主題

센서 기반 침입 탐지 및 대응 기술

대구가톨릭대학교 대학원 박사과정 장 정 숙
대구가톨릭대학교 컴퓨터정보통신 공학부 교수 전 용 희

차 례

1. 서 론

센서(sensor)의 원래 의미는 측정 대상물로부터 정보를 감지하여 측정량을 전기적인 신호로 변환하여 주는 장치로써 그 응용 분야가 광범위하다. 최근에 신호처리 기술과 MEMS(Micro Electro Mechanical Systems) 기술을 이용한 미세 가공 기술이 발전함에 의하여 센서를 정보통신 분야에 이용하고자 하는 응용 기술 개발이 많이 진행되고 있으며, 이미 산업 현장의 자동화 분야, 의료 분야, 우주 항공 분야, 대기 및 수질 오염 측정을 위한 환경 분야, 군사 분야 등 다양한 분야에 걸쳐서 정밀 계측 및 자동화에 필수적으로 사용되고 있다[1, 2].

시큐어넷 센서 사용자 가이드[3]에 의하면 센서는 엔진(engine)이라고도 하며, 고속 네트워크 상의 대량의 트래픽을 감시하기 위한 설치 가능한 소프트웨어 혹은 어플라이언스(appliance)-기반 기술이다. 센서는 네트워크의 특정 위치에 놓여진다. 센서는 처리기-집중 장치이며 일반적으로 정확하게 동작하기 위하여 자체적인 PC나 어플라이언스를 요구한다. 어플라이언스는 목적에 따라서 입출력(I/O) 선택 사항, 처리 속도 및 메모리가 다른 특정 컴퓨터이다. 센서는 침입 증거를 찾기 위하여 모든 네트워크 트래픽을 분석하여 네트워크 침입 탐지 시스템(IDS: Intrusion Detection System) 정책의 매개변수에 따라서 매니저에게 정보를 보고한다.

한편 센서는 네트워크 침입에 능동적으로 대처하기 위한 소프트웨어 모듈로써, 이동 에이전트의 특성인 이동성을 지니고 있고 액티브 패킷 내의 수행 가능한 코드 형식으로 전달되는 소프트웨어로 정의하고 있다[4, 5]. 이에 의하면 센서를 이동 범위에 따라서 고정형 센서(stationary sensor)와 이동형 센서(mobile sensor)로 나누고, 수행 위치에 따라서 호스트 센서(host sensor)와 네트워크 센서(network sensor)로 다시 분류하고 있다. 이동형 센서는 생성되어 소멸될 때까지 여

러 노드들 사이를 자유롭게 이동하며 실행하는 센서이고, 고정형 센서는 생성되어 특정 노드로 이동된 후 그 노드에서 소멸할 때까지 상주하는 센서이다. 호스트 센서는 호스트에 상주하는 센서를, 네트워크 센서는 네트워크 노드에 존재하는 센서로 분류한다. 센서를 [4]에서와 같이 소프트웨어 모듈로 본다면, 에이전트 개념과도 유사하다. [6]에서 에이전트는 호스트 기반 침입탐지 시스템에서 특정 PC에 설치된 소프트웨어로 정의하였다. 즉 네트워크 기반 IDS에서 네트워크 트래픽 감시를 위한 장치는 센서로 정의하고, 반면 호스트에 상주하며 특정 파일이나 로그를 감시함으로써 IDS 기능을 수행하는 소프트웨어를 에이전트로 구별한다. ISS(Internet Security Systems)의 RealSecure[7]에서는 호스트 기반 에이전트를 센서에 포함시키고 있다. 그러므로 센서와 에이전트를 별도로 구분할 필요는 없다고 보여 진다. 현재 대부분의 침입 탐지 시스템은 센서를 기본 탐지기로 사용하고 있다. 본 논문에서는 [3]의 정의와 같이 센서를 소프트웨어 혹은 하드웨어로 본다. 그러나 센서가 특정 의미로 사용될 때에는 해당 부분에서 분명히 하기로 한다.

센서는 완전한 독립 감시 도구이며 모든 경보와 다른 기록 정보를 지역 하드 드라이브에 저장하며, 실시간으로 경보를 발생한다. 센서와 매니저 사이의 통신은 DES(Data Encryption Standard)와 같은 암호 기술을 사용하여 안전한 터널을 사용한다. 각 매니저는 한 개의 프로세서 시스템에서 수십 개의 센서들을 수용할 수 있다. 모든 센서들이 특정 위치에 있는 것을 보증하기 위하여 혹은 특정 형태가 동일한 정책을 적용받기 위하여 논리적으로 함께 그룹 될 수 있다. 이것을 트리 안에서 센서 그룹이라 한다.

본 논문에서는 센서 기반 침입 탐지 및 대응 기술에 대하여 고찰하고자 한다. 침입(intrusion)은 컴퓨터가 사용하는 자원의 기밀성(confidentiality), 무결성(integrity), 가용성(availability)을 저해하는 일련의 행위들의 집합 또는 컴퓨터 시스템의 보안정책(SP: Security Policy)을 파괴하는 행위로 규정한다. 침입 탐지는 유형에 따라 비정상 탐지(Anomaly Detection), 오용 탐지(Misuse Detection) 등으로 구분한다. 일반적으로 접근 시 정해진 모델을 벗어나는 경우를 탐지하는 것을 비정상 탐지라 하며, 침입이라고 정해진 모델과 일치하는 경우를 오용 탐지라 한다. 또한, 웹 서비스와 같은 호스트에 설치되어 설치된 호스트만을 대상으로 침입탐지를 하는 것을 호스트 IDS라고 하며 일정 부분의 네트워크 전체를 대상으로 공격 트래픽을 분석함으로써 침입탐지를 하는 것을 네트워크 IDS(NIDS: Network Intrusion Detection System)라고 한다. NIDS는 서비스 거부(DOS: Denial of Service) 공격을 포함하여 광범위한 공격을 탐지할 수 있고, 내향 및 외향 네트워크 트래픽을 감시하는데 효과적이다.

2. 관련 연구

2.1 시스템 요구 사항

센서는 실시간으로 네트워크 트래픽을 디코딩하고 분석하는 계산 집중적인 일을 수행하며, 많은 양의 로그 데이터를 생성할 수 있다. 그러므로 센서를 설치하기 위한 하드웨어 및 소프트웨어 플랫폼의 선정에 유의하여야 한다. 센서에 대한 대표적인 시스템 요구 사항은 표 1과 같다[3].

<표 1> 센서 시스템 요구사항의 예

요구 사항	기술
RAM 크기	최소 128 MB(256 MB 추천)
디스크 공간	최소 265 MB
네트워크 인터페이스 카드(NIC)	두개의 100 Mbps 이상 카드
처리기	500 MHz 이상의 CPU
운영체제	리눅스 7.x
커널	2.4.7-10 혹은 신규

센서를 설치하기 위해서는 센서가 설치될 컴퓨터의 IP(Internet Protocol) 주소, 감시되는 IP 주소의 범위 등이 필요하다.

2.2 네트워크 토폴로지

센서는 무차별(promiscuous) 모드에 의하여 네트워크를 통하여 전송되는 모든 데이터를 포획한다. 다음과 같은 네트워크 토폴로지가 가능하다: 성형, 버스, 경계(border) 네트워크

1) 성형 네트워크 토폴로지

성형 네트워크 토폴로지에서는 중앙에 있는 허브 혹은 교환기의 기능에 따라서 센서의 연결 위치가 틀리게 된다. 교환 능력이 없는 경우, 센서는 네트워크 허브 상의 오픈 포트에만 연결된다: 무차별 모드 인터페이스 상으로 동작하는 센서는 교환 네트워킹의 경우 특정 호스트로 향하는 데이터만 전송하기 때문에 문제가 발생한다. 이 문제로 이러한 환경에서는 워크어라운드(workaround)가 사용되어야 한다. 가장 많이 사용되는 방법은 "모니터 포트"를 지원하는 네트워크 허브 혹은 교환기의 사용이다. 이러한 허브를 흔히 "Managed 허브" 혹은 "Monitored 허브"라고 한다. 모니터 포트로 부착된 호스트로 전송되는 트래픽을 선별적으로 선택할 수 있다.

2) 버스 네트워크 토폴로지

버스 네트워크에서는 센서의 설치가 단순하다. 모니터링 호스트를 네트워크의 어느 위치에도 둘 수 있어, 융통성이 있다.

3) 네트워크 경계 설치

근거리 통신망(LAN)과 인터넷과 같은 외부 소스 사이의 통신을 모니터링하기 위하여 사용될 수 있다. 이 경우 보호되는 LAN 상의 호스트 사이의 트래픽을 감시하는 것이 아니라, 보호되는 LAN과 보호되지 않는 LAN 사이의 전송만을 감시하게 된다. 이 방법을 사용하면, 외부 공격에 대해서만 호스트를 보호할 수 있다. 보호되는 네트워크의 경계로 진입하고 떠나는 전송만을 감시하기 때문에, 지역적인(local) 침입에 상관하지 않고 외부 공격으로부터의 보호에 관심이 있을 때 유용하다. 그러나 다른 종류의 센서가 요구되는 상황이 존재한다.

2.3 전역 필터링(global filtering)

전역 필터링은 중요하지 않는 패킷들을 여과하는 반면 분석을 위하여 흥미 있는 네트워크 패킷들을 지정하게 한다. 이렇게 함으로써 센서가 처리해야 하고 매니저에게 전송해야 할 패킷의 양을 최소화함으로써 센서의 성능을 증가할 수 있다. 필터링은 센서 스크립트에 의하여 패킷들이 분석되기 전해 발생한다. 다음의 매개변수에 기반을 두고 네트워크 패킷을 여과할 수 있다.

- 이더넷 MAC 주소: 주어진 패킷의 소스, 목적지 MAC 주소가 명시된 적용 가능한 MAC 주소와 비교된다. 더 많은 처리가 허용되면, 패킷은 이더넷 패킷 데이터에 기반한 분석을 수행하는 센서 컴포넌트로 보내진다.
- IP 주소: 주어진 패킷의 소스, 목적지 IP 주소가 명시된 적용 가능한 IP 주소와 비교된다. 더 많은 처리가 허용되면, 패킷은 IP 패킷 데이터에 기반하여 분석을 수행하는 센서 컴포넌트로 보내진다.
- 포트 번호: 주어진 소스, 목적지 TCP 혹은 UDP 패킷의 포트 번호가 명시된 적용 가능한 포트 번호와 비교된다. 더 많은 처리가 허용되면, 패킷은 상위 레벨 프로토콜에 기반하여 분석을 수행하는 센서 컴포넌트로 보내진다.

3. 침입 탐지 및 대응 기술

사실상의 모든 침입 탐지 시스템은 유사한 구조를 가지고 있다. 한개 이상의 데이터 소스들을 감시하는 센서가 있고, 침입 탐지 알고리즘 및 대응 메커니즘이 있다. 또한 보안 관리자가 침입 데이터를 감시하고, 배열하고, 분석하기 위한 관리 시스템이 존재한다. 이런 컴포넌트들이 같은 장치 내에 존재할 수도 있고 그렇지 않을 수도 있다 그리고 전부다 존재하지 않을 수도 있다[7].

3.1 초기 탐지 기술

초기 침입 탐지 시스템들은 데이터 소스로 운영 체제 로그 파일들을 사용하였다. 이런 시스템들은 입력 로그에 대하여 단순한 패턴 매칭(pattern matching)을 수행하였다. 패턴은 보통 ASCII 스트링 혹은 스트링 프래그먼트들이고, 패턴 테이블은 침입자들이 시스템에 침투를 시도하는 알려진 방법을 나타내었다. 로그 파일은 흔히 저장소에 저장되어 추후 분석된다. 이것은 로그 파일들의 더욱 완전한 분석을 허용하지만, IDS의 대응 능력을 제한하였다.

이러한 방법이 네트워크에 그대로 사용되었다. 초기 네트워크 침입 탐지 시스템은 네트워크 상에 지나가는 모든 패킷을 알려진 공격 스트링의 목록과 비교한다. 스트링은 ASCII이거나 각 스트링은 센서에게 보이는 모든 트래픽과 바이트별로 비교되기도 한다. 이 접근은 구현이 용이하고 시스템과 네트워크 관리자의 업무를 빠르게 자동화하는 한 방법이 되었다. 그러나 이 방법은 확장성에 문제가 있다. 패턴 수와 데이터 소스 량의 증가는 지수적인 처리력(processing power) 증가를 필요로 한다. 게다가 초기 패턴 알고리즘들은 비교적 정교하지 못하여 침입 탐지 시스템들이 잘못된 긍정(false positive)을 야기 시킨다.

3.2 프로토콜 인식(protocol awareness) 기술

다음 기술은 패킷과 프로토콜의 지식을 네트워크 트래픽에 적용하는 것이었다. 패킷에 대한 지식이 어떤 형태의 행위가 악성인지 알게 한다. 패킷들이 프로토콜 표준을 지키는지 검정함으로써 의심스러운 행위를 보고할 수 있다. 이런 예가 "Ping of Death"이다. 게다가 프로토콜 헤더의 디코드는 패턴 매칭이 수행되는 방법의 개선을 가져다주었다. 예를 들어, 포트 25가 SMTP 트래픽을 위한 잘 알려진(well-known) 포트이기 때문에, 포트 25로 향하는 패킷에 대하여는 메일 서비스에 대한 공격만 분류함으로써 성능 개선을 가져올 수 있다.

다른 발전적인 기술은 행동(actions)을 고려한 것이다. 예를 들어, 어떤 활성 서비스에 대한 단일 포트 조사(probe)에는 별관심이 없지만, 한 호스트에서 다른 호스트로 다량의 포트를 포함하는 많은 수의 probe는 정보 수집 공격으로 간주될 수 있다.

위의 것을 종합하여 이런 메커니즘들이 처음으로 침입자 행위의 포괄적인 개관을 제공하였다. 그러나 시간이 지남에 따라 침입자는 더욱 정교하여 졌다. 한 예는 단편화(fragmentation)라고 부르는 프로세스이다. 공격자는 각 패킷을 더 작은 조각으로 나눈다. 완전한 패턴이 단일 패킷에 없기 때문에 IDS에 의하여 공격이 보이지 않는다. 희생(victim) 네트워크는 이런 패킷들을 재결합하고 침해가 성공하게 된다. 이 행위를 탐지하기 위하여, 단편화 재결합이 IDS에 추가되었다. 트래픽이 센서를 통과할 때 어떤 패킷 단편도 보관되고, 재결합되고, 그리고 평가된다.

3.3 세션-기반 공격 대응 기술

단일 패킷 분석을 넘어 IDS 기술은 세션-기반 공격에 대한 대응을 개발하였다. 이런 형태의 공격을 효과적으로 보기위하여, 스트림 재결합이 필요하다. 스트림 재결합은 연결에 대하여 보관

되는 또 다른 형태의 상태이다. 이 상태는 소스와 목적지 사이의 완전한 교환이 고려되는 것을 허용한다. 이렇게 함으로써 스트림 재결합은 IDS가 대화의 각 부분을 볼 수 있고 악성 행위에 대하여 충분히 검토하게 된다. 이런 상태-기반(state-based) 평가는 일반적으로 CPU 및 메모리 집중 처리로 간주되어 실시간 IDS에는 비경제적인 것으로 생각되어 왔으나, 적절한 구현으로 경제적인 구축이 가능한 것으로 발견되었다.

3.4 완전 프로토콜 분석

프로토콜에 대한 구체적인 지식과 각 프로토콜에 고유한 단계별 프로세스에서 발견된 요소들을 적용함으로써 상당한 효율을 가지고 알려진 나쁜 행위를 탐지할 수 있을 뿐만 아니라 어떤 형태의 비정상 행위를 의심스러운 것으로 플래그(flag)하여 그들이 공포되기 전이라도 새로운 공격 기술을 잡을 수 있다.

시그너처 기반 패턴 매칭의 근본적인 문제점은 회피자가 어떤 방법으로든 그의 공격을 수정 가능하고 그래서 시그너처를 IDS가 찾으려고 하는 것과 매치가 되지 않도록 할 수 있다. IDS 제공자는 기법에 대한 더 많은 변형을 추가하여 대응하게 되고, 공격자는 침입을 회피하기 위하여 새로운 더 많은 방법을 찾게 된다. 완전 프로토콜 분석은 주어진 요소가 목적지에 의하여 어떻게 번역될지를 식별하기 위하여 프로토콜에 대한 지식을 적용하기 때문에, 공격의 모든 변형이 하나의 메커니즘을 통하여 식별될 수 있다. 그리하여 탐지와 대응을 크게 단순화 시킨다.

한 가지 예로 특별한 포맷의 명령을 cgi-bin 스크립트에게 전송하는 HTTP 공격이 있다. 시그너처 기반 IDS는 공격을 식별하기 위하여 어떤 특정 패턴을 매치하여야 한다. 경로와 파일명의 일부로 UNICODE를 사용함으로써 같은 스트링의 많은 변형이 생성가능하다. 만약 IDS가 그 변경의 각 부분을 식별하고 번역할 수 있다면, 공격자에 의하여 어떻게 전송되는지에 관계없이 공격은 항상 동일한 것으로 식별될 것이다. 이 기술이 없이 이런 변형을 정확하게 식별하는 것은 극도로 어렵다.

진보된 프로토콜 분석의 다른 혜택은 공격 방법을 예측하기 위하여 사용될 수 있다는 것이다. 어떤 변경의 특정 부분이 일정 길이보다 적을 때, 그 프로토콜에 대하여 공격을 전송하려는 시도가 비정상으로 탐지될 수 있다. 버퍼 오버플로는 가변 길이를 적절히 조사하지 않는 프로그램을 이용하려는 시도에 의하여 구현되는 공격 방법이다. 많은 환경 변수들은 정상적으로 일정한 길이 안에 있어야 한다. 이 양을 초과하는 어떤 필드는 버퍼를 오브플로하기 위한 시도일 것이고 특별한 버퍼 오브플로 공격을 위한 특정 시그너처 없이도 진보된 프로토콜 분석에 의하여 탐지될 것이다. 현대적인 IDS는 포트 번호와 관계없이 주어진 세션이 사용하고 있는 프로토콜을 탐지할 수 있어 모든 가능한 공격이 보여 질 수 있다.

4. 사례 연구

Internet Security Systems 사는 초기에 상용 네트워크 침입 탐지 시스템을 생산한 업체 중 하나이다. 본 장에서는 ISS RealSecure 7.0을 중심으로 기술한다[8].

RealSecure 7.0은 기존 제품과 새로운 NIDS 센서 기술을 통합하여 만든 제품이며, 다음과 같은 컴포넌트로 구성된다.

4.1 센서의 종류

가. 네트워크 센서

이 센서는 네트워크 세그먼트를 감시하기 위

하여 전용 호스트 상에서 운용되며, 트래픽 플로우를 분석하고 침입과 네트워크 오용 흔적을 조사한다. 침입이 탐지되면, 센서는 아래와 같은 방법을 포함하여 여러 가지 방법으로 응답한다.

- 이벤트 일자, 시간, 소스 및 목적지 기록
- 이벤트의 내용 기록
- 네트워크 관리자에게 통보
- 방화벽 재설정
- 이벤트 자동 종료

이 센서는 HTTP, FTP, SMTP, SNMP, RPC, SMB, NetBios와 같은 60가지 이상의 응용 계층 프로토콜을 인식한다. 이 센서는 콘텍스트를 인식하기 위하여 프로토콜 분석에 초점을 맞추고 악성 콘텐츠를 적극적으로 식별하기 위하여 적절한 곳에 패턴 매칭(pattern matching) 기술을 채용한다. 또한 TCP와 HTTP 세션 재결합과 완전한 IP 패킷 de-fragmentation을 수행할 수 있다.

나. OS 센서

운영 체제(OS) 센서는 커널-레벨 이벤트, 호스트 로그, 중요 서버상의 네트워크 활동을 분석함으로써 실시간 침입 감시, 탐지 및 악성 행위를 방지하는 호스트-기반 컴포넌트이다. 정책 위반이 탐지되면, 센서는 아래 사항을 포함하여 여러 가지 방법으로 응답할 수 있다:

- 모든 이벤트 정보 기록
- 일자 및 시간 기록
- 계정 유보 혹은 정지
- 네트워크 관리자에게 통보
- 침입이 인식되었다는 경고를 침입자에게 텍스트 메시지 전송

다. 서버 센서

서버 센서는 OS 센서 기능을 포함하고 네트워크 트래픽 감시, 지능적인 경보, 차단 능력을 가진다. 서버 센서는 운영체제에 도착하기 전에 의심스러운 트래픽을 차단하고 패킷을 포획한다. 양방향 통신을 감시하고 제어할 수 있다.

4.2 매니저 설치 환경

매니저의 설치 환경은 다음과 같이 3가지가 있다: 안전한(secure) LAN, 신뢰되지 않는(untrusted) LAN 및 동일 호스트 상에 설치. 각 방법은 각각 다음과 같은 장단점이 존재한다.

1). Secure LAN

전역 매니저가 공중 LAN에 설치되는 것이 아니라, 신뢰되지 않은 LAN과 직접 링크가 없는 안전한 사설망에 위치한다. 이 사설망은 센서와 매니저 사이의 통신을 위하여 단지 사용된다. 이 방법은 관리용 호스트가 외부 망과는 완전히 별개로 유지하게 한다. 사설망을 사용함으로써 감시되는 네트워크 상에서 센서는 "보이지 않게"(invisibly) 작동될 수 있다. 이 방법이 가장 안전하고 효율적인 것으로 간주된다.

2) 신뢰되지 않은 LAN

이 방법은 안전한 별도의 LAN이 구축될 수 없을 때 사용된다. 이 경우 센서와 매니저 사이의 통신이 침해되는 가능성을 감소하기 위하여 3DES(Triple Data Encryption Standard), Blowfish, DES와 같은 링크-레벨 암호화와 메시지 무결성 검증을 사용한다.

3) 동일한 호스트 상에 설치

이 방법은 센서 호스트 상에 직접 매니저 콘솔을 설치하는 것이다. 이 것은 하드웨어 자원이 절약되어야 하는 경우 유용하나, 동일한 호스트 상에 두개의 응용을 실행함으로써 통상적으로 모니터링 성능과 사용자 인터페이스 응답성의 저하를 초래할 수 있다.

4.3 매니저 콘솔

각 콘솔은 다음과 같은 업무를 수행한다.

- 넹령 및 배열: 센서에 변화가 있을 때 마스터 상태를 가지도록 실행한다.
- 정책 편집: 센서 배열, 탐지를 위한 공격의 종류, 대응 방법 등에 대한 정책을 센서에게 하달한다. 정책 편집은 콘솔 내부나 외부 정책 편집기를 사용하여 행하여진다. 변경된 정책은 콘솔에 저장되며 센서에 적용된다.
- 배열 변화: 이것은 관리자에게 시스템 매개변수(네트워크 감시를 위한 네트워크 카드 지정, 센서의 구동 및 정지 등)를 수정하게 한다. 마스터 상태를 필요로 한다.
- 실시간 데이터 분석: 센서로부터 실시간 경보를 받을 수 있다.
- 사고 후 분석: 사고 경향 보고나 조사 분석을 위하여 톱 10 이벤트 관리 보고, IP 주소별 공격 혹은 시그너처 별 공격 등과 같은 보고서를 생성할 수 있다.

정책이 콘솔에서 정해지면 각 센서에 적절히 적용된다. 어떤 세그먼트 상의 기대 트래픽이나 세그먼트의 중요도에 의존하여 필요한 경우 다른 정책들이 각 센서에 적용될 수 있다. 센서가 적용된 정책을 가지고 운용되면, 경보가 센서 상에 지역적으로 기록되고 지정된 데이터베이스 저장소로 실시간으로 전송된다. 콘솔은 새로운 이벤트들을 검색하여 배당된 위험 레벨(severity level)에 따라서 우선순위별로 디스플레이 할 수 있다. 이벤트 또한 소스 주소, 목적지 주소, 혹은 이벤트 기술에 의하여 분류되고 그룹화 된 계층적 행위 트리로 디스플레이 된다.

4.4 보고 및 분석

통상적인 보고서로는 다음과 같은 것이 있다[8].

- 이벤트 이름: 이벤트 이름에 의하여 분류된 이벤트 상세 목록
- 이벤트 우선순위: 우선순위 및 시간에 의하여 분류된 이벤트 상세 목록
- 목적지 IP: 목적지 IP 주소와 시간에 의하여 분류된 이벤트 상세 목록. 이벤트의 목표가 된 호스트 별로 분류된 네트워크 이벤트에 대한 조사를 할 수 있다.
- 톱 20 이벤트: 톱 20 이벤트 발생을 그래프로 보여준다.
- 톱 20 목적지: 각 주소와 연관된 이벤트의 수에 의하여 표시된 톱 20 IP 목적지를 그래프로 보여준다.
- 이벤트 우선순위 빈도 그래프: 명시된 시간 프레임 동안 탐지된 위험도 별 침입 이벤트의 수를 그래프로 보여준다.

5. IDS 성능 평가

침입 탐지 기술이 진보됨에 따라, IDS 평가 기술도 진보되고 있다. 초기 IDS 시절에는 어떤 제품이 식별할 수 있는 알려진 공격의 수가 주요한 차별이었다. 항목에 더 많은 조사를 제공하면 더 좋은 성능을 가진 것으로 간주되었다. 그러나 이러한 관점은 더 이상 유효하지 않다. 진보된 프로토콜 분석 기술로 더욱 효율적인 트래픽 평가가 많은 수의 exploit를 단일 클래스의 악성 행위로 축소할 수 있기 때문에, 한 시스템에서 조사의 수가 사실상 감소될 수 있다. 오늘날 사용될 기준은 이해되는 프로토콜의 수 그리고 각 내부에서 얼마나 많은 파라미터들이 고려되는지가 될 것이다. 본 장에서는 IDS 성능 평가를 위한 지침을 소개하고자 한다[9].

IDS 성능 시험에 관계되는 주요한 특징들은 다음과 같다.

- 구조, 호스트 및 네트워크 기반 센서/에이전트의 설치 용이성
- 배열, 관리의 용이성
- 콘솔과 센서/에이전트 사이의 안전한 통신을 위한 인증 및 암호화
- 정책 정의, 정책 분배 및 정책 변경 절차
- custom 공격 시그너처의 허용성
- 신규 공격 시그너처의 획득 및 배열 방법, 갱신 주기
- 기타: 보고서, 콘솔 인터페이스, 완전 프로토콜 디코드 능력, 통합성, 유지 관리 등

5.1 네트워크 IDS 평가

가. 공격 인식

성능 평가에 무엇보다도 우선적으로 적용 가능한 것이 시스템의 공격 인식 능력이다. 아래와 같은 영역의 공격에 대한 시험이 가능하다.

- 응용 버그, 백 도어, 트로이 목마, DOS, 분산 DOS
- 핑거, FTP, HTTP, ICMP, 이메일, 악성 데이터 입력
- reconnaissance, SNMP, SANS 항목 등

나. 인가 부하 성능

네트워크의 변하는 백그라운드 트래픽 상황에서 침입 탐지 능력을 시험하는 것이다. 다음과 같은 시험이 가능하다.

· 작은(64 바이트 UDP) 패킷

패킷은 유효한 소스/목적지 IP 주소와 포트를 가진다. 이 방법은 IDS 센서의 스니핑 능력을 나타내기 위하여 설계된 혹독한 시험 방법이다. 만약 센서가 이 시험에서 100% 부하에서 100%의 공격을 탐지한다면, 센서에게 주어지는 어떠한 것도 취급 가능함을 말해준다. 그러나 어떤 제품은 개별 패킷 스니핑에서는 좋은 성능을 보여주는 반면, 실제 세션 취급에서는 좋지 못한 경우도 있다. 이 시험은 주로 raw sniffing 속도를 나타낸다. 시험은 트래픽 발생기를 이용하여 각각 25, 50, 75, 100% 네트워크 이용률 부하에서 반복될 수 있다.

· 실제 패킷 혼합

인터넷상의 실제 패킷 크기에 대한 조사를 바탕으로 평균 패킷 크기(예를 들어, 300 바이트)를 결정하고 패킷 분포(예를 들어, 64에서 1,514 바이트)를 이용하여 트래픽을 발생시켜 네트워크에 부하를 인가하는 방법이다. 모든 패킷은 유효한 페이로드와 주소를 포함한다. 이 시험은 다양한 네트워크 부하에서 실제 네트워크를 훌륭하게 표시할 수 있는 방법이다. 각 IDS 센서는 적어도 60% 부하 수준까지는 좋은 성능을 나타내는 것이 기대된다. 시험은 트래픽 발생기를 이용하여 각각 25, 50, 75, 100% 네트워크 이용률 부하에서 반복될 수 있다.

· 대형(1,514 바이트) 패킷

패킷은 유효한 페이로드와 주소 데이터를 포함한다. 이 시험은 작은 패킷 시험의 정반대이다. 이 방법은 대형 패킷을 사용하여 좋은 결과를 얻는 것이 용이하다는 것을 보여준다. 시험은 트래픽 발생기를 이용하여 각각 25, 50, 75, 100% 네트워크 이용률 부하에서 반복될 수 있다.

다. IDS 회피(evasion) 시험

아래와 같은 IDS 회피 기술을 채용하여 공격을 시도하는 시험을 수행하는 것이다. 먼저 기본적인 공통 공격의 부분집합을 수행한다.

- 다양한 크기의 순서화 되거나(ordered), 순서에 어긋나는(out-of-order) IP 프래그먼트
- 중복 프래그먼트
- TCP 단편화 중복

· TCP 및 IP chaffing

다음에 타켓 머신의 기본 WWW CGI 스캔을 수행하여 IDS가 이러한 공격을 탐지하는지 검증할 수 있다. 그런 후, 아래와 같이 포함되는 다양한 IDS 회피 공격을 사용하여 시험을 반복한다.

· URL 인코딩, /./ 디렉토리 삽입, 조기 URL 종료, 긴 URL
· 거짓 파라미터, TAB 분리, case 민감도, 윈도우 \ 경계자, 세션 묶음(splicing)

마지막으로, 아래와 같은 다양한 회피 기술에 변경된 동일한 시험을 수행할 수 있다.

· RPC 기록 fragging, 명령 라인 공간 삽입
· 데이터 스트림에 비 문자 Telnet opcode 삽입
· 단편화, 다형태 mutation

공격을 시행한다. 전반적인 센서 성능과 로깅 능력에 대한 영향이 조사된다. 또한, 많은 수의 동시 연결을 감시할 때 센싱 엔진의 성능을 시험하게 된다.

5.2 호스트 IDS 평가

가. 공격

HIDS 제품의 시험을 위하여 NIDS에서처럼 단순한 벤치마크가 없다. 변하는 네트워크 부하에서의 공격 인식과 성능은 호스트 기반 IDS에 쉽게 적용될 수 있는 척도가 되지 못한다. 이런 이유로 초기 probe로부터 중요 자료의 훔침(stealing)이나 변경에 이르기까지 기본적인 공격 시나리오가 적용될 수 있다[9]. 공격의 여러 단계를 통하여, 대표적인 HIDS에 의하여 무엇이 행해질 필요가 있으며 각 제품이 어떻게 수행되는지 조사하게 된다. 아래 표 1에 공격 시나리오의 예를 보여준다.

<표 1> 호스트 기반 IDS 성능 시험 시나리오의 예

공격자 행동	HIDS 조사 내용
nmap을 이용한 공격자 probe 시스템	포트 스캐닝 시도
brute force 패스워드 추측	다중 실패 로긴
디렉토리 목록 탐지	비인가 사용자의 중요 데이터 파일 접근
패스워드 파일 복구 시도	패스워드 데이터베이스 접근 시도
유효 계정 로긴	모든 로긴 감시
감춰진 트로이/바이러스/공격 도구 설치	등록 키 및 시스템 디렉터리에 신규 실행파일 추가, 실행 파일 수정
호스트에 백도어 생성	특혜 계정 추가, 게스트 계정 실행, 관리자 그룹에 게스트 추가
특권 상승	사용자 권한 변경
추적 감추기	보안 로그 변경, 감사 파일 삭제, 감사 정책 수정
중요 데이터 접근	유효하지 않은 시스템으로부터 공유 데이터 접근
중요 데이터 다운로드 및 수정	FTP 혹은 이메일 경유 데이터 파일 전송, 사용자 인사 데이터 변경 지원
침해된 호스트 이용 네트워크 스캔	nmap과 같은 금지된 프로그램 수행

라. stateful 운용 시험

이 시험에서는 센서가 플러딩 혹은 잘못된 부정에 취약한지를 조사하게 된다. 이를 위하여 유효한 소스와 목적지 그리고 프로토콜들을 사용하여 보호되는 서버넷에 대한 다량의 거짓 경보를 발생한다. 이 공격동안, IDS 센서가 실제 공격을 탐지할 수 있는지를 결정하기 위하여 통상적인

나. Forensic Investigation

좋은 IDS에게는 정확한 손상(damage) 평가를 수행하기 위하여 공격 개시 점까지 추적하여 다시 공격 지점 및 그 이상까지 진진할 수 있어야 한다. HIDS가 침입자의 행동을 가능한 많이 보안 분석가가 결정할 수 있도록 충분한 forensic 자료를 기록해야 한다. 중요한 것은 보고 시스템

이며, 앞 절의 시험에서 공격 후 다음과 같은 보고를 제공하는 능력에 대하여 평가한다.

- 실패 로긴 목록, 특정 시간 프레임내의 전체 로긴 목록
- 중요 데이터 전체 접근 목록
- 시스템 파일에 대한 비인가 접근 목록
- 등록키에 대한 비인가 접근 목록
- 실행 파일 수정 변경 전체 목록
- 감사 정책 변경 전체 목록
- 기타

6. 맺음말

과거 수년 동안 DARPA는 침입 탐지 프로젝트에 많은 지원을 하여 왔다. 일반적인 목표는 이미 알려진 그리고 알려지지 않은 공격에 대하여, 1% 이하의 거짓 경보율을 가지고 모든 공격의 99% 이상을 탐지할 수 있는 시스템을 개발하는 것이었다. 그러나 대부분의 최근 평가 결과에 의하면 이 목표와는 상당한 거리가 있다.

네트워크 데이터 전송율의 증가도 침입 탐지 능력에 상당한 부담을 준다. IDS에서 사용되는 처리 알고리즘에서는 CPU 사이클보다 메모리 사이클이 제한 요소일 가능성이 높다. 메모리 속도는 CPU 클록 속도보다 훨씬 증가 속도가 늦다. 헤더 처리만 하는 경우 기가비트 네트워크 속도까지 가능하지만, 단일 패턴에 대한 단순 패턴 매치도 이용 가능한 바이트 당 메모리 사이클보다 많은 것을 요구하기 때문에 패킷 내용의 실질적인 처리는 초당 50메가비트 이상 불가능한 것으로 보여 진다[10].

네트워크 센서도 또한 공격을 당할 수 있다. 단일 플랫폼 위에 센싱과 분석을 결합하는 것이 대표적이다. 만약 플랫폼이 많은 세션과 관련된 상태 정보를 유지하는 것을 요구하는 스테이트풀(stateful) 분석을 수행한다면, 자원 소모 공격을 당하기 쉽다.

침입 탐지에서 오탐율과 성능 문제를 해결하기 위하여, 주요한 두 가지의 연구 작업이 요구된다. 첫 번째는 "정상" 행위의 특성화이다. 어떤 주어진 환경에서 최적의 비정상 탐지기가 정의될 수 있도록 주어진 환경을 충분히 특성화하는 방법을 개발하는 것이 가능하다. 이런 작업으로 기대되는 거짓 경보율을 미리 결정하고 탐지기의 한계점을 허용할 수 있다. 다른 분야는 개별적인 것이 아닌 클래스의 공격들을 탐지하기 위하여 구체적인 행위를 충분히 함축할 수 있는 탐지기 설계에 사용될 수 있는 적절한 침입 행위 이론을 개발하는 것이다.

국내에서도 침입 탐지 및 대응 기술에 대하여 상용 시스템 기술의 개발과 아울러 기본적인 이론 개발이 또한 절실히 요구된다고 하겠다.

참고문헌

[1] Ian F. Akyildiz et al., "A Survey on Sensor Networks", IEEE Communications Magazine, pp. 102-114, August 2002.

[2] Hairong Qi et al., "Distributed sensor networks-a review of recent research", Journal of the Franklin Institute 338, pp. 655-668, 2001.

[3] Intrusion Inc. SecureNet Sensor 4.4 User Guide, part 700-0546-101 REV. B, Dec. 2002.

[4] 오승희, 남택용, "능동 보안 관리 기술에서 센서 무결성 보장 기법", JCCI 2002, V-A.2.1~4.

[5] 한국전자통신연구원, 차세대 인터넷을 위한

능동 보안 기술 백서, 2001년 5월.
[6] 전용희, 장정숙, "정책 기반 네트워크에서 침입탐지에 대한 연구", 한국통신학회지 제 20권, 제 7호, pp.892-907, 한국통신학회, 2003년 7월.
[7] ISS(Internet Security Systems) Technical White Paper, The Evolution of Intrusion Detection Technology, ISS.
[8] file://\ISS%20RealSecure%207_0.htm
[9] NSS, IDS Performance Testing
[10] John McHugh, "Intrusion and intrusion detection", IJIS(2001) 1: 14-35, Springer Verlag.

전 용 희

1978년 고려대학교 전기공학과 졸업(공학사)
1985~1987년 미국 플로리다공대 대학원 컴퓨터공학과
1989년 미국 노스캐롤라이나주립대 대학원 Elec. and Comp. Eng. 졸업(공학석사)
1992년 미국 노스캐롤라이나주립대 대학원 Elec. and Comp. Eng. 졸업(공학박사)
1978~1978년 삼성중공업(주) 근무
1978~1985년 한국전력기술(주) 근무
1989~1989년 미국 노스캐롤라이나주립대 Dept of Elec. and Comp. Eng. TA
1989~1992년 미국 노스캐롤라이나주립대 부설 CCSP(Center For Comm. & Signal Processing) RA
1992~1994년 한국전자통신연구원 광대역통신망연구부 선임 연구원
1994~현재 대구가톨릭대학교 컴퓨터·정보통신공학부 교수
2001.3~2003.2 동 공과대학장 역임

<관심분야> 네트워크 보안, 차세대인터넷, 통신망 성능분석, QoS 보장 기술

장 정 숙

1991년 경일대학교 공과대학 컴퓨터공학과 졸업(학사)
1992년~1995년 대구가톨릭대학교 교육대학원 전자계산교육전공 (석사)
1998년~현재 대구가톨릭대학교 대학원 컴퓨터·정보통신공학 전공 박사 수료

<관심분야> 네트워크 보안, 차세대인터넷, 통신망 성능분석, 고속 통신망 응용 서비스