情 報 保 護 學 會 誌
第 13 券 第 1 號, 2003. 2

NIDS에서 False Positives를 줄이기 위한 동적 중요도 계산 방법에 대한 연구

이 은 영*, 김 병 학*, 박 찬 일**, 정 상 갑***, 임 채 호*, 이 광 형***

요 약

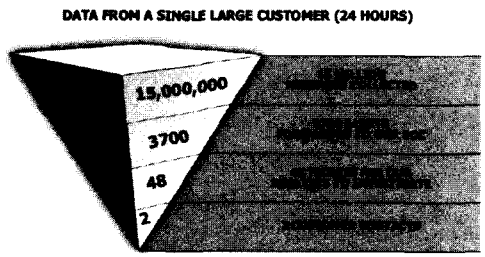
NIDS(Network Intrusion Detection System)은 실시간에 침입을 탐지하는 방안을 제시하는 시스템이지만 침입에 대한 탐지 보다 더 많은 false positives 정보를 발생시키고 있다. 많은 false positives로부터 실제 침입을 찾아내는 것은 NIDS를 효율적으로 운영하기 위해서 필요한 새로운 일이 되고 있다. 본 논문은 NIDS에서의 false positive를 줄이기 위한 동적인 중요도 계산 모델을 제시한다.

제안된 방법은 공격의 4가지 특성(공격 의도, 공격자의 지식정도, 공격의 영향 그리고 공격의 성공 가능성)을 이용한다. 만약 공격자가 공격의 의도가 크거나 많은 지식을 가지고 있다면, 보통의 경우보다 공격에 성공할 확률이 높다. 또한 공격의 대상이 특정 공격에 취약하거나 특정 공격이 대상 시스템에 미칠 영향이 큰 경우에는 더욱 더 중요한 공격이 된다고 할 수 있다. 이런 4가지의 특성을 이용하여 제시한 본 논문은 결과는 상당히 많은 부분에 대한 false positives를 줄이는 효과를 가지고 왔으며, 또한 공격에 대한 중요도의 정확성을 향상시켜서 NIDS의 관리를 쉽게 할 수 있도록 한다.

Ⅰ. 서 론

NIDS(네트워크 침입 탐지 시스템)은 네트워크 상의 지나가는 패킷을 분석하여 실시간에 침입을 탐지하기 위해서 사용되는 대표적인 네트워크 감사도구이다. 한대의 설치로 여러 시스템을 보호할 수 있어 관리상 비용이 저렴하고 시스템에 대한 부하가 적기 때문에 많은 기업들이 NIDS를 이용하여 침입을 감시하고 있다. 하지만 지속적인 스위칭 기술의 발달과 Bandwidth의 향상 등 네트워크 기술의 향상이 일어남에 따라 네트워크 속도가 증가하고 네트워크 상의 트래픽이 증가하고 있는 상황에서 NIDS도 여러대가 설치되어야 하고 감시 데이터의 증가로 인한 많은 양의 이벤트를 발생시키고 있다.

그림 1-1은 미국 보안 관제 업체인 Counterpane에서 발표한 한 개의 회사에서 하루 동안 발생한 이벤트를 분석한 결과이다[10]. 그림에서 알 수 있듯이 하루 동안에 천 오백만 개의 이벤트가 발생하였으며, 48개만이 관찰대상으로 분류 되었으며, 이를 분석을 해본 결과 단 두 개의 이벤트만이 대응할 만한 의미를 가졌다.



〔그림 1-1〕 한 회사에서 하루 동안 발생한 이벤트의 수 (Counterpane Inc.)

* 한국과학기술원 전산학과 ({comang, bhkim, chlim}@if.kaist.ac.kr)
** 한국과학기술원 전산학과 (cipark@camars.kaist.ac.kr)
*** 한국과학기술원 전산학과 ({sgjeong, khlee}@bioif.kaist.ac.kr)

이러한 사실로 볼 때 현재의 IDS는 많은 문제점을 가지고 있으며, 특히 False positive의 양과 이에 대한 분석 문제는 침입에 대한 대응을 제대로 할 수 없게 하는 주요한 원인이 되고 있다. 현재의 NIDS가 가지는 주요한 문제점을 정리하면 다음과 같다.

NIDS가 침입 감지 능력을 향상하는데 초점을 두고 개발되어 많은 침입을 판단하지만 이와 함께 false positive 정보를 많이 발생해서 실제의 공격이 무엇인지 구분 하기가 힘들며, 감지 후의 대응 부분에 대해서는 강조가 이루어 지지 않고 있다.

NIDS는 공격이 시스템에 미치는 정도에 따라 이벤트에 몇 단계의 정적인 중요도를 설정 한다. 하지만 시스템의 취약성의 정도는 시간에 따라 변하며 이를 반영하지 못하고 있다.

보다 위험한 공격은 적은 시도로 공격에 성공한다. 적은 정보로 정확한 판단이 이루어 지지 않으면 안 된다.

NIDS는 가장 기본적으로 발생되는 이벤트들에 대해서는 무시하는 경향이 있다. 예를 들어 스캔 같은 공격은 공격자들이 대상 시스템의 상태를 파악하기 위해 거의 모든 경우에 실행하는 공격이고 이런 사용자는 앞으로 공격을 수행할 가능성이 많으므로 중요시 다루어져 한다. 즉 이전에 정보 수집을 한 사용자는 더 민감하게 관리되어야 한다.

위와 같은 문제점을 해결하고자 본 논문은 이벤트의 새로운 위험도 평가 방법을 제안한다. 즉, NIDS에서 False positive 정보를 감소 시키고 시스템의 상황과 사용자의 공격의도에 따라 동적으로 위험도를 설정 하여 관리자가 효과적으로 이벤트를 분석 할 수 있는 위험도 평가 방법을 제안한다. 2장에서는 false positive를 줄이기 위해 현재 사용되고 있는 방법들을 소개 한다. 3장에서는 제안한 위험도 방법을 소개 한다. 4 장에서는 제안한 방법의 구현을 통한 실험 결과를 분석한다. 마지막으로 5장에서 결론을 맺는다.

Ⅱ. 관련연구

2.1 Reducing False positives

2.1.1 침입 패턴을 개선

NIDS가 정상 패킷을 공격으로 오인 하는 경우는 원초적으로 침입을 판단하는 침입 패턴이 잘못되었기 때문이다. 따라서 NIDS의 False positive를 줄이기 위해서는 signature을 정확하게 만들어야 한다. 모든 상황에서 정확한 침입을 탐지하는 false positive를 만드는 것은 어렵다. 따라서 false positive가 발생 했을 경우 이를 IDS 개발자에게 알려주어 개발자들이 signature를 개선하여야 한다.

IDS를 개발하고 있는 많은 회사들은 정확한 침입 패턴을 생성하는데 중점을 두고 있다. 정보보호 회사인 ISS도 false positive를 심각하게 다루고 있다. 새로운 침입 패턴을 발표하기 전에 ISS는 광범위한 품질 보증 연구팀과 24×7의 ids 관리 보호 모니터링 서비스 팀에서 signature를 테스트 한다. 이런 광범위한 테스팅 과정은 여러 네트환경에서 발생할 수 있는 예기치 않는 false positive를 줄인다[2].

그러나 오늘날 네트워크의 복잡성과 다양성으로 인하여 모든 가능한 false positive를 줄이는 것은 무척 어렵다. 이런 이유로 인해 IDS vendor들은 예기치 않는 false positive를 인식하고 해결하는데 사용자의 경험에 의지 하고 있는 실정이다.

2.1.2 시스템 네트워크 상황과 정책에 맞게 NIDS 환경 설정

IDS로부터 발생한 이벤트가 시스템 상황에 맞지 않는 경우에는 IDS개발자가 아닌 시스템 관리자에 의해 행해지는 데 다음과 같은 방법들이 연구되고 있다.

시스템 관리자가 자신의 네트워크와 시스템 상황에 맞게 침입 탐지 시스템 Signature의 환경설정을 통해 false positive를 줄이는 방법이다. Sync flood 같은 많은 signature들은 환경을 설정할 수 있다. Signature의 default 세팅은 모든 네트워크 상황에 맞지 않는다. 이들이 알맞게 조율되어 있지 않기 때문에 signature는 일반적인 네트워크 트래픽상에 있다고 여기게 되며 잘 못된 이벤트를 발생한다. 즉, 관리하고자 하는 네트워크의 설정에 맞게 signature들을 튜닝 함으로써 false positive를 줄이는 방법이 있다.

그리고 회사의 보호 정책에 맞게 IDS의 환경 설정함으로써 false positive를 줄이는 방법이 있다. 즉, 정책에 맞지 않는 signature를 무력하게 함으로써 false positive를 줄인다. 예를 들어 보호하고자 하는 네트워크에 솔라리스 시스템의 정보가 없다면 솔라리스 플래폼과 관련된 정보를 무력화시키고 단지 기록만 하도록 하거나 위험도 설정을 무력화시킨다.

2.1.3 시스템 환경에 맞는 Signature를 정의

다른 연구로는 NIDS에서 제공하는 signature

를 사용하는 대신에 관리자가 직접 상황에 맞는 signature를 정의 하여 사용하는 방법이 있다. 예를 들어 http get signature를 사용하여 도박 사이트, 해킹 사이트, 또는 포르노 그래픽 사이트에 대한 접근과 같은 비적절한 웹 사용을 감시하고자 하는 경우, http get signature은 사용자가 어느 웹사이트에 접근할 때 마다 경고를 발생한다. 또한 많은 트래픽을 발생함으로써 시스템에 주는 부하가 크다. 일반적인 HTTP-GET signature를 사용하는 대신에 Http url의 특정 단어나 위치를 감시하는 signature를 사용함으로써 시스템의 부하를 적게 하고 정책에 맞는 이벤트를 탐지해 낼 수 있다.

2.1.4 주변에 의해 차단되는 실패한 이벤트들을 인식

다른 정보보호 시스템을 설치 함으로써 이로부터 나오는 정보를 이용하여 false positive를 줄이는 방법이 연구되고 있다. 침입 차단 시스템은 공격 자체를 필터링 해버리기 때문에 차단되는 공격은 성공하지 못한 이벤트로 인식함으로써 false positive를 감소 시킬 수 있다. 다른 정보 보호 제품을 이용한 자산평가는 보호하고자 하는 네트워크에 대한 정보를 제공한다. 이런 정보와 IDS 시스템을 연관시킴으로써 false positive를 줄이는 방법이 시행되고 있다.

2.1.5 문제점

위의 연구들은 환경설정을 통하여 줄이는 방법이 대부분이기 때문에 네트워크 환경이 자꾸 바뀌는 상황에서는 관리하기 어렵다는 단점이 있다. 또한 취약성의 정도가 다른 개별 시스템을 보호하는 경우에는 이용하기 어렵다.

2.2 침입 탐지 시스템에서의 위험도 평가방법

2.2.1 시스템에 미치는 영향에 따른 위험도 설정

대부분의 NIDS는 자체적으로 시스템 취약성의 위험도에 따라 공격의 위험도를 측정한다. 각 시스템 환경에 따라 그 기준은 조금씩 다르기도 하지만, 그림 2-1과 같이 "상", "중", "하"로 구분하여 표시해 준다[8,9].

위험도 "상"	• 원격 공격자가 시스템 보안을 위배할 수 있는 취약점 (예, 사용자 또는 관리자 권한 획득) • 시스템의 완전한 통제 권한을 획득 가능한 로컬 공격 • 일반 필드에서 많이 발생하는 사고와 관련된 취약성 (CERT/CC와 연관) (service attacks and compromises? database, web, ssh)
위험도 "중"	• "상" 이나 "하"에 해당하지 않는 취약성 (Denial of service : and)
위험도 "하"	• 시스템의 중요한 정보나 통제의 손실을 야기하지는 않지만, 공격자에게 또 다른 취약성을 발견하거나 다른 공격에 사용할 수 있는 정보를 제공하는 취약성 • 대부분의 기관에 중요하지 않다고 판단되는 취약성 (ports scan, automated scans and sweeps)

〔그림 2-1〕 NIDS의 이벤트 위험도

2.2.2 공격 횟수에 따른 위험도 설정

같은 이벤트가 계속 발생하였을 때 그 횟수에 따라 위험도를 주는 방법이다. 인터넷 웜이나 바이러스 같은 공격들은 프로그램이나 스크립트에 의해 실행되므로 항상 같은 방법으로 공격을 시도한다. 인터넷 웜이나 바이러스는 불특정 다수를 대상으로 짧은 시간에 많은 시스템을 감염시킨다는 특징을 가지고 있다. 같은 공격이 계속 적으로 탐지된다는 것은 인터넷 웜이나 바이러스가 네트워크 상에 퍼졌다는 것을 의미한다. 공격의 횟수로 위험도를 설정함으로써 이러한 공격을 미리 탐지하고 대응할 수 있다.

2.2.3 문제점

취약성은 시간에 따른 life 사이클을 가진다. 취약성이 발견되면 이를 보안할 수 있는 패치가 나오고 또 다시 새로운 취약성이 발견되면 다시 이를 보안하는 방법들이 고안된다. 즉 시스템의 취약성은 시간이 지나면서 그 위험도가 적어진다. 하지만 시스템에 미치는 영향에 따른 위험도 설정방법은 이벤트가 정적인 위험도를 가지므로 시간이 지나도 이벤트는 과거와 같은 위험도를 나타내는 문제를 가지고 있다.

예를 들어 과거에 9라는 위험도를 가졌다면 시간이 흘러 대부분의 시스템들이 취약성을 보안한 후에도 9라는 위험도를 나타낸다. 격 횟수에 따른 위험도 설정은 웜이나 바이러스 같이 동일한 공격이 많이 시행되었을 때에는 의미를 가진다. 하지만 사람들에 의해 행해진 공격들은 매우 다양하므로 적용하기 힘들다. 또한 전문 해커 같이 적은 시도로 공격을 수행하는 경우는 탐지하기 어렵다는 문제점이 있다.

Ⅲ. 제안한 위험도 평가 방법

3.1 위험도 평가 모델

의미 있는 이벤트라고 하는 것은 관리자가 우선적으로 관리해야 할 필요성이 있는 이벤트를 말한다. 시스템에 대한 공격자들을 보면 Script kids 와 전문 해커들로 크게 나 수 있다. Script kids는 공격에 대한 지식이 없이 공격을 시도하는 사용자로 대부분 자동화 도구를 이용. 많은 공격 시도를 하지만 공격 성공률은 무척 적다. 대부분의 공격은 Script kids들의 공격이 대부분 이다. 반면에 전문 해커들은 공격에 대한 지식이 많고 이를 이용하여 대상 시스템에 대한 정보를 수집하고, 취약성을 공격, 아주 적은 시도로 공격에 성공하는 사용자들이다. 관리자는 Script kids, 에 의한 공격보다는 전문 해커에 의한 공격을 우선적으로 대응해야 한다. 이렇듯 침입에 대한 지식이 많고 의도를 가진 공격자에 의해 행해진 공격이라면 우선적으로 관리하여야 할 필요가 있다.

공격자의 의도 정도(I)
공격자가 가진 지식 정도(K) → 시스템에 미치는 영향 (P)
공격 성공 가능성(C)

즉 침입에 대한 지식이 많고 공격의도를 가진 공격자가 한 공격일수록 우선적으로 관리해야 하면 또한 그 공격이 보호하고자 하는 시스템에 미치는 영향이 높을수록 관리자는 먼저 대응을 하여야 한다. 그리고 시스템의 자산과 취약성을 비교하여 보호하려는 시스템에 대한 공격이 아닌 경우 false positive 를 감소시킬 수 있다.

본 논문에서는 이벤트의 위험도를 평가하기 위한 다음의 모델을 제안한다.

$$\text{이벤트의 위험도} = (I \times K) \times (C \times P)$$

각 구성성분을 평가하기 위해 표 3-1의 정보를 이용하였다.

〔표 3-1〕 위험도 평가를 위해 사용되는 항목

Event Priority	공격자 의도 정도(I)	전체 패킷 중 공격 패킷의 비율 (P1) 정보 수집의 정도 (P2)
	공격자가 지닌 지식 정도(K)	시간에 따른 Signature 위험도 (P3)
	시스템에 미치는 영향(P)	기존의 NIDS의 위험도 (P4)
	공격의 성공 가능성(C)	취약성과 시스템의 자산 비교 (P5)

$$\text{이벤트의 위험도} = ((P_1 + P_2) \times P_3) \times (P_4 \times P_5)$$

P1~P4의 위험도는 1~10으로 평가하였고 P5는 공격의 성공 여부에 따라 1과 0으로 평가하였다.

3.2 공격자가 지닌 의도 파악 방법

각 호스트 별로 전체 패킷 중 공격 패킷의 비율과 정보 수집의 정도를 가지고 공격의도를 파악하였다.

전체 패킷중 공격 패킷의 비율에 따른 공격의도 파악

공격 의도가 있는 사용자들의 패킷에는 일반 사용자 보다는 공격 패턴이 많이 발생할 것이다. 즉, 실제 패킷 중에 공격이라 탐지된 이벤트의 비율이 높은 사용자는 공격에 대한 의도를 가지고 있다고 생각하였다.

Script kids 같은 초보 공격자들은 대부분 자동화 도구를 사용하기 때문에 많은 공격 시도를 하나 성공 횟수가 적고 반면에 전문 해커들은 적은 시도로도 공격에 성공한다. 이런 이유로 기존에 사용되는 이벤트 횟수에 따른 공격 위험도 판단은 전문 해커에 의해 행해지는 적은 시도의 공격을 탐지하지 못한다는 문제점이 있다.

이런 문제점을 해결하고자 호스트 별로 전체 패킷 중 공격 패킷의 비율을 이용하여 위험도를 평가하였다. 이 방법은 적은 정보로도 위험한 공격자에 대한 정보를 얻을 수 있다.

$$P_1 = \frac{Alert\ request}{Total\ request} \times 10$$

정보 수집의 정도에 따른 공격 의도 파악

공격의도를 가진 사용자는 대부분 정보 수집의 단계를 거친다. 즉 공격하고자 하는 시스템에 대한 정보를 수집한 후 이를 이용하여 시스템의 취약성을 공격한다. 예를 들어 스캔이나 ping과 같은 방법을 이용하여 시스템의 운영체제나 제공하는 서비스들에 대한 정보를 얻고 이의 취약성을 이용하여 공격을 시도한다. 어떤 호스트로부터 스캔과 같은 이벤트가 발생하였다는 것은 이 호스트가 시스템을 공격할 의도를 가지고 있음을 나타낸다. 이를 바탕으로 스캔과 같은 정보 수집에 관련된 이벤트를 호스트 별로 관리함으로써 공격의도를 가진 사용자에 대한 정보를 평가하였다.

정보를 수집하는 방법은 대상 시스템에 대해 서비스 요청 후 이에 대한 시스템의 반응을 분석하여 정보를 수집한다. 따라서 정상적 사용자의 요청이 공격으로 오인될 수도 있다. 따라서 시스템이 제공하는 서비스, 예를 들어 ftp, dns 에서 발생하는 스캔 이벤트는 그 중요성 정도가 적어야 하며, 반면에 시스템에서 보호해야 된다고 생각하는 서비스나 공격이 많이 시도되는 서비스들은, 예를 들어 rpc를 이용한 스캔 이벤트, 중요성이 크게 설정되어야 한다.

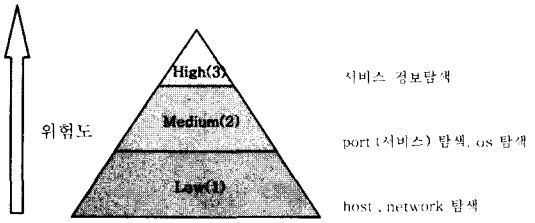
이와 같이 각 시스템에 맞는 수집된 정보를 평가하고자 정보 수집 단계를 세 단계로 나눈 후 각 단계별로 테이블을 만들었다. 그림 3-1 정보 수집이라고 판단된 이벤트가 각 경우를 만족 하는 경우 호스트 별로 위험도를 더함으로써 전체 정보 수집 정도에 비해 호스트에서 발행한 정보 수집 정도를 이용하여 공격의도를 평가하였다.

이벤트 단계	출발지 IP	출발지 포트	도착지 IP	도착지 포트	Weight

〔그림 3-1〕 공격 타입 단계 별 테이블

$$P_2 = \frac{\sum Weight}{Max\_Weight} \times 10$$

정보 수집의 단계는 탐지 정보의 위험에 따라 세 가지로 나누었다. 그림 3-2 먼저 공격 하고자 하는 호스트나 네트워크를 탐색하는 단계(1)와 탐색한 호스트의 운영체제나 제공하는 서비스에 대한 정보를 얻는 단계(2) 그리고 앞서 수집한 정보를 통해 특정 서비스에 대한 정보를 얻으려는 단계(3)로 나누었다. 각 단계는 다음 단계를 나아가기 위한 정보를 제공하게 되므로 점차 그 위험이 높아 짐을 알 수 있다. 이에 따라 다음과 같이 정보 수집 단계를 분류하였다.



〔그림 3-2〕 정보 수집 단계

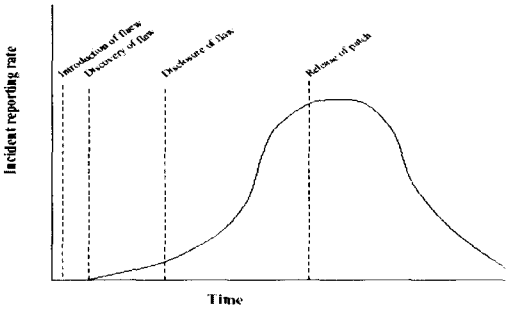
3.3 침입 지식에 대한 평가

초보 공격자들은 침입에 관한 지식이 적어 이미 잘 알려진 취약점을 이용하여 공격을 시도한다. 이러한 공격들은 시스템의 취약점이 대부분 보안 되었기 때문에 성공할 가능성이 없다. 반면 전문 해커들은 침입에 대한 지식이 풍부하기 때문에 최근에서야 알려지게 된 취약성을 이용하여 공격을 시도함으로써 그 성공의 가능성이 크다.

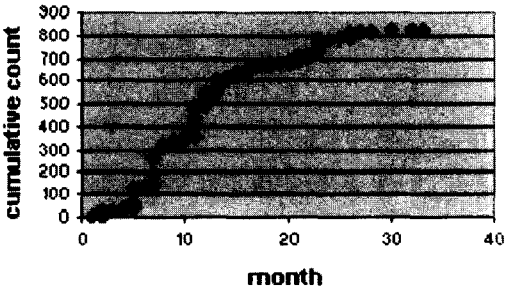
이를 바탕으로 새롭게 발견된 취약점을 찾아 공격하는 사용자들은 침입에 대한 지식이 높다고 판단한다. 새로운 취약점을 이용한 공격이 발견되었을 시에는 이를 탐지 할 수 있는 signature에 더 높은 우선 순위를 두어야 한다. 즉 새롭게 발견된 취약점을 탐지하는 signature에 오래된 취약점을 탐지하는 signature 보다 높은 위험도를 주어 탐지되었을 때 위험의 정도를 더 심각하게 고려 할 수 있어야 한다. 기존의 NIDS의 위험도 평가 방법은 침입을 탐지하는 signature의 위험도가 피해 시스템에 미치는 영향에 따라 정적으로 설정되어 있어 시간에 따라 변하는 시스템의 취약성과 다르게 항상 같은 위험도를 나타내었다. 본 논문에서는 제시한 방법은 각 signature에 시간에 따른 위험도를 설정하여 침입 지식에 대한 평가의 자료로 이용하므로 이벤트의 위험도가 시스템 취약성 정도에 따라 변한다.

그림 3-3은 이상적으로 이론적으로 생각한 시간에 따른 침입탐지기록에 관한 그래프이다. 취약점을 발견되었을 때 처음에는 이를 이용한 공격이 적지만 취약점이 알려지면서 이에 대한 공격이 급격히 증가한다고 생각했다. 취약점을 보완할 수 있는 패치가 나오게 되면 이를 이용한 공격이 점차적으로 줄어

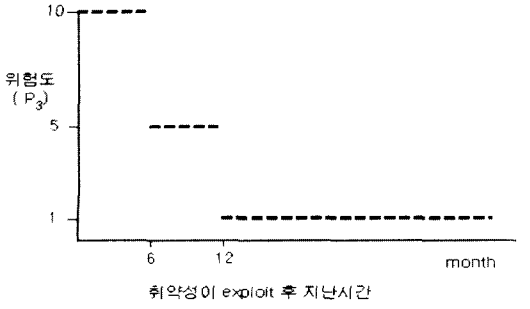
들 것이라 생각하였다. 그러나 Browne은 실제로 행해진 공격들의 분석을 통해 이와 다름을 밝혀 내었다[11]. 그림 3-4는 시간에 따란 phf라는 취약점을 공격한 사건들을 기록한 것이다. 시간이 지남에 취약점이 보안된 이후에도 계속적으로 공격이 증가함을 보여준다 하지만 취약점이 보안된 이후에는 그 이벤트의 위험도는 감소해야 한다. 본 논문에서는 일반적인 경우를 가정으로 대부분의 시스템이 취약점이 발표된 후 1년 후에는 그 취약점을 보완하였다고 가정하여 시간에 따른 signature에 그림 3-5 같이 위험도를 설정하였다.



〔그림 3-3〕 이론적인 침입탐지기록



〔그림 3-4〕 Phf Incident : Cumulative Count Plot



〔그림 3-5〕 시간에 따른 signature의 위험도

3.4 시스템에 미치는 영향에 따른 평가

기존의 NIDS에서 시스템에 미치는 영향에 따라 각 signature에 따른 중요도를 설정하였다. 이 중요도는 공격이 성공 했을 경우 시스템에 미치는 영향을 고려하여 작성되었다. 이 중요도도 무시할 수 없는 중요한 기준이며, 이를 이용하여 시스템에 미치는 영향에 따른 평가의 기준으로 삼았다.

P_4 = NIDS에서 사용하는 기존 중요도

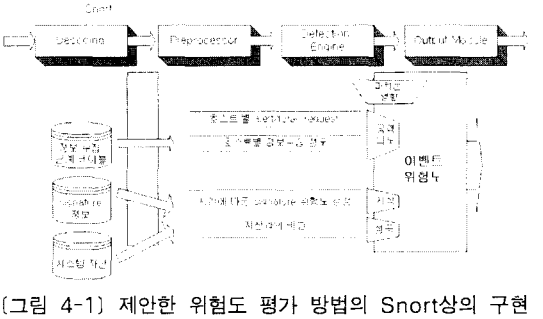
3.5 공격의 성공 여부 판단

모든 취약성은 취약성이 적용 될 수 있는 자산의 범위가 있다. 예를 들어서 CAN-1999-0763이라는 CVE 넘버를 가진 취약성은 Microsoft IIS 4.0 alpha 버전에 해당되는 취약성이며, showcode.asp라는 signature를 가진다. 곧 apache web server에 이 공격을 하면 공격이 이루어 질 수 없을 것이다. 곧 취약성과 자산을 데이터 베이스화하여 이 정보를 중요도의 하나로 사용한다면 필요 없는 false positive를 줄일 수 있게 된다.

P_5 = 1 (자산과 취약성이 연관성이 있을 때)
= 0 (자산과 취약성이 연관성이 없을 때)

Ⅳ. 실험 및 분석

4.1 구현 구조 설명



〔그림 4-1〕 제안한 위험도 평가 방법의 Snort상의 구현 구조

Snort[5,6] 초기 설정 시 정보 수집 단계에 따른 테이블과 Signature 정보, 그리고 시스템 자산을 읽어와 데이터 베이스를 구축한다. 패킷이 들어오면

호스트 별로 공격의도를 가졌는지를 판단한다. 전체 패킷에 들어 있는 공격 패킷의 비율을 유지(P1)하고 이벤트가 정보 수집인 경우 테이블과 비교하여 정보 수집 정도(P2)를 계산한다. 침입에 대한 공격자가 지닌 지식의 정도를 시간에 따라 Signature의 위험도를 설정(P3)함으로써 판단한다. 공격에 따른 시스템에 미치는 영향은 Snort가 기존에 설정한 이벤트의 위험도(P4)를 이용한다. 마지막으로 자산과의 비교로 공격의 성공 여부(P5)를 판단하였다.. 각 과정에서 얻은 P1~P5를 이용하여 3장에서 제안한 평가 모델에 적용함으로써 전체 위험도를 산출해 내었다.

4.2 실험 방법

실험 방법은 기존 Snort와 제안된 위험도 방법을 적용한 Snort를 실행하여 발생하는 이벤트의 수와 위험도를 비교하였다. 공격의 침입 시도를 위해 취약점 점검 도구인 Nessus를 사용하였으며 실험에 이용된 Snort는 포트 스캔 detector를 가동한 상황과 보안 룰은 인스톨 시 기본 옵션으로 되어 있는 상황에서 사용하였다 실험을 위해 연구실 컴퓨터에 Snort를 설치하였으면 보호하고자 하는 시스템은 연구실 시스템의 자산을 이용하였다.

실험 평가 항목은 기존 Snort와 제안된 위험도 방법을 이용한 Snort에 나오는 이벤트 수를 비교하여 false positive의 감소 양과 위험도의 분석를 통한 제안한 모델에서의 각 위험도 평가 항목의 전체 위험도에 미치는 영향을 분석해 보았다.

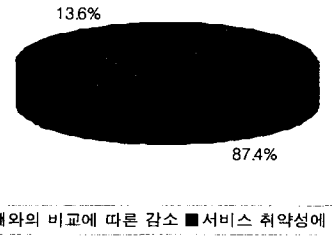
4.3 실험 결과 및 분석

표 4-1는 제한한 방법에 따른 false positive의 감소를 보여준다.

〔표 4-1〕 제안한 방법에 따른 false positive 감소

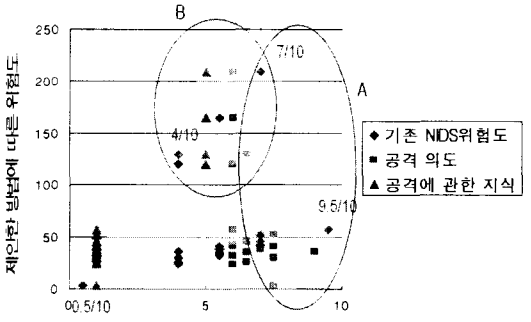
	before	After	감소율
NIDS1	1118	639	43.8%
NIDS2	4476	627	86%

표 4-1은 시스템 자산과 제공하는 서비스에 따라 차이가 있으나 이벤트 양이 약 50% 이상이 감소됨을 알 수 있다. 그림 4-2는 감소된 false positive의 원인을 분석해 본 그래프이다



〔그림 4-2〕 false positive 감소 원인 분석 그래프

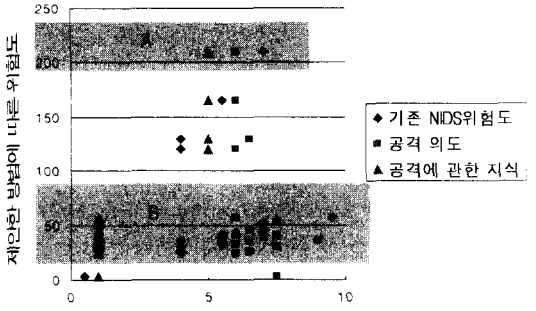
위의 결과를 통해 시스템 자산과 취약성의 비교를 통해 많은 false positive가 감소하였음을 볼 수 있다. 그림 4-3은 위험도를 비교하기 위한 그래프이다. X축은 제안한 방법을 사용된 각 성분의 위험도 값이고 Y축은 제안한 방법에 따른 위험도 이다. 위 그래프에서 노란 가로로 긴 원에 있는 각 성분은 한 이벤트에서 추출된 값들이다.



〔그림 4-3〕 위험도 변화 분석 그래프

위 그래프에서 나타나고 하는 바는 기존 NIDS의 위험도 평가 방법에 의해서 높은 위험도를 가지는 값들은 원 A에 몰려 있지만 제안한 방법에 따른 위험도에서는 높은 위험도들이 B에 위치 한다는 것을 알 수 있다. A의 맨 오른쪽에 위치해 있는 기존의 위험도가 가장 높은 값은 공격 의도나 공격자의 지식이 낮기 때문에 새로운 방법에서는 중요하지 않는 이벤트로 분류된다. 하지만 B원에 위치한 것들은 모든 값들이 높은 형태로 나타나 새로운 방법에서 중요하게 인식되고 있다.

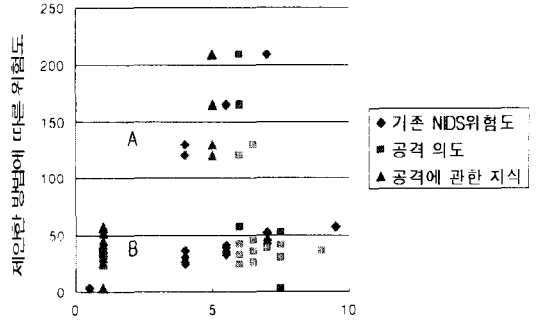
그림 4-4그림 4-5는 위험도의 변화를 분석한 그래프로 공격에 관한 지식의 정도에 따라 위험도가 어떻게 변하였음을 보여주고 있다.



〔그림 4-4〕 공격에 관한 지식에 따른 위험도 변화 분석 그래프

B 지역의 높은 위험도를 가지고 있던 이벤트들의 대부분이 공격에 대한 지식이 거의 없음을 볼 수 있다. 즉 오래된 취약성을 공격함으로써 Script kids 같은 공격자라고 판단할 수 있다. 반면에 A지역에 있는 이벤트는 B지역의 이벤트에 비해 새로운 취약성을 공격함으로써 지식의 정도가 높아 위험도가 높아졌음을 볼 수 있다.

그림 4-5는 공격의도에 따른 위험도 변화를 보여준다. A 지역의 경우 기존의 NIDS의 위험도는 같았지만 공격의도가 높은 이벤트가 더 높게 위험도가 평가 되었다. B지역의 경우도 이와 마찬가지로 공격의도가 높은 이벤트가 더 높게 위험도가 평가되었음을 보여주고 있다.



〔그림 4-5〕 공격 의도에 따른 위험도 변화 분석 그래프

이를 종합적으로 분석해보면 A지역의 이벤트들의 침입에 대한 지식 정도가 낮아 오래 전 취약성을 이용하여 공격함으로써 전체적 위험도가 적어졌음을 볼 수 있다. 즉 Script kids에 의한 공격으로 볼 수 있다. B지역의 이벤트들은 침입 의도도 높으며 최신 취약성을 공격하는 것으로 보아 지식의 정도가 높다. 이는 전문 해커의 공격으로 알 수 있다. 이의 결과로 제안한 위험도 측정법이 공격자의 침입에 대한 지식과 공격의도가 이벤트의 위험도에 반영됨을 알 수 있다.

4.4 기존 연구와의 비교

	환경 설정을 통한 && false positive 감소법	미치는 영향에 따른 위험도 평가	제안된 위험도 평가 모델
False positive 감소	보통		좋음
위험도에 시간에 따른 취약성 변화 반영	나쁨		좋음
공격자의 지식이나 의도에 대한 정보	알수 없음	알수 없음	좋음
자산과의 비교에 따른 시스템에 미치는 영향	없음	없음	보통

V. 결론 및 향후 연구

기존의 false positive를 줄이는 연구들은 환경설정을 통하여 줄이는 방법이 대부분이기 때문에 네트워크 환경이 자꾸 바뀌는 상황에서는 관리하기 어렵다는 단점이 있다. 또한 취약성의 정도가 다른 개별 시스템을 보호하는 경우에는 이용하기 어렵다. 이를 해결 하기 위해 보호하고자 하는 시스템 별로 운영체제와 제공하는 서비스를 데이터 베이스 화하여 공격하는 취약성과 비교함으로써 false positive를 감소 시키는 방법을 제안하였다. 이 방법은 네트워크 상황의 변화에 영향을 받지 않으며 취약성이 다른 각각의 시스템을 보호 할 수가 있다.

기존의 NIDS는 공격이 시스템에 미치는 영향에 따라 정적으로 위험도를 평가하는 방법과 탐지 횟수가 높은 이벤트에 보다 높게 위험도를 평가하는 방법을 사용 하였다. 그러나 이러한 방법은 적은 시도로 공격에 성공하는 전문 해커들의 공격을 탐지 하기 어려우며 과거에 취약성이 보안된 이후에도 계속 같은 위험도를 나타내어 시간에 따른 시스템의 취약성 정도를 반영하지 못하는 단점을 가지고 있다. 또한 스캔과 같이 시스템에 미치는 영향은 없으나 정보를 수집하는 이벤트들에 낮은 위험도를 부여하고 무시하는 경향이 있다. 그러나 이런 이벤트는 공격

하고자 하는 시스템의 정보를 수집하는 과정이므로 사용자가 앞으로 더 위험한 공격을 수행할 위험을 가지고 있기 때문에 잘 관리되어야 할 필요성이 있다.

이를 해결하기 위해 본 연구에서는 새로운 방법의 위험도 평가 방법을 제안하였다. 먼저 위험한 공격자를 평가 하여 이를 위험도 평가에 이용하였다. 전체 패킷 중에 공격 패킷의 비율을 사용함으로써 적은 시도로 공격에 성공하는 전문 해커들의 공격을 탐지 할 수 있었으며 정보를 수집하는 이벤트가 탐지된 호스트 별로 관리함으로써 앞으로 위험한 공격을 수행할 사용자에 높게 위험도를 부여하였다. 또한 Signature에 시간에 따른 위험도를 설정함으로써 시간에 따른 시스템의 취약성 정도를 위험도에 반영함으로써 기존의 위험도 평가 방법의 단점을 보안하였다.

제안한 방법의 실험결과 보호하고자 하는 시스템의 자산과 제공하는 서비스에 따라 차이가 있었지만 약 50%이상의 false positive를 감소를 확인할 수 있었다. 기존에 방법과 달리 공격자의 의도와 지식의 정도를 새로운 평가 방법으로 부여함으로써 기존에 위험도가 높았던 과거 취약성을 이용해 공격을 한 이벤트들 위험 정도가 낮아졌음을 확인 할 수 있었다. 또한 같은 위험도의 이벤트라 하더라고 공격자의 의도에 따라 위험도가 다르게 설정됨을 볼 수 있었다. 결과적으로 과거에 비해 이벤트의 위험도 평가에 더 많은 정보를 부여하고 시간에 따라 시스템 상황을 반영함으로써 관리자가 보다 의미 있는 이벤트를 판단하여 알맞은 대응을 하는데 도움되었다는데 의미를 가진다.

향후 연구과정으로는 시스템의 취약성과 IDS의 침입 탐지 패턴들을 연관시키는 연구가 필요하다 시스템의 취약성을 평가하는 연구는 많이 연구되어 왔으며 현재는 시스템의 취약성에 global 한 ID를 부여하는 연구가 진행 중이다. 또한 IDS별로 시스템의 취약성을 공격하는 침입 패턴을 만드는 연구가 계속 진행 중이다. 이와 함께 이를 연관시키는 연구가 필요하다. 시스템의 취약성과 공격하는 침입 패턴을 연관시킨다면 시스템에 맞지 않는 더 많은 false positive를 줄일 수 있을 것이다.

참 고 문 헌

[1] James P. Anderson. "*Computer Security threat monitoring and surveillance*," Thechnical Report Contract 79F26400, February 26,revised April 15 1980.

[2] Internet Security Systems. "The Truth about False Positive," White Technical Report, 2001.

[3] Klaus Julish. "Dealing with False positives in Intrusion Detection," RAID 2000..

[4] Gyu-min Cho, Sang-Ho Kim, Koung-goo Lee. "Analysis on Security Functional Requirements for Intrusion Detection System," WISC 2000.

[5] Snort, www.snort.org

[6] Snort Rule management, www.whitehats.com

[7] 박현미, 오은숙, 이동연, "IP 네트워크 스캐닝 기법", www.certcc.org

[8] CVE Vulnerability Search Engine, http://icat.nist.gov/icat.cfm

[9] Common Vulnerability and Exposures. http://cve.mitre.org/

[10] Bruce Schneier. "Counterpane and Management Security Monitering," http://www.counterpane.com/. April 2002.

[11] Hillay K.Browne, William A. Arbaugh, John Mc Hugh, William L. Fithen. "A Trend Analysis of Exploitations," In *Proceedings of the IEEE Symposium on Security and Privacy May 2001*

〈著 者 紹 介〉

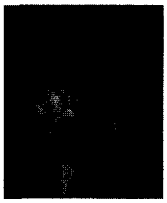
이 은 영 (Eun Young Lee)
학생회원

2001년 : 아주대학교 정보 및 컴퓨터공학부
현재 : 한국과학기술원 전산학과 석사
관심분야 : 정보보호

김 병 학 (Kim, Byunghak)

1990년 : 한국과학기술원 전산학과 졸업
1993년 : 한국과학기술원 전산학과 석사
현재 : 한국과학기술원 전산학과

박사과정
관심분야 : 정보보호



박 찬 일 (Chanil Park)

1999년 : 인하대학교 수학과 졸업
2001년 : 한국과학기술원 수학과 석사
현재 : 한국과학기술원 전산학과 박사과정
관심분야 : 정보보호

정 상 갑 (Sang-Gab Jeong)

1990년 : 육군사관학교 전자공학과 졸업
2003년 : 한국과학기술원 전산학 석사
현재 : 육군 소령 (전산분야)
관심분야 : 정보보호, 정보전

임 채 호 (Chaeho Lim)
종신회원

1986년~1992년 : KIST SERI 연구원
1993년~1994년 : 대전우송정보대학 교수
1995년 : KIST/SERI CERT-KR 팀장
한국인터넷정보보안그룹(KIS-G) 의장
1996년~2000년 : CERTCC-KR 팀장
현재 : 한국과학기술원 초빙교수
관심분야 : 정보보호

이 광 형 (Kwang H. Lee)

1978년 : 서울공대 산업공학학사.
1980년 : 한국과학원 산업공학석사
1982년 : 프랑스 INSA 전산학과 석사(DEA)
1985년 : 프랑스 INSA 전산학과 공학박사
1988년 : 프랑스 국가박사(전산학 INSA-LYON1대)
현재 : 한국과학기술원 전산학과 교수
관심분야 : 전산학, 퍼지 시스템, 정보보호