패킷 필터링에서 명령어와 인자 결합 규칙을 이용한 로그 데이터의 감축 방법

서현진[O], 박성인, 이재영
한림대학교 컴퓨터공학부

A Method to Reduce Log Data Using the Rule to Combination Commands and Arguments in Packet Filtering

H.J.Seo[O], S.I.Park, J.Y.Lee
Division of Computer Engineering, Hallym Univ.

요 약

UNIX 시스템에서 로그 시스템은 공격시 쉽게 변경 및 삭제되는 위험성이 있고 제한된 시스템 및 네트워크 정보를 제공하므로, 보다 안전하고 풍부한 정보의 제공을 위해 패킷 필터링을 이용한 로그 시스템 등이 제안되어 왔다. 그러나 기존의 패킷 필터링을 이용한 로그 시스템에서는 모든 패킷을 기록하여 많은 량의 데이터가 발생하였으므로, 관리자가 그 정보를 분석하기란 어려웠다. 본 논문에서는 패킷을 처리하는 과정에서 각종 유형의 침입에 대한 사전 조사와 분석으로 얻은 명령어와 인자들의 결합에 의한 판정 규칙을 적용하여, 위험 가능성이 내재된 패킷만을 수집, 기록함으로서 데이터의 양을 줄이고 보다 효율적인 로그 정보를 기록할 수 있었다.

1. 서론

최근 들어 전자 상거래, 인트라넷, 클라이언트, 클라이언트/서버 등과 같은 개방된 인터넷을 이용한 각종 수익 사업이 이루어지고 있으며, 인터넷을 통해 중요한 정보의 이동이 급격히 확산되고 있다. 특히 인터넷의 발전과 더불어 개방 지향적인 유닉스 운영체제를 사용하는 컴퓨터의 보급이 확산되고, 정보의 개방을 통한 각종 해킹 기술의 접근이 쉬워지면서 인터넷에서의 불법적인 시스템 크래킹이나. 해킹을 통한 정보 유출 및 불법적인 사용 등이 증가하고 있다[1, 2].

UNIX 시스템들은 login, logout 정보 및 사용자의 시스템내의 행위를 기록하기 위해 로그 파일들을 사용하며, 로그 파일들은 보안 영역에서 중요한 역할을 차지한다. 관리자는 로그 파일들을 이용함으로서 시스템에서 나타날 수 있는 버그의 원인이나 침입시 침입에 대한 정보, 피해의 정도까지도 파악할 수 있다. 또한 로그 파일들은 시스템을 복구하는데도 필요할 것이다. 그러나 로그들은 시스템 자체에 기록되므로, 침입시 가장 우선적으로 공격받아 쉽게 변경 및 삭제될 수 있으며, 또한 공격을 받지 않더라도 제한된 정보만을 기록하는 로그의 특성 때문에 침입시 많은 정보를 제공하기 어렵다[3].

따라서, 이러한 UNIX 로그 시스템의 문제점을 보완하기 위해 Mike Accetta와 Rick Rashid가 1980년에 제안하여 발전된 기술인 패킷 캡쳐와 필터링 기술을 이용 네트워크 상에서 특정 호스트에 전달되는 패킷을 가로채어 패킷들 안의 정보들을 모아 기록하여 기존의 로그 시스템을 보완하는 방법 등이 제시되었다[7]. 그러나, 이 경우에는 패킷 필터링을 통해 많은 양의 데이터가 발생되고, 그 데이터를 통한 효율적인 정보 추출이 어려웠다.

본 논문에서는 시스템 자체가 패킷을 처리하는 과정에서 각종 유형의 침입에 대한 사전 조사와 분석으로 얻은 명령어와 인자들의 결합에 의한 판정 규칙을 적용하여, 위험 가능성이 내재된 패킷만을 수집, 기록함으로서 데이터의 양을 줄이고 보다 효율적인 로그 정보를 기록하는 새로운 시스템 구성을 제안하고자 한다.

2. 패킷 필터링 로그 시스템

패킷 필터링 로그 시스템은 침입시 직접적인 공격을 피하기 위해 로컬 네트워크 상에 임의의 신뢰적인 시스템에 위치하고, 특정 서버에 전송되는 패킷들을 감시 및 분석을 하게 된다.

패킷 필터링 로그 시스템은 대략적으로 아래 그림과 같이, 로컬 네트워크 상에서 특정 서버 및 특정 포트에 전송되는 모든 패킷을 감시, 수집하는 패킷 수집 모듈, 위험 가능성이 내재한 패킷을 판정하는 패킷 내용 분석 모듈, 위험 가능성이 내재된 패킷으로 얻은 정보로 그 패킷의 접속 IP와 접속 포트로 접속한 사용자의 지속적인 패킷 상태를 기록하는 로그 기록 모듈로 구성되어진다.

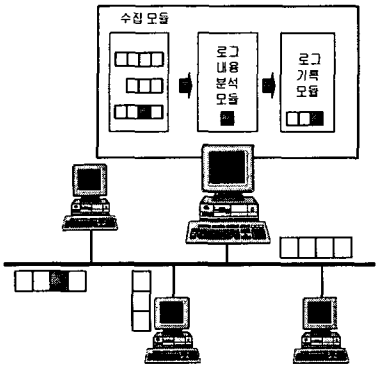

그림1. 패킷 필터링 로그 시스템 구성

2.1 네트워크 패킷 수집 모듈

네트워크 패킷 수집 모듈은 로컬 네트워크 상에서 패킷 송수신에 관련된 물리적인 장치의 유무를 확인하고 장치 상태 점검과 네트워크 접속 상태 확인하여, 네트워크 상에 있는 모든 패킷들을 감시 및 분석하게 된다.

이를 위해서 패킷 수집 모듈이 설치된 시스템의 호스트의 네트워크 상태는 모든 패킷을 수신할 수 있도록 Promiscuous mode로 되어있어야 한다.

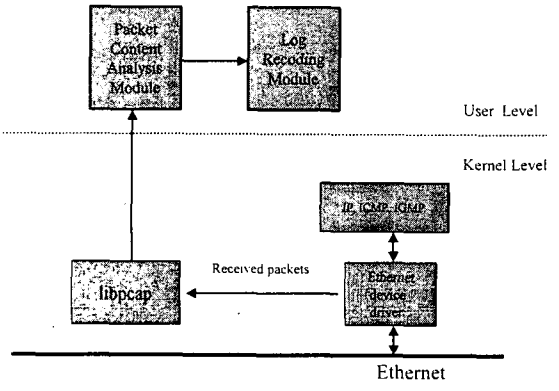

그림2. libpcap의 동작 구조

이러한 기능을 구현하기 위해서 시스템에 관계없이 사용자 레벨에서 패킷에 대해 처리할 수 있도록 프로그래밍할 수 있도록 해주는 libpcap를 사용하여 하위 계층에서 모든 패킷들을 수집 가능하게 하였다.

telnet 프로토콜의 경우 입력된 명령을 라인단위가 아닌 문자단위로 전송하게 되므로, 각각의 연결에 대하여 버퍼를 두어 순서대로 정리하게 되며, 정리된 문자열은 CR코드 기준으로 나누어진다.

2.2 패킷 내용 분석 모듈

네트워크 패킷 수집 모듈에서 발생된 데이터들은 분석에 어려움이 있으므로, 시스템 자체가 패킷을 처리하는 과정에서 위험 가능성이 내재된 패킷만을 추출하는 것이 중요한 일이다. 특정 서비스에서 전송을 위해 사용되는 패킷의 헤더 부분을 제외한 실제 데이터가 삽입된 부분을 분석하여 위험 가능성을 판단할 수 있다.

이러한 위험 가능성에 대한 판단은 일반 사용자가 사용자와는 달리 침입시 자주 사용되는 시스템 및 네트워크 정보에 관련된 명령어나 디렉토리 및 파일 정보에 대한 프로파일로 구성하여 이것을 사용자의 행위와 비교한다. 프로 파일에 대한 기본 설정은 아래 표와 같이 각종 유형의 침입에 대한 사전 조사와 분석으로부터 침입 유형들에 대한 특징을 발견하고 이를 바탕으로 데이터를 크게 명령어와 인자로 구분하고 각각에 대하여 수행되는 특징을 중심으로 세분화하였다.

표1. 명령와 인자의 결합 판정 규칙

Command	Argument
ReadCmd	FReadArg DReadArg
WriteCmd	FReadArg DReadArg FWriteArg DWriteArg
DeleteCmd	FReadArg DReadArg FDeleteArg DDeleteArg
ModeCmd	FReadArg DReadArg FModeArg DModeArg
SuCmd	관리자 권한의 명령
InfoCmd	시스템 정보과 관련 명령
AvaCmd	침입에 사용가능한 명령

명령어와 인자들에 의한 각각의 검색 모듈을 구성하고, 이를 바탕으로 먼저 명령어 모듈과 비교하고, 다음에 인자 모듈과 비교하여 두 조건에 모두 만족하는 경우에 위험 요소로 판단하게 된다.

명령어 검색 모듈에는 사용자들에 의해 자주 사용되고 침입과는 직접적인 관계는 없으나 파일이나 디렉토리 조작에 있어 침입의 영향을 미치는 명령어, 현재 사용중인 시스템이나 네트워크의 정보를 제공하는 명령어들로 침입시도에 앞서 사전 정보를 얻기 위해 공격자들에 의해 자주 사용되는 명령어, 관리자 권한을 가진 사용자만이 사용할 수 있는 명령어들을 구성하였다. 또한 인자 검색 모듈에는 로그 파일과 로그 파일이 위치한 디렉토리, Root의 권한으로 동작되는 프로그램만 read가 가능한 파일들, 일반 사용자들에 대한 write permission 없는 디렉토리나 파일들로 구성하였다.

2.3 로그 기록 모듈

로그 기록 모듈에서는 패킷 내용 분석 모듈에 의해 위험 가능성을 내재한 정보에 전송 받게 된다. 이 경우에 전송 받은 패킷 뿐 아니라 이 패킷의 접속 포트와 동일한 이후의 패킷에 대해서도 계속해서 기록을 하게 된다. 또한 기록할 경우에는 연속적인 방향키 등과 같은 중요하지 않은 정보들은 삭제하고 정리하여 로그 파일에 기록하게 된다.

이러한 새로운 형태의 로그들은 불법적인 시스템 사용자에 대한 경고나 침입자에 대한 침입 가능성 또는 침입시에 중요한 정보를 제공하여, 역추적 시스템 등을 이용한 침입자 추적에 이용될 수 있다.

3. 구현 및 평가

패킷 필터링을 이용하여 얻어진 데이터들은 CR 코드 기준으로 각각의 버퍼에 존재하고, 이 데이터들은 토큰 단위로 나우어 진다. 이때, UNIX 시스템의 명령어 형태가 명령어 이름, 인수 순서로 구성되어 있으므로, 첫 토큰을 명령어 이름으로 간주하고, 두 번째 토큰을 인자로 간주할 수 있다.

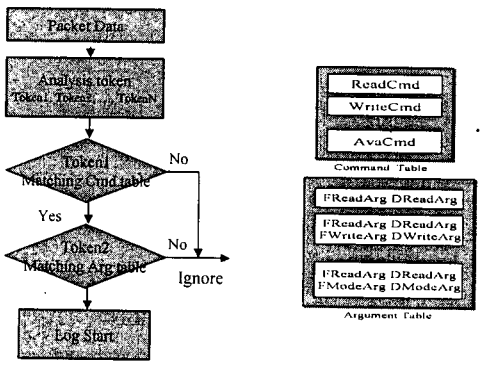

그림3. 패킷 분석 및 로그 기록

그리고, 명령어 검색 모듈과 인자 검색 모듈은 모두 검색이 실시간으로 이루어져야 하는 특성상 위험 요소 판단 근거의 명령어들과 인수들을 해싱 테이블로 구성하여 효율적으로 검색할 수 있게 하였다. 그리고 첫 토큰을 명령어 검색 모듈에서 검색하고 명령어 검색 모듈과 일치하는 명령어가 있을 경우, 그 명령어에 대한 인자 검색 모듈에서 인자 검색 모듈에서 검색을 하게 된다. 그리고 두 검색 모듈에서 동시에 만족하는 경우에 한하여 불법 가능 포트로 판단되어 그 패킷의 접속 IP와 포트를 사용하는 사용자의 행위를 감시하기 위해 포트로 전송되는 모든 패킷들을 계속해서 기록하게 되며, 검색 모듈에 일치하지 않는 나머지 패킷 데이터는 폐기하게 된다.

그림4는 제안된 시스템과 기존의 시스템을 일주일동안 비교 실험 결과로, 패킷 내용 분석 모듈 없이 모든 패킷을 기록하던 기존의 시스템에 비하여 제안된 시스템의 로그의 양이 약 95%가 줄어들었음을 볼 수 있다.

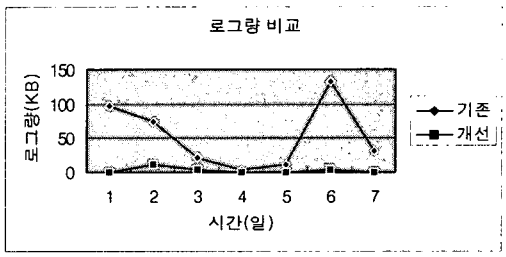

그림4. 로그량 비교

이러한 개선된 시스템의 로그는 명령어 검색 모듈 및 인자 검색 모듈에 있어서 판정 기준의 엄격한 제약으로 인하여 일반 사용자들의 활동까지도 위험요소로 간주하고 기록하는 경우가 발생하였다.

4. 결론

기존의 패킷 필터링을 이용한 로그 시스템에서는 패킷 필터링을 통해 발생한 패킷 정보를 그대로 기록하여 대량의 로그정보가 발생하는 문제점이 있었다.

이러한 문제점을 해결하기 위하여 개선된 시스템에서는 패킷 수집 모듈을 통하여 수집된 패킷들의 내용을 명령어와 인자의 결합 규칙을 이용, 분석하여 위험 가능성이 있다고 판단되는 연결만을 기록하므로 대량의 로그 데이터 발생을 줄일 수 있었다. 또한, 관리자가 쉽게 로그 정보를 이용하여 시스템들에 대한 감사가 이루어 질 수 있도록 하였다.

참고문헌

[1] 한국 정보 보호 센터, 불법 침입자 실시간 역추적 시스템 개발에 관한 연구, 1998

[2] 한국 정보 보호 센터, 실시간 네트워크 침입 탐지 시스템, 1998

[3] S.Garfinkel & G.Spafford, Practical UNIX & Internet Security, O'Reilly Associates, 1996

[4] D.E.Denning, "An Intrusion-Detecion Model", IEEE Transation on Software Engineering Vol. SE-13, No.2, Feb 1987 222-232

[5] 한국 정보 보호 센터, 실시간 LAN기반 패킷 모니터링 개발, 1998

[6] Andrew P. Kosoresow, Steven A. Hofmeyr, "Intrusion Detection via System Call Traces", IEEE Software, pp 35-42, September/October 1997

[7] 박성인, 서현진, 이재영 "패킷 필터링을 이용한 UNIX 로그 시스템의 구현에 대한 연구", 한국통신학회 Vol.19.No.2, 1999