR);
        receive();
        smtp.close();
        } catch (IOException e)
        {System.out.println("Error sending: " + e);
        }
}
}
```

(그림 10) 멀티미디어 메일 전송 코드
(Fig. 10) Multimedia mail transferring code



(그림 11) 전자서명을 제공하는 멀티미디어 메일러의 동작과정
(Fig. 11) Stage of a multimedia mailer supporting digital signature scheme

12)에서의 메시지는 (그림 13)와 같은 MIME 형식으로 변환되어 선송된다.

```
<HTML>
Hello. This is an example of multimedia E-mail
message.<P>
<embed src="voice.mid" autostart=TRUE
hidden=TRUE><p>
And here is a motion picture. <embed src="play.mpg"
height=144 width=160>
</HTML>
```

(그림 12) 본 시스템에서 저작된 HTML메시지의 예
(Fig. 12) An example of HTML message

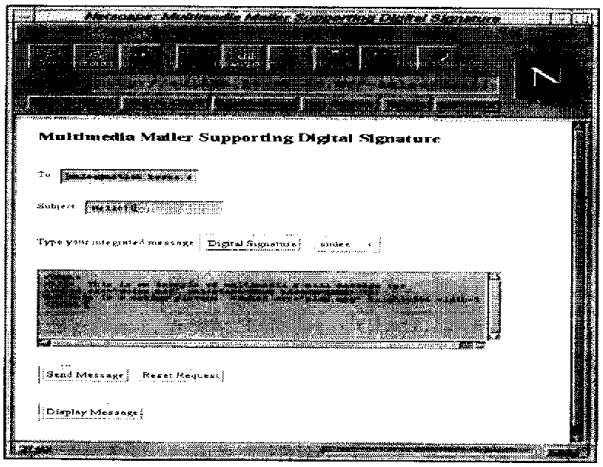
(그림 12)의 메시지의 64비트 해쉬값이 송신자의 비밀키로 전자 서명되어 (그림 13)에서와 같이 원래의 메시지에 붙여진다. 이때 전자 서명은 MIME의 메시지 바디의 멀티파트중 텍스트가 기록된 부분을전자 서명하도록 하였는데, 그 이유는 이미지나 오디오 등의 멀티미디어 데이터를 서명하면 속도나 암호문 크기에 있어서 효율이 떨어지기 때문이다. 그리고 본 논문에서 전자 서명이 들어간 부분은 Content-Type을 application/X-DigitalSig nature라고 쓰기로 정의한다. 또한 원래의 메시지와 서명 부분은 경계를 두어서 붙였는데, 서명은 (그림 13)에서와 같이 -**Digital Signature라고 시작되는 경계 아래에서부터 -**Digital Signature**-라고 끝나는 부분 사이에 위치한다.

전자 메일의 메시지는 개인의 프라이버시 보장을 위해서 메시지 소유자만이 확인할 수 있어야 한다. 이를 위해서 웹브라우저에서 URL로 서버에 접속하여 자신의 메시지를 확인할 때, 패스워드를 요구하여 사용자 인증을해야한다. 본 시스템에서는 .htacess 파일을 통해서 사용자 인증을 구현하였다[16].

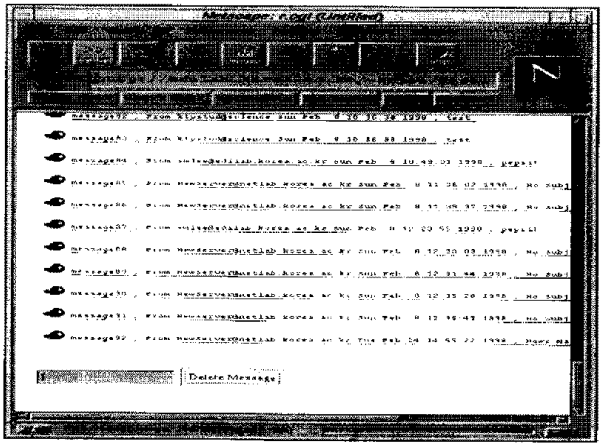
```
From: smlee@netlab.korea.ac.kr
To: Alice@somewhere.com
Subject: Sample Message
MIME-Version: 1.0
Content Type:multipart/mixed;
boundary="SMLEEBND"

- SMLEEBND
Content-Type: application/X DigitalSignature

-**Digital Signature
아래의 메시지의 해쉬값이 RSA 알고리즘을 이용하여
송신자의 비밀키로 전자 서명되어 저장된다.
-**Digital Signature**-
<HTML>
Hello. This is an example of multimedia E mail
message.<P>
```

```
<embed src "voice.mid" autostart TRUE
hidden=TRUE></p>
And here is a motion picture. <embed src "play.mpg"
height=144 width=160>
</HTML>
  SMLEEBND
Content-Type: audio/x-midi; name "voice.mid"
Content Transfer Encoding: base64
Content Disposition: inline; filename="voice.mid"
encoded data is here.
  SMLEEBND
Content-Type: video/mpeg; name="play.mpg"
Content Transfer Encoding: base64
Content-Disposition: inline; filename="play.mpg"
encoded mpeg data is here
--SMLEEBND--
```

(그림 13) 메시지 변환
(Fig. 13) Message translation



(그림 14) 메시지 저작
(Fig. 14) Message editing



(그림 15) 메시지 디스플레이
(Fig. 15) Message displaying

(그림 14)은 본 논문에서 제안된 멀티미디어 메일러이다. (그림 15)은 메시지를 저작하여 전송하기 위한 것으로 전자 서명 버튼을 누르면 생성된 비밀키와 해쉬값을 이용하여 전자 서명이 되고 그렇지 않은 경우는 일반적인 멀티미디어 메일만 전송하도록 구현하였다.

(그림 15)에서 하이퍼텍스트를 클릭하면 멀티미디어 데이터를 포함한 메시지를 확인할 수 있다. 또한 각 메시지와 관련된 데이터들이 테이블화되어 메시지와 함께 관련된 멀티미디어 데이터를 삭제할 수 있도록 구현하였다.

5. 결 론

전자 메일의 편리성 때문에 전자 메일은 거의 모든 분야에서 유용하게 쓰여지고 있다. 많은 사람들이 사용하고 또한 중요한 문서를 전자 메일을 통해서 전송하는 일이 빈번해짐에 따라서 송신자와 메시지가 변경되지 않았는가에 대한 인증이 필요하게 되었다. 또한 기술의 발전으로 인해서 사용자들은 멀티미디어 데이터를 전자 메일에 포함시키기를 원하게 되었다.

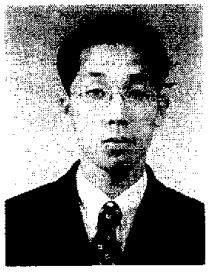
본 연구에서는 CGI를 이용한 전자 서명을 제공하는 멀티미디어 메일러를 제안하였다. SMTP에서의 멀티미디어 데이터 전송 문제를 해결하기 위해서 MIME 프로토콜이 이용되었고, 멀티미디어 데이터들을 하나의 메시지 안에 통합하여 저작하기 위해서 HTML 문법이 적용되었다. UA로서 넷스케이프를 이용하여 메시지의 저작과 확인을 하였고, CGI를 통해서 저작된 메시지를 변환, 전송, 디스플레이 및 메시지 관리를 하였다. 본 논문에서 제안한 멀티미디어 메일러는 사용자에게 메시지의 기밀성과 송신자에 대한 신뢰성을 제공한다.

향후 연구 과제로는 메시지 저작의 편의를 제공하기 위해서 메시지 입력후 버튼 클릭만으로 HTML 태그를 자동으로 생성해 주도록 하고 송신자의 공개키를 수신자가 좀더 효율적으로 획득할 수 있는 방법을 연구하고자 한다. 본 시스템은 기업에서의 전자 보고, 전자 결제에도 응용될 수 있다.

참 고 문 헌

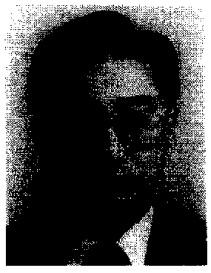
[1] Jonathan B. Postel, "Simple Mail Transfer Protocol," RFC 821, 1982.
[2] David H. Crocker, "Standard for the format

of ARPA Internet Text Messages," RFC 822, 1982.
[3] Douglas E. Comer, "Internetworking With TCP/IP," Prentice-Hall International, pp 433-446, 1995.
[4] W. Richard Stevens, "TCP/IP Illustrated, Volume 1," Addison Wesley, pp.441-459, 1996.
[5] N. Freed & N. Borenstein, "MIME(Multipurpose Internet Mail Extensions) Part One: Mechanisams for Specifying and Describing the Format of Internet Message Bodies," RFC 1521, 1993.
[6] K. Moore, "MIME(Multipurpose Internet Mail Extensions) Part Two: Message Header Extensions for Non-ASCII Text," RFC 1522, 1993.
[7] WHITFIELD DIFFIE AND MARTIN E. HELLMAN, "New Directions in Cryptography," IEEE Trans. Inform. Theory IT-22, 6 (Nov. 1976), pp.644-654.
[8] R. L. Rivest, A. Shamir and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, Vol.21, No.2, pp.120-126, 1978.
[9] Bruce Schneier, "APPLIED CRYPTOGRAPHY," SECOND EDITION, WILEY, 1996.
[10] Alfred J. Menezes, Paul C. van Oorschot, Scott A. Vanstone, "HANDBOOK of APPLIED CRYPTOGRAPHY," CRC Press, 1997.
[11] Joseph H. Silverman, "A Friendly Introduction to Number Theory," PRENTICE HALL, 1997.
[12] David M. Burton, "Elementary Number Theory," Third Edition, WCB, 1994.
[13] IAN S. GRAHAM, "The HTML Sourcebook," John Wiley & Sons, Inc., 1995.
[14] W. Richard Stevens, "UNIX NETWORK PROGRAMMING," Prentice Hall Software Series, 1994.
[15] Nemeth & Snyder & Seebass & Hein, "UNIX SYSTEM ADMINISTRATION HANDBOOK," Prentice Hall PTR, pp.439-518, 1995.
[16] Jeffry Dwight and Michael Erwin, "Special Edition Using CGI," QUE, 1996.
[17] Shishir Gundavaram, "CGI Programming," O'Reilly & Associates, Inc., 1996.
[18] Ken Arnold, James Gosling, "The Java Programming Language," Addison Wesley, 1996.
[19] JDK 1.1.5 API Documentation, Sun Microsystems, 1996.
[20] 이성민, 이형우, 김태윤, "웹에 기반한 멀티미디어 메일러의 설계 및 구현," 한국정보과학회 '97 가을 학술발표회, pp.495-498, 1997. 10.
[21] 한국전자통신연구소편, "현대 암호학," 한국전자통신연구소, 1991.



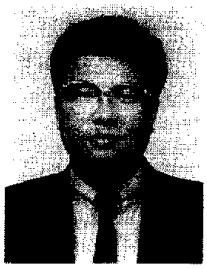
이 성 민

1997년 한림대학교 컴퓨터공학과 (공학사)
1997년~현재 고려대학교 컴퓨터학과 석사과정 재학중
관심분야 : 컴퓨터 네트워크, 네트워크 보안, 멀티미디어, 전자상거래 등



김 태 우

1984년 인하대학교 전자공학과(공학사)
1991년 인하대학교 전자계산학과 (공학석사)
1996년 고려대학교 전산과학과(이학박사)
1984년~1986년 (주)LG전자 컴퓨터사업부 시스템엔지니어
1986년~1997년 시스템공학연구소 선임연구원
1997년~현재 성공회대학교 정보통신학과 조교수
관심분야 : 컴퓨터통신, 분산시스템, 이기종컴퓨팅, 메타컴퓨팅



김 태 윤

1981년 고려대학교 산업공학과 (학사)

1983년 미국 Wayne State University 전산과학과(석사)

1987년 미국 Auburn University 전산과학과(박사)

1988년~현재 고려대학교 컴퓨터학과 교수

관심분야 : 컴퓨터 네트워크, 네트워크 보안, EDI 시스템, ISDN, 이동통신, 위성통신, 컴퓨터 그래픽스 등