침입차단시스템에서 안전한 메일시스템 설계 및 구현

조영남*,전문석
숭실대학교 컴퓨터학과

Design and Implementation of Secure Mail System for Firewall System

Young-nam Cho*, Moon-suk Jun
Dept. of Computing, Soongsil University

요 약

정보화 시대를 맞이하여 인터넷의 빠른 성장을 이끌고 있는 대표적인 서비스가 바로 E-mail 이다. 그만큼 다른 서비스보다 대중적이라 할 수 있다. 그러나 다른 서비스와 마찬가지로 보안상 많은 문제점을 가지고 있다. 첫째로 메일서비스 시스템 자체의 보안상 허점을 이용해서 시스템에 침입, 파괴 활동을 할 수 있으며, 둘째로 메일의 기밀성 및 무결성이 확보되지 않아서 안전한 전달이 이루어 질 수 없다.

본 논문에서는 위에서 언급했던 문제점들을 극복할 수 있는 해결책을 침입차단 시스템에서 모색하려 한다. UNIX에서 가장 공통적인 SMTP 서버가 Sendmail 이다. Sendmail 은 매우 강력하지만 오랫동안 크고 작은 보안 문제를 가지고 있다. 다른 종류의 메일러에는 smail3,MMDF,Z-Mail 이 있지만 현재의 Sendmail 보다 더 안전하지는 못하다. 본 논문에서 Sendmail 의 문제점을 알아보고 해결방법을 찾아낸다.
또한 메일의 내용의 기밀성 및 무결성 , 송신 부인 방지, 수신 부인 방지를 위한 방법에 대하여 논의해 본다.

1. 서론

인터넷의 폭발적인 대중화로 정보화 시대가 빠르게 다가왔다. 이는 삶의 질을 한 단계 높이는데 중요한 역할을 했음은 두말할 나위가 없다. 특히 사용상의 편리함과 유용성으로 인해 컴퓨터를 전공하지 않은 사람도 쉽게 접할 수 있는 서비스 중의 하나가 바로 전자메일이다.

하지만 전자메일 서비스의 폭발적인 증가와 더불어 나타나는 문제점도 적지 않게 증가하고 있다. 첫째로 악의 있는 공격자가 메일 시스템의 허점을 노려서 메일 서비스를 제공하는 시스템에 침입하여 피해를 입힌다. 둘째로 메시지 자체를 공격하는 유형으로 메시지를 가로채어, 변조, 위조하는 행위로 메시지가 아무나 가로채서 조작할 수 있는 형태로 전달되기 때문에 일어나는 경우이다. 본 논문에서는 침입차단 시스템을 이용하여 시스템 보안, 즉 메일 서비스 시스템을 보호하며 접근제어 기능과 필터링 기능을 설계 구현하고 네트웍 보안, 즉 메시지를 암호화하여, 신뢰성 있는 메일 서비스를 가능케 한다.

2 보안 요구사항

2.1 시스템 보안 요구사항

UNIX 시스템에서 일반적으로 사용되는 메일러는 Sendmail 이다. Sendmail은 매우 강력하지만 오랫동안 크고 작은 보안 문제를 갖고 있다. 다른 종류의 메일러에는 smail3, MMDF, Z-Mail 이 있지만 현재의 Sendmail 보다 더 안전하지 못하다. 그런데 Sendmail이 왜 보안 문제를 갖고 있는 것일까. 이유 중의 하나는 매우 복잡한 프로그램이기 때문이다. 이는 여러 가지 기능을 수행하고 처리하기 위해서 root 권한을 필요로 하는데에 문제가 있다. root로서 수행하는 일을 간단히 살펴보면

- 수신되는 SMTP 접속에 대해 25번 포트를 듣는다.
- 특정 사용자로서 사용자가 소유한 .forward 파일과 include: ailas 파일을 읽고 이 파일에 의해 지정된 프로그램을 수행케 한다.
- root로 실행되도록 하는 프로그램 제한하는 커널 시스템콜을 실행하는데, 예로서 수신될 메시지를 받기 위해서 빈 공간을 결정하는 것이 있다.
- 권한 없는 사용자에 의한 snooping 으로부터 메일 큐에 있는 파일들을 보호한다.

위의 과정에서 Sendmail이 SMTP 서버로 동작 할 때 root 허가 권한은 매우 위험한데, SMTP 접속을 통해 버그를 활용하는 공격자는 root로 실행되고 있는 프로세스와 통신을 하게 된다. 프로세스는 root의 권한을 가지고 있기 때문에 공격자가 의도하는 대로 어떤 것이든지 수행할 수 있다.

위의 내용처럼 시스템 자체에 위험성을 제거하는 것을 시스템 보안이라 한다.

2.2 네트워크 보안 요구사항

인터넷 전자우편들은 일반적으로 쓰이는 우편들과 달리 봉투에 넣어서 밀봉되어 있지 않기 때문에 전송되는 과정에서 얼마든지 탈취 및 변조되어질 가능성을 가지고 있다. 전자 우편 서비스의 요구사항으로, 메시지를 작성하는 사용자는 작성된 메시지가 원하는 수신자만이 읽을 수 있기를 바라며, 수신자는 수신된 메시지가 수신자가 알고 있는 실제 송신자로부터 온 메시지임을 확인 할 수 있기를 바라고, 수신된 메시지가 중간에 변조되지 않기를 바란다. 또한 송신자가 자신이 보낸 메시지를 부인하지 않기를 원하며 반대로 수신자가 자신이 수신한 메시지를 수신하지 않았다고 부인 할 수 없게 하기를 바란다.

이렇듯 데이터가 네트워크에서 위험적인 요소로부터 안전하게 전달될 수 있게 하는 것을 네트워크 보안이라 한다.

3. 시스템 보안

3.1 기능

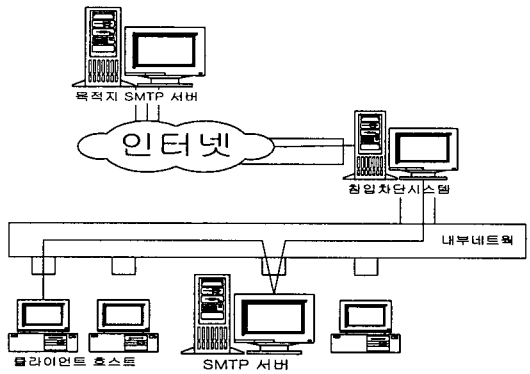
[그림 1]은 침입차단 시스템에서 메일서비스를 그림을 도시화 한 것이다. 본 논문에서 제안하는 시스템 보안은 다음과 같은 기능을 가지고 있다.

- 침입차단 시스템이 설치되는 베이스천 호스트에서 Sendmail 이 root가 아닌 setuid 로 수행 할 수 있게 만든다.
  여기서 베이스천 호스트에는 어떤 사용자도 가질 수 없기 때문에 어떤 사용자도 메일 큐 디렉토리를 억세스 할 수 없다.
- 외부로부터 오는 메일의 크기를 정하고 원하는 크기보다 크면 수신 할 수 없게 한다.
- 외부로부터 오는 동일한 메일의 사용자 수를 제한 할 수 있게 한다.
- 외부로부터 오는 메일의 특정 호스트를 제한하고 외부에서 오는 메일의 특정 사용자를 제한 할 수 있게 한다.
- 메일에 파일이 첨부되어 있는지를 체크하여 수신 여부를 가릴 수 있게 한다.
- 메일의 내용을 검사하여 특정 단어가 들어있는지를 체크하여 수신 여부를 가릴 수 있게 한다.
- 수신된 메일을 일정기간 DB에 저장하여 검색 할 수 있게 한다.

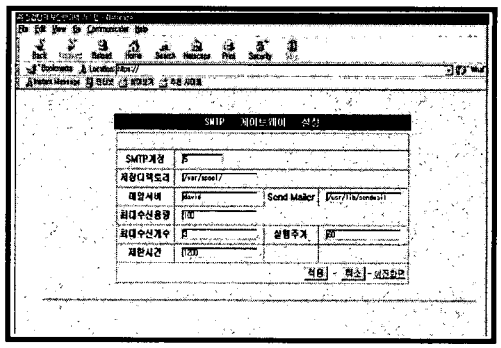
3.2 설계 및 구현

우선 sendmail 의 기능 중 받기 기능을 수정한다. 여기서 중요한 점은 root가 아닌 일반 uid의 권한으로 메일을 받게 하고 기본적인 명령어만을 해석할 수 있게 수정한다. 침입차단 시스템에는 일반 사용자의 계정을 만들지 않기 때문에 일반 사용자가 접근 할 수 없다. 외부에서 침입차단시스템 안으로 메일을 보내기 위해선 SMTP 기본 포트인 25번 포트로 접속을 시도한다. 그러면 inetd에 의해 바인딩 되어 있는 수정된 Sendmail을 작동 시켜서 메일을 받기 시작한다.

25번 포트로 접속을 하고 메시지가 전달될 때 미리 정한 크기 이상으로 받게 되면 연결을 끊고 그 동안 받은 메시지도 지운다. 메일 헤더를 분석하여 목적지와 도착지 주소를 얻어 접근제어를 할 수 있다. 또한 MIME 포맷으로 전달되는 메일의 헤더를 분석하여 파일의 첨부 유무를 확인 할 수 있다. 이렇게 해서 제대로 도착한 메일을 복사해서 제목, 날짜, 송신자, 수신자 별로 검색할 수 있도록 DB에 저장한다. 일단 침입차단 시스템에 올바르게 도착한 메일은 내부망의 메일서버로 보내지게 되고 내부 사용자는 USER AGENT를 통해 메일을 받아 볼 수 있게 된다. 구현 그림[2] 에서 보듯이 제어를 위한 초기치 입력은 웹을 이용한다. 또한 원격에서 안전하게 제어 할 수 있도록 SSL을 도입했다.



[그림 1] 침입차단시스템에서의 메일시스템



[그림 2] 구현

4.네트워크 보안

4.1 기능

본 논문에서 제안하는 네트워크 보안은 다음과 같은 기능을 가지고 있다.

- 기밀성
  송신자가 보낸 메시지가 원하지 않는 사용자에게 가로채어지더라도 메시지의 내용은 노출되어지지 않도록 한다.
- 사용자 인증
  메일을 실제로 보낸 사람이 송신자라고 주장한 사람과 일치하는가를 확인해 줄 수 있게 한다.
- 무결성
  수신자가 수신한 메시지가 전송도중 권한 없는 사용자에 의해 변

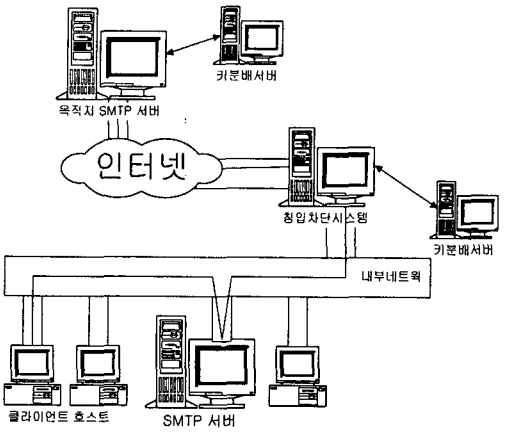
조되었다면 이러한 사실을 수신자가 확인 할 수 있게 한다.
· 송신자 부인 방지
송신자가 메시지를 수신자에게 전송한 후, 해당 메시지 전송사실을 부인 할 수 없도록 한다.
· 수신자 부인 방지
수신자가 메시지를 수신한 후 수신 사실을 부인 할 수 없도록 한다.

4.2 설계

침입차단 시스템이 외부에서 들어오는 암호화된 메일을 필터링 하기 위해서는 일단 침입차단 시스템에서 복호화를 해야 한다. 그리고 필터링 된 메일은 다시 암호화 하여 내부 메일 서버로 보낸다.
반대로 내부에서 암호화된 메시지는 침입차단시스템에서 복호화 된 후 필터링 된다. 그리고 다시 암호화 되어서 수신자에게로 전해지게 된다.

4.2.1 침입차단 시스템과 수신자 사이의 암호화
우선 침입차단시스템은 키 분배 서버를 가지고 있다
수신자도 자신의 도메인에 맞는 키 분배 서버를 갖는다고 가정한다.
침입차단시스템에서 송신자의 ID와 수신자의 ID,그리고 메시지를 가지고 hash함수를 이용해서 얻은 해쉬값을 키분배 서버에 전달한다. 여기서 해쉬값은 무결성을 위한 값도 되지만 나중에 송,수신 부인 방지를 위한 식별자가 되기 위한 값이다. 위의 값을 받은 키분배 서버는 침입차단시스템에 일회용 키인 세션키와 자신의 비밀키로 암호화한 해쉬값과 수신자측 키분배서버의 public 키로 암호화한 송신자ID,수신자ID, 세션키를 전달한다.
침입차단시스템은 위에서 전달받은 세션키로 메시지를 암호화한 값과, 수신자 키분배 서버의 public키로 암호화한 송신자ID, 수신자ID, 세션키를 여러 MTA를 거쳐서 전달하고자 하는 수신자에게 전달한다.
수신자는 암호화된 메시지를 뺀 암호화된 송신자ID,수신자ID,세션키,해쉬값을 수신자측 키분배 서버에 전달한다.
값을 전달받은 수신자측 키분배 서버는 자신의 비밀키로 송신자 ID, 수신자ID, 세션키를 복호화하고 송신자 키분배서버의 public키로 해쉬값을 복호한다. 송신자 ID와 수신자ID,해쉬값은 저장하고 세션키와 해쉬값을 수신자에 전달한다.
수신자는 세션키를 이용해서 암호화된 메시지를 복호화하고 해쉬함수를 이용해서 무결성을 검증한다.

4.2.2 침입차단시스템과 송신자 사이의 암호화
침입차단시스템의 내부에 있는 사용자가 외부에 메일을 보내려면 반드시 침입차단시스템을 거쳐야 되는데, 이때 내부의 임의적 사용자가 메일을 가로채서 훔쳐 볼 수 없도록 암호화해야 한다.
송신자가 외부의 수신자로 메일을 보낼 때 송신자는 메시지를 대칭키를 이용해서 암호화하고 침입차단시스템의 public키로 대칭키를 암호화해서 침입차단시스템에 보낸다. 또한 무결성을 위해서 메시지를 해쉬함수로 돌려서 얻어낸 해쉬값을 자신의 비밀키로 암호화 한다음 침입차단시스템에 보낸다. 침입차단시스템은 자신의 비밀키를 이용해서 대칭키를 복호화하고 복호화된 대칭키로 메시지를 복호화 하고, 송신자의 public 키로 암호화된 해쉬값을 복호화 해서 복호화한 메시지1의 해쉬값과 비교한 뒤 무결성을 검증한다.



[그림 3] 침입차단시스템에서의 암호화

5.결론 및 향후 과제.

본 논문에서 제안하고 있는 안전한 메일 시스템은 시스템 보안 목적의 침입차단 시스템과 네트워크 보안 목적인 암호화 시스템을 연관 시켜서 메일 시스템의 보안을 보다 향상시켜준다.
유닉스 메일시스템의 기본적인 mailer인 Sendmail 이 root 환경에서 실행되는 이유로 보안상으로 많은 문제점을 가지고 왔다. 물론 패치들도 꾸준히 올라오고 있지만 본 논문에서는 보다 근본적으로 해결하기 위해서 root가 아닌 일반 uid로 수신되는 메일을 처리하도록 하고 있다. 또한 메시지 크기, 수신자 수의 제한, 송·수신 ID,호스트 별 접근통제, 파일 첨부 통제, 로그 기록 등 침입차단시스템으로 있어야 할 기능도 설계 및 구현 되었다.
메일 메시지 자체를 보호하기 위한 네트워크 보안에 있어서 메일을 송수신 할 때 침입차단시스템은 중간에서 감시를 할 수 있어야 하기 때문에 자체적으로 암호화 복호화를 해야 한다. 이는 많은 부하를 줄 수 있기 때문에 키분배 서버를 두어서 해결하도록 한다.
위에서 제안한 시스템 보안, 즉 침입차단시스템은 설계 및 구현이 끝난 상태이지만 네트워크 보안인 침입차단시스템에서의 암호화 시스템의 설계와 구현은 마무리 중에 있다. 앞으로 남은 과제는 위의 구현을 모두 끝내고 모의 실험을 통해 성능을 분석하고 평가하는 것이다.

참고 문헌

[1] Simson Garfinkel, Gene Spafford, "Practical Unix & Internet security," O'Reilly & Associates,Inc, 1996.
[2] D.Brent Chapman, Elizabeth D.Zwichy, "Building Internet Firewalls," O'Reilly & Associates,Inc, 1995.
[3] J.Postel, "Simple Mail Transfer Protocol," RFC 821, 1982
[4] B.Kaliski, "Privacy Enhancement for Internet Electronic Mail, Part4 : Key Certificate and Related Services," RFC 1424, 1993.