Apache-Shttp : 안전한 웹 서버 개발

박 정수*, 조 은경**, 함진호*

*한국전자통신연구원, **대덕대학

Apache-Shttp: The Development of Secure Web Server

Jungsu Park*, Eunkyung Cho**, Jinho Hahm

*ETRI, **Daedok College

E-mail : *jspark@pec.etri.re.kr, **ekcho@mail.ddc.ac.kr, jhhahm@pec.etri.re.kr

요 약

다양한 응용분야에서 사용되고 있는 웹은 최근 전자상거래 등에서의 이용에 대한 기대가 한층 고조되고 있다. 그러나 이러한 웹은 전자상거래 행위에서 특히 중요한 부인봉쇄 서비스가 제공되고 있지 않으며 64비트 이하의 비도로 기밀성 서비스를 제공하고 있다.

이 논문에서는 WWW의 보안 요구사항 및 이를 위한 WWW 프로토콜로 IETF(Internet Engineering Task Force)에서 제안한 S-HTTP(Secure HTTP, Secure HyperText Transfer Protocol)를 기반으로 하는 안전한 WWW시스템 개발, Apache-Shttp를 위한 기능 설계, 개발 환경 및 개발된 시스템의 시연을 기술한다.

I. 서론

WWW의 폭발적인 사용과 함께 공개적으로 주로 사용되던 WWW를 폐쇄 집단 또는 상업적으로 이용하고자 하는 움직임이 매우 활발하게 이루어지고 있다.

그러나 기존의 WWW의 보안기술로는 이런 용도로 사용하기에는 충분치 않다고 지적되고 있다.[1] 기존의 WWW보안 메카니즘으로는 사용자 계정과 패스워드에 의한 기본 인증(Basic authentication), IP 주소에 의한 IP 필터링, 또한 최근에 이들을 보안하고자 제안된 일방향 해쉬함수를 이용한 메시지 다이제스트 인증 기법이 있다.[1]

이 논문에서는 전자 상거래 등 보안이 서비스의 중요한 영역을 차지하는 응용에서 WWW을 이용하기 위해 요구되는 WWW 보안 요구사항과 이러한 요구사항에 만족하는 WWW 보안기술로 IETF에서 연구, 제안된 S-HTTP에 근거한 안전한 WWW시스템, Apache-Shttp의 설계 및 개발에 관한 내용을 기술하고 있다.

II. 보안 요구사항 및 관련 연구

지금까지 웹에서의 보안을 위해 암호를 이용하지 않는 방법으로 기본 인증/IP 필터링/메시지 다이제스트 인증과 같은 방법이 이용되고 있으며, 암호를 이용한 방법으로 SSL(Secure Socket Layer)을 이용되고 있다. 암호를 이용하지 않는 방법들은 간단하나 점점 복잡, 다양해지는 웹 응용에서의 보안 요구사항 들을 근본적으로 해결할 수 없는 방법으로 기본적인 클라이언트 인증 서비스가 제공되고 있을 뿐으로 현재 비상업적인 용도에서만 쓰일 뿐이며 전자상거래 등 상업적으로는 현재 대부분 SSL을 이용하고 있는 추세이다.

웹은 처음부터 보안을 별로 염두에 두지 않았기 때문에 다른 대부분의 인터넷 도구들과 마찬가지로 보안을 요구하는 민감한 분야에서는 사용하기가 적합치 않다. 그러나 최근 웹을 단순한 정보 검색만이 아닌 신용카드 정보와 같이 타인에게 노출되어서는 안될 중요한 정보의 전송 등 다양한 용도로 사용하고자 하는 욕구로 다음과 같이 다양한 응용에 따른 보안 서비스가 요구된다.

○ 폐쇄 집단 구성원간의 정보 공유 : 특정 집단 구성원 사이에서 중요한 정보를 공유하고자 하는 형태로 서버에 접근하는 클라이언트를 제어하기 위해 클라이언트 인증 서비스가 요구되며 특정 정보로의 접근을 제한할 수 있다.

○ 중요 정보를 안전한 방법으로 교환/발행하기 위한 응용 : 구매 주문서를 제출하거나 공문서 같

은 중요한 정보를 발행할 때 클라이언트/서버 정보의 출처가 진짜인지와 데이터가 중간에 변경되지 않았음을 증명하는 무결성 서비스가 요구된다. 또한 이것은 클라이언트와 서버에 대한 상호 인증 서비스를 요구하며 교환되는 메시지 또는 문서 자체에 대한 인증도 요구된다.

○ 전자 지불 응용 : 웹을 이용한 전자 상거래 응용은 트랜잭션에 포함된 당사자 특히 판매자의 정당성을 입증하기 위해 인증을 포함할 수 있으며, 구매자의 지불 자체에 대해 제삼자에 의한 검증이 필수적으로 요구된다.

○기밀성 서비스 응용 : 웹을 이용하여 교환되는 정보 자체가 타인에 노출되거나 변경되지 않기를 바라는 기밀성 서비스가 요구된다.

이러한 다양한 웹 응용에 따른 요구 사항들을 요약하면 클라이언트 인증, 서버 인증, 서버상의 문서에 대한 접근 제어, 서버와 클라이언트 사이에 일어나는 트랜잭션 데이터 인증, 무결성, 그리고 기밀성 서비스가 요구된다. 이와 같은 보안 서비스 요구사항을 제공하기 위해서는 현재 널리 사용되고 있는 SSL에 의해서는 부인봉쇄 서비스가 제공되지 않기에 이러한 서비스를 필요로 하는 응용에서의 사용을 할 수가 없다. 이러한 문제를 해결하기 위해서 IETF에서는 웹 보안 프로토콜로 S-HTTP를 표준화하였다.

이 논문에서는 전자서명, 암호 및 인증을 이용하여 기밀성, 무결성, 부인봉쇄, 메시지 및 사용자 인증 서비스를 제공하는 웹 시스템으로 Apache-Shttp의 시스템 설계, 개발환경, 개발된 시스템 등을 기술하였다.

III. 웹 보안프로토콜 S-HTTP

IETF WTS(Web Transaction Security) 작업반에서 제안된 WWW 보안 프로토콜인 S-HTTP는 기본적으로 트랜잭션의 기밀성, 무결성, 발신자 인증, 발신 부인 봉쇄, 접근 제어 서비스를 제공한다. 또한 이러한 보안 서비스를 제공하기 위해 요구되는 키 관리 메카니즘, 보안 정책, 암호 알고리즘등의 협상이 가능하다. S-HTTP는 HTTP와 함께 사용을 위해 설계된 안전한 메시지 기반 통신 프로토콜 HTTP의 메시지 모델과 함께 공존하도록 설계되었으며, HTTP 응용에 쉽게 통합되도록 설계되었다.[2] S-HTTP의 동작모델은 그림 1과 같다.

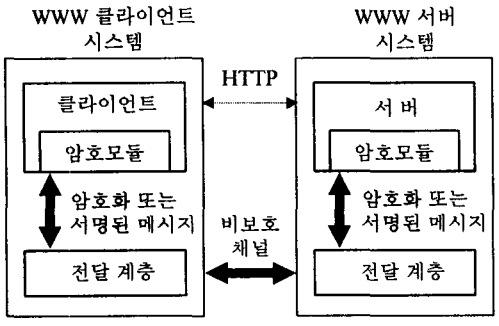

그림 1. S-HTTP의 동작 모델

VI. Apache-Shttp의 설계

안전한 WWW서비스를 제공하기 위해 확장되어야 할 WWW 서버 기능은 그림 2와 같으며, 이 기능들을 간략히 기술하면 다음과 같다.

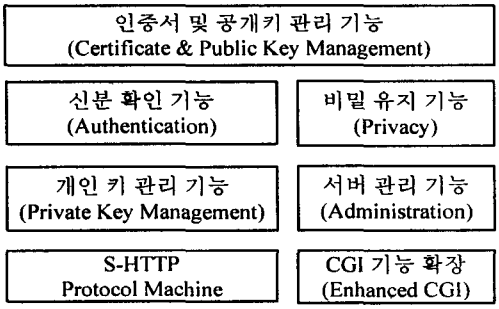

그림 2. Apache-Shttp의 확장 기능

(1) 인증서 및 공개키 관리 기능

서버는 자신의 RSA 공개키/개인키 쌍과 대응하는 공개키 인증서 요구를 생성할 수 있으며, 또한 공개키 인증서를 발급하는 기관으로부터의 공개키 인증서를 Import할 수 있어야 한다. 그리고 공개키 인증서가 자신의 시스템에 인증서 DB로 암호되어 저장된다. 지원하는 공개키 인증서 타입은 *X.509, PKCS(Public Key CryptoSystem)-7*이다.

(2) 신분 확인 기능

개인키/공개키를 이용한 전자서명을 이용하여 신분확인과 함께 전자 서명된 메시지(보통 문서)의 무결성을 검증한다. 지원하는 디지털 서명 알고리즘은RSA이며, 메시지 다이제스트 알고리즘은 MD (Message Digest)5이다. 약한 신분확인을 위해서는 사전에 준비된 키(prearranged key)를 이용한다.

(3) 비밀 (Privacy) 유지 기능

클라이언트로 송신되는 메시지를 암호하기 위해 서버의 개인키를 사용할 수 있다. 지원하는 대칭 암호 알고리즘은DES(Data Encryption Standard) 등이며 비대칭 암호알고리즘으로는 RSA이다. 키 교환 알고리즘으로는 RSA, Inband 키(즉, 사전에 암호화된 전송채널을 통해 교환되는 트랜잭션 암호 키), Outband 키(즉, S-HTTP를 통하지 않고 다른 외적인 방법에 의해 교환되는 트랜잭션 암호키) 교환을 이용한다.

(4) 개인 키 관리 기능
서버는 자신의 RSA 공개키/개인키 쌍과 대응하는 개인키를 인증서 및 공개키와는 다른 보안등급에 의해 관리하여야 한다.
(5) 서버 관리 기능
관리자는 서버를 시작 시킬때(startup) 패스워드를 이용하여 관리한다.
(6) S-HTTP Protocol Machine
WWW보안 프로토콜 S-HTTP의 Request/Response 메시지를 구성(Encoding) 또는 분해(Decoding)한다. 또한 확장된 HTTP 메시지를 구성/분해한다.
(7) CGI기능 확장
CGI(Common Gateway Interface) 변수를 이용하여 수신되는 Request의 암호상태(예, 서명, 암호화, 둘 다 또는 아무 것도 안함), 클라이언트에 의해 사용된 서명자와 대칭 키를 CGI 프로그램에 제공한다.

V. Apache-Shttp의 개발환경 및 시연

1. 개발 환경

웹 보안 프로토콜 S-HTTP를 지원하기 위하여 일반적으로 널리 쓰이고 있는 Apache 서버 1.2 Release에 S-HTTP 프로토콜을 추가하였다. 그림 3에서는 Apache서버의 구성과 함께 확장된 S-HTTP 프로토콜 엔진의 위치를 도식화하고 있다. Apache서버는 대부분의 서버 구성에 포함되는 기본 기능인 Apache core와 선택기능으로 각각의 모듈을 플러그인 형태로 쉽게 짜 맞출 수 있는 형태로 구성되어 기능확장이 매우 쉽게 되어 있다.[6] 그러나 프로토콜 확장은 이와 같지 않다. Apache 서버는 오직 HTTP만을 지원하도록 설계되어 있다. 즉, HTTP만을 지원하도록 만들어진 설계상의 제약 때문에 다른 여타의 프로토콜을 약간의 수정만을 가해서 새롭게 구현하기는 쉽지 않다. 즉, 확장되어야 할 프로토콜을 Hard coding에 의해 확장 및 수정하여야 한다.[5]

클라이언트와 서버가 공통적으로 사용하고 있는 보안 모듈은 독일 GMD에서 개발한 SECUDE -5.1 preview release III를 이용하고 있으며, 이 보안모듈은 RSA, DES, DSA, MD5, SHA와 같은 암호 및 해쉬함수, 인증서 관리기능, CA기능과 인증서 교환기능, PEM 처리기능, GSS(Generic Security Services) API 제공, PKCS(Public Key Cryptography Standards)인코딩/디코딩 기능, ASN.1 인코딩/디코딩 함수, 기밀성 및 무결성 서비스에 의한 사용자 정보의 저장기능 등을 제공하고 있다.

그림 4는 Apache-Shttp의 동작 환경으로 클라이언트의 프로토콜 엔진은 브라우저의 Plug-in 형태로 동작하고 있다.[3]

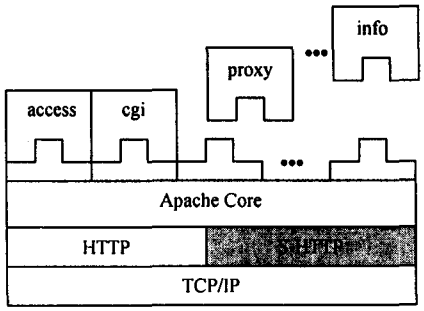

그림 3 . Apache 서버 구조

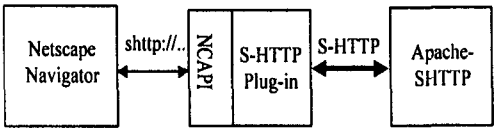

그림 4 . Apache-Shttp 동작환경

2. 시연

회원 가입 및 회원 정보 수정 과정에서의 기밀성 서비스 제공, 판매자 인증 서비스, 구매자 인증 서비스, 판매자 가격 정보의 무결성 서비스, 구매자에 의한 주문 및 구매 정보의 무결성 서비스, 부인봉쇄 서비스 등을 위해 이용되는 기본적인 몇 가지 트랜잭션의 시연 화면을 기술하면 다음과 같다.

(1) 암호화 및 서명 시연

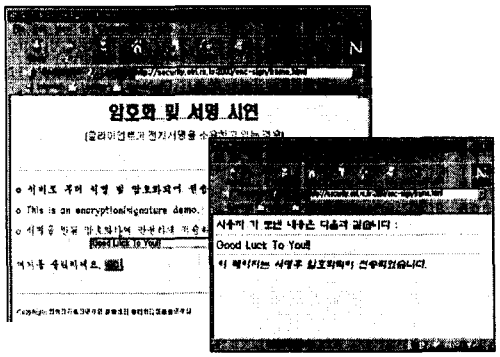

그림 5 . 암호화 및 서명 시연

그림 5의 암호화 및 서명 시연은 클라이언트 시스템이 자신의 비밀 서명 키를 가지고 있는 경우에 수행되는 화면을 보여주고 있다. 암호화와 서명 메카니즘을 동시에 사용함으로써 안전도를 높

일 수 있으며, 서명에 의해 클라이언트 인증, 부인 봉쇄 서비스 등이 제공되게 된다. 공유키가 없는 경우에는 PKCS-7의 SignedAndEnveloped Data 형식이 사용되며, 공유키가 있다면 Signed Data와 EncryptedData 형식을 순차적으로 사용한다. 이 경우에도 EncryptedData 형식이 사용되면 Prearranged-Key-Info 헤더가 항상 사용되어야 한다.

(2) 키 교환 시연

키 교환은 Inband 방식과 Outband 방식이 있다. Outband 방식은 E-Mail, Fax, 전화 등을 이용하는 방식을 의미하며, Inband 방식은 암호화 서비스 과정에서 함께 제공된다고 할 수 있다. 그림 6은 Inband 방식에 의한 키 교환 시연으로 이때 이용되는 헤더는 S-HTTP 메시지 몸체 영역에 포함될 Key-Assign 헤더이다. 물론 키의 중요도에 따라 암호화 서비스와 함께 사용되지 않고 서명 또는 인증 서비스 과정에서 사용될 수도 있다. 그러나 키는 일반적으로 제삼자에게 노출되지 않아야 하므로 암호화 서비스와 함께 사용되는 것이 바람직하다고 할 수 있다.

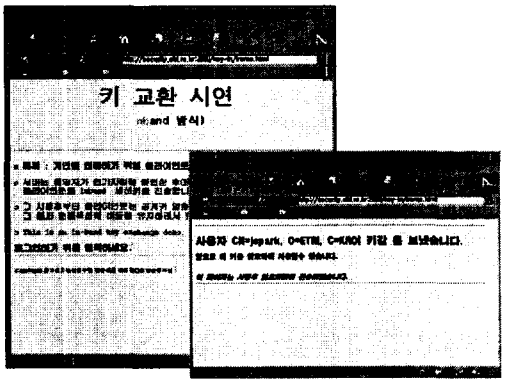

그림 6 . 키 교환 시연

(3) 인증 시연

인증 시연 화면, 그림 7은 기본적으로 웹 서버에서 제공되고 있는 기본 인증과 유사하다. 기본 인증 방식은 단지 클라이언트의 userid와 password를 서버가 알고 있고 이 정보를 정확하게 제시하는 사용자를 인증하는 방식이다. 그러나 그림 7에 의한 인증 방식은 password 정보를 S-HTTP 메시지의 몸체에 대한 해쉬 값을 계산할 때 사용하므로써 사용자를 인증하는 것과 동시에 메시지에 대한 무결성 검사도 함께 수행한다. 이때 PKCS-7의 Data 형식을 이용하게 된다. 즉, 평문으로 전송하게 된다.

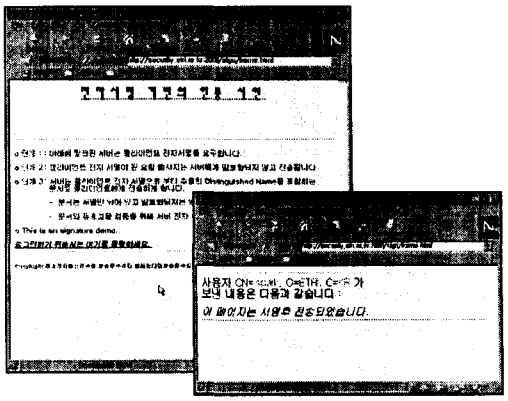

그림 7 . 인증 시연 화면

VI. 결론

본 논문에서는 웹 보안 프로토콜인 S-HTTP를 Apache웹 서버에서 지원하도록 한 Apache -Shttp웹 서버의 개발을 위한 기능설계 및 개발에 대하여 기술하였다.

Apache-Shttp는 트랜잭션의 기밀성, 무결성, 발신자 인증, 발신 부인 봉쇄, 접근 제어 서비스를 제공하므로 최근 활성화되고 있는 전자상거래 등을 위한 웹 서버로 활용될 수 있을 것으로 예측된다. 또한 보안 제품에 대한 미국 등의 수출 억제 정책으로 비도가 높은 - 키 길이가 64 bits 이상 - 보안 제품을 국내에서 사용할 수 가 있으므로 이에 대한 대안으로도 활용될 수 있을 것으로 기대된다.

참고 문헌

[1] Adam Cain, Web Security, NCSA, 5th International WWW Conference Tutorial Notes, pp. 1-31, 1996.

[2] E.Rescorla, etc. The Secure HyperText Transfer Protocol, Internet-Draft <draft-ietf-wts-S-HTTP-04.txt>, 1997.3.

[3] Netscape, OLE Automation in Navigator, http://home.netscape.com/newsrefstd/oleapi.html, 95.3.

[4] EIT, Secure HTTP, http://www.eit.com/creations/s-http

[5] API notes, http://www.apache.org/docs/misc/API.html.

[6] Welcome to the Apache Server Project, http://www.apache.org.