표준 보안 소프트웨어 프레임워크에 대한 제안

곽주현, 이창훈
건국대학교 컴퓨터공학과
e-mail : decoze@empal.com

A Study on Standard security software framework

Ju-hyun Kwak*, Chang-hoon lee
Univ of Computer Science, konkuk University

요 약

지속적으로 복잡해져가는 침입 기술과 이에 대응하는 보안소프트웨어의 다양화에 따라 보안 시스템의 개발 및 업데이트 비용은 지속적으로 증가하고 있다. 표준적인 프로토콜과 미들웨어를 구축함으로서 개발 시간 및 중복 개발의 비용을 절감 함과 동시에 각 모듈간의 크로스오버에 의한 성능 향상을 꾀할 수 있다. 본 연구는 이런 표준 프레임워크의 효용성과 이러한 프레임워크에 요구되는 여러 요소들을 제안한다.

1. 서론

…
시장에는 다양한 보안 솔루션이 나와있다. 간단한 네트워크 모니터링 툴부터 방화벽, IDS, 백신, 취약점 분석기 등등의 시스템이 그것이다. 이들은 대부분 각각 독립적인 소프트웨어로서 각자 역할을 담당한다.

그러나 이런 보안용 소프트웨어는 어느 정도 중복적인 부분을 가지고 있으며 또한 서로간의 협력을 통해서 더욱 그 효율을 높일 수 있다. 대표적인 예로서 IDS 와 Firewall 의 연동으로 직접적인 공격방어를 수행하는 IPS 가 등장하였다.

보안 관리자입장에서는 이러한 각종 보안 소프트웨어에 대한 이해도와 지속적인 업데이트를 수행해줘야 하는데 이것은 매우 부담되는 작업일 수 있다. 본 논문에서는 이러한 각종 보안 소프트웨어의 특징을 알아보고 이것들의 연동가능성을 분석한다. 또한 이러한 분석을 바탕으로 이러한 각종 보안 소프트웨어를 통합 개발 및 관리할 수 있는 프레임워크의 모델을 제안한다.

2. 보안 프레임워크 구조

일반 사용자에게 보안의 큰 흐름은 크게 정보의 수집, 분석, 처리 이렇게 크게 3 단계로 구성된다.

본 연구는 건국대학교 산학 협력단의 연구 지원 하에 수행되었음

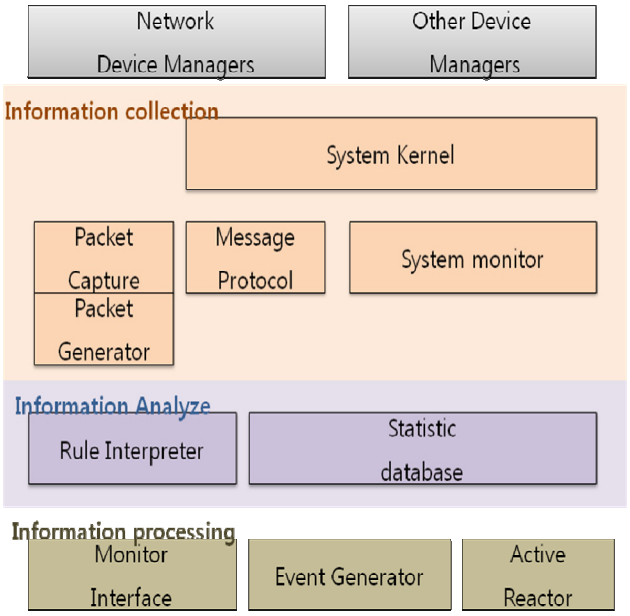
정보의 제공은 네트워크 상태, 토폴로지, 내부 트래픽, 시스템 상태 등 보안에 영향을 미칠 수 있는 다양한 data 를 수집하여 이를 의미 있는 정보로 바꾸는 과정이다. 예를 들어 Syn Flood 의 검출을 위해서 들어오는 Packet 의 Syn Flag bit 는 단순한 데이터이지만 이를 초당 Syn 패킷 숫자로 카운트할 경우 이것은 의미 있는 '정보' 가 된다.

정보의 분석은 이러한 정보들을 분석하여 보안상의 위협여부를 판별하는 단계이다. 기존의 각각의 보안 어플리케이션은 각자의 목적으로 정보를 수집하는 기능을 구현 함으로서 서로 중복되는 정보수집기능을 가지거나 또는 제한된 정보처리밖에 수집하지 못한다는 단점이 있었다. 보안 프레임워크에서 이것을 표준화된 기능으로 제공할 경우 이런 중복성이 해결되며 또한 풍부한 종류의 보안관련 정보를 얻게 됨으로 더 효율적인 대처가 가능하다.

정보의 처리 단계는 크게 즉각적인 대응과 경고 , 자료화 이렇게 3 가지로 나뉜다. 백신 프로그램이 바이러스나 웜을 찾은 후 자동으로 이것을 치료해주는 과정이나 또는 해킹의 시도가 발견된 경우 해당 IP 를 블록 하라는 커맨드를 방화벽에 전달하는 것은 대응의 예이다. 이것보다 보안의 위협이 적은 경우 관리자에게 이를 알려주는 경고를 생성하기도 한다.

자료화는 이것을 일정자료로 저장해 둠으로 인하여 인위적인 정보를 제공하거나 해당 정보를 기반으로 보안시스템을 최적화시키는 기능을 위해 이것을 일정한 형태로 저장해두는 기능이다.

이것을 위해 다음과 같은 보안 구조 모델을 제안한다.



[그림 1] 침입 탐지 프레임워크 구조도

3. 기존 소프트웨어와의 통합

실제 패킷을 캡쳐 하는 모듈은 커널의 Ethernet Device Driver 와는 다르다. 이것은 어디까지나 정보의 수집과 구체적인 대응을 목적으로 이루어진 장치관리자이며 이를 논리적 장치관리자로 정의한다. 이때 여러 장치에서 발생하는 정보를 표준화 시킴으로 (snmp 와 유사한 형태를 연상하면 된다.) 표준화된 계층적 접근을 할 수 있다.

이것은 반드시 하드웨어와 1:1 장치를 의미하는 것이 아니며 경우에 따라서는 기존의 여러 취약점 분석 및 스캔소프트웨어와도 호환할 수 있는 가능성을 열어준다. 실제로 이런 시도는 nmap 이나 nikto 등의 네트워크 스캔 툴이 여러 취약점 분석기에 포함 및 외부 모듈로 접목되어 사용되는 것을 생각하면 된다. 즉 이런 외부 툴의 결과를 직접 표준화된 정보로 변환해주는 중간 변환기가 있다면 이것 역시 하나의 논리적 정보수집 장치로 간주된다.

그러므로 외부 툴의 경우는 이런 결과를 파싱 하거나 표준화 시켜주는 그런 구조를 이용해서 LDM 을 생성함으로서 기존의 소프트웨어와의 호환 및 여러 다양한 플러그인 형태의 확장을 꾀할 수 있다. (물론 이런 정보를 분석 엔진에서 포함해야 의미가 있다.)

4. 당면 과제 및 결론

본 연구에서는 여러 다양한 네트워크 보안용 툴들의 분야를 정리하고 이런 다양한 보안 Domain 을 통합시킴으로 얻는 이점을 설명하였다. 그러나 현실적으로 이러한 거대한 시스템의 개발의 문제점을 해결하기 위해 개방화된 프레임워크 구조를 구축함으로서 이를 구현할 수 있다는 점을 설명하였다.

프레임워크를 기반으로 한 시스템 구축의 장점은 그것을 매우 작은 형태에서 확장시켜 갈 수 있다는 점이다. 이러한 통합시스템을 위해 1 차적으로 필요한 것은 각 계층별 프로토콜로서 각 메시지 및 정보 접근 형태를 표준화 시키는 것이다.

그와 동시에 논리적인 정보를 제어할 수 있는 범용 스크립트 엔진이 필요하다. 이러한 스크립트 엔진은 다양한 형태의 데이터를 정보로 가공하거나 또는 가공된 정보를 일정한 규칙하에 걸러내고 또 의미 있는 행동으로 연결 키기 위한 Rule 베이스와 통합되어야 한다.

참고문헌

[1] Frincke, Tobin, et al. “ A Framework for Cooperative Intrusion Detection “ , University Idaho, 1998

[2] Marko Jahnke, Michael Bussmann, Sven Henkel, Jens Tölle , “Components for Cooperative Intrusion Detection in Dynamic Coalition Environments “, Proceedings of NATO/RTO IST Symposium on Adaptive Defence in Unclassified Networks, 2004

[3] Phillip A. Porras, Peter G. Neumann, “EMERALD: Event monitoring enabling responses to anomalous live disturbances”,In Proceedings of the 20th National Information Systems Security Conference, 1997

[4] M Arboi ,”The Nessus Attack Scripting Language Reference Guide”,
http: //www.nessus.org/doc/nasl2_reference.pdf, 2002