IE Interfaces
Vol. 21, No. 1, pp. 33-42, March 2008.

AOI에 기반을 둔 침입탐지시스템의 알람 분석

정인철 · 권영식†

동국대학교 산업공학과

The Analysis of IDS Alarms based on AOI

In Chul Jung · Young S. Kwon

Dept' of Industrial and Systems Engineering, Dongguk University, Seoul, 100-715

To analyze tens of thousands of alarms triggered by the intrusion detections systems (IDS) a day has been very time-consuming, requiring human administrators to stay alert for all time. But most of the alarms triggered by the IDS prove to be the false positives. If alarms could be correctly classified into the false positive and the false negative, then we could alleviate most of the burden of human administrators and manage the IDS far more efficiently. Therefore, we present a new approach based on attribute-oriented induction (AOI) to classify alarms into the false positive and the false negative. The experimental results show the proposed approach performs very well.

Keyword: intrusion detection system (IDS), attribute-oriented induction (AOI), data mining

1. 서 론

1.1 연구배경

2005년 말 우리나라의 초고속 인터넷가입자수는 약 1,200만 가구에 달하며 인터넷 사용자수는 약 3,300만 명으로 전 국민의 70%를 넘어섰다(정보통신부 2005). 이와 같은 인터넷사용 인구의 증가는 인터넷을 기반으로 하는 기업정보화를 가속시켜왔으나 해킹, 스패밍, 컴퓨터 바이러스 확산 및 다양한 글로벌 사이버 범죄의 증가등 각종 역기능도 커지고 있다. 이에 따른 막대한 사회 · 경제적 비용증대는 정보보안의 필요성과 중요성을 증대시켜 이제 기업의 정보보안은 더 이상 선택의 문제가 아닌 필수적인 기업 인프라 요소가 되었다(한국정보보호진흥원, 2003).

최근의 정보보안은 단순한 네트워크 및 시스템의 접속제어뿐만 아니라 시스템보호, 추적, 암호 등 정보보안시스템 전반에 대한 토털 솔루션을 확보하려는 방향으로 나아가고 있으며 이 중에서 침입탐지시스템(intrusion detection system)은 방화벽, VPN, 사용자 인증 등과 연계되는 통합보안 관리 시스템으로 그 이용이 증대되고 있다.

그러나 침입탐지시스템을 설치한 것으로 보안문제가 해결되는 것이 아니라 보안 관리자가 침입탐지시스템의 결과물인 알람 로그 데이터를 지속적이고 효율적으로 분석, 관리하여 오탐을 정확히 구별해내야 하는데, 현실적으로 현업에서의 침입탐지시스템이 하루에 수십만 개의 알람 로그 데이터를 발생시키므로 이를 분석하고 관리하여, 신속하게 대처하는 것은 매우 어렵다(Julisch, 2000).

알람 로그 데이터를 일일이 확인해가며 조사하는 것은 많은 시간이 걸리며, 보안전문가에 의해서 수동으로 이루어지기 때문에 비효율적이며 오탐율도 높다. 이러한 문제를 해결하기 위해서는 보다 효율적인 알람 로그 분석 방법이 필요하다.

1.2 연구 목적

수많은 알람 로그를 육안으로 파악하여 이상치를 찾아내는 작업은 많은 시간과 숙련을 필요로 한다. 또한 시스템의 규모

†연락저자 : 권영식 교수, 100-715 서울특별시 중구 필동3가 26번지 동국대학교 정보산업대학 산업시스템공학과, Fax : 02-2269-2212, E-mail : yskwon@dongguk.edu
2006년 11월 접수, 2007년 08월 게재확정.

나 성능, 네트워크의 사용이 증가하면서 더 많은 알람 로그 데이터가 발생하고 있기 때문에, 보안전문가에 의한 수동적인 분석 방법은 제한적일 수밖에 없다.

따라서 본 논문에서는 침입탐지시스템의 문제점의 하나인 대량으로 발생되는 알람 로그 데이터 중에서 오탐 알람을 효율적으로 가려내기 위한 방법을 제안을 하고자 한다. 알람 로그 데이터에는 정상적인 네트워크 행위를 공격행위로 오인하여 발생한 알람 로그 데이터(false positive)들이 전체 알람 로그 데이터의 상당수를 차지하고 있다. 따라서 전체 알람 로그 데이터 가운데서 이와 같이 오인하여 발생한 알람 로그를 가려내어 분류만 할 수 있다면, 보다 손쉽게 판단을 할 수 있을 것이다. 이와 같은 점에 착안하여 클러스터링 기법 중에 하나인 AOI(Attribute-Oriented Induction) 알고리즘을 기반으로 하여, 효율적으로 침입탐지시스템의 오탐을 가려내고, 실제 공격 알람을 쉽게 발견할 수 있도록 하며, 보안 전문가나 관리자의 업무량을 크게 경감시키고자 한다.

1.3 연구의 구성

본 연구는 다음과 같이 구성되어 있다. 제 2장에서는 침입탐지시스템과 관련연구동향을 조사하고 본 논문에서 제안하는 시스템의 기반이 되는 AOI 알고리즘과 문제점에 대하여 살펴보았다. 제 3장에서는 제 2장에서 제기된 문제점을 해결할 수 있는 방안을 제안하고, 제 4장에서는 실험을 통하여 제안된 방법의 타당성을 보이고자 하였다. 제 5장에서는 본 논문의 결론 및 문제점들을 지적하고 향후 연구 방향에 대해 정리하였다.

2. 침입탐지시스템과 관련 연구

일반적으로 침입이란 어떠한 자원의 무결성(integrity)과 비밀성(confidentiality)과 가용성(availability)을 해치는 일련의 행위들을 지칭하며 이러한 침입의 행위들을 자동으로 탐지해 내는 시스템을 침입탐지시스템(intrusion detection system)이라고 한다 <그림 2-1>.

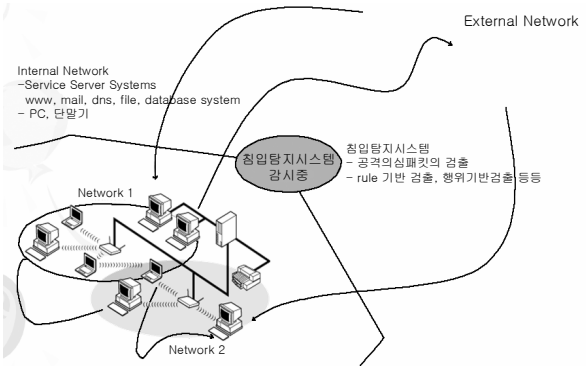

그림 2-1. 침입탐지시스템

침입탐지시스템은 크게 오용침입탐지시스템(misuse detection system)과 이상 상태침입(anomaly intrusion detection system)으로 나눌 수 있다. 오용침입탐지시스템은 시그너처기반(signature-based) 침입탐지시스템이라고도 한다. 이것은 네트워크의 트래픽 패턴 혹은 어플리케이션으로의 악의적 접근 시도 등의 일련의 패턴들을 알고 이것들을 식별하는 시스템이다. 이것을 통해 일반적인 공격 유형의 패턴을 알고, 시스템에서 일어나는 모든 동작 유형들을 기록한 로그들을 분석함으로써 정해진 유형의 침입을 탐지할 수 있다. 그러나 새로운 형태의 공격이나 알려지지 않은 패턴의 공격에는 취약할 수밖에 없는 단점을 가지고 있다(Magbag, 2004). 이와는 대조적으로 이상상태침입시스템은 위와 같은 일정한 패턴을 갖고 있지 않고 일련의 정상적인 패턴들과는 다른 패턴들을 탐지하여 분별하는 시스템을 말한다.

2.1 침입탐지시스템 알람의 특성

본 논문에서는 침입탐지시스템의 탐지 방법에 상관없이, 탐지된 알람 데이터 중에서 오탐을 가려내어, 보안 관리자의 판단 능력을 극대화 하고자 하는 것이다. 따라서 오탐에 대한 정이가 필요한 데, 본 논문에서의 오탐이란, 실제 공격이 아닌데 공격으로 잘못 판단된 로그(false positive)를 나타내고 있다. 통상적으로 침입탐지시스템의 오탐 원인은 몇 가지의 근본 원인에 의해 발생하며 이로 인한 잘못 판단된 알람(false positive)이 전체 알람의 90% 이상을 차지한다. 이런 오탐을 발생시키는 근본 원인은 다음과 같다.

- 시스템의 네트워크 관련 모듈이 깨져서 TCP/IP 스택에 이상이 생겼을 경우, 침입탐지시스템에서는 이로 인해 지속적으로 “Fragmented IP”관련 알람을 발생시킴.
- 환경설정이 잘못된 2차 DNS 서버의 경우 30분마다 DNS Zone transfer 처리를 1차 DNS 서버와 무한히 반복하게 되어 과도한 네트워크 트래픽 을 유발시키고 이로 인해 “DNS Zone Transfer”관련 알람을 발생시킴.
- 새로운 시스템 관리 프로그램을 개발하여, 임의의 포트를 사용할 경우 새로운 공격으로 잘못 판단하여 알람 로그를 발생시킴.
- 시스템 모니터링 툴은 시스템들이 제대로 작동하고 있는지 확인하기 위해 일정시간마다 ping패킷을 전송하지만 이것을 공격으로 잘못 판단하여 알람로그를 발생시킴.

이와 같이 단순한 원인으로 인해 발생한 알람은 실제적으로 의도된 공격 행위에 의한 것은 아니지만, 침입탐지시스템에서는 공격으로 잘못 오인하여 알람 로그를 남기게 된다. 특히, 이러한 오탐 알람 로그 데이터들은 실제 공격 행위와는 달리 단순하며 반복적이며 지속적으로 알람 로그를 남기는 경향을 띄고 있다. 따라서 이러한 특성들을 이용하여 정상적인 알람 로그들과 오탐 알람 로그들을 분별해 내야 한다. 만일 이와 같은

불필요한 알람 로그들을 효과적으로 분별해 낼 수 있다면, 오탐 로그들을 따로 모아 분석하여, 오탐 알람 로그 데이터를 발생시킨 근본 원인들을 찾아내어, 필터링 룰을 작성하여 걸러내거나, 물리적으로 그 근본 원인을 제거함으로 차후에 똑같은 원인으로 인해 같은 증상의 오탐 알람의 발생을 차단함으로써 알람의 분석 효율을 높일 수 있다(Julisch and Dacier, 2002; Julisch, 2001).

그러나 이때 물리적인 원인 제거 방법은 원인을 원천봉쇄하여 확실하고 안전하게 근본원인을 제거하여 오탐알람 문제를 해결할 수는 있겠지만, 주변 환경이나 여건에 따라 높은 비용이 요구된다. 필터링 룰을 이용하는 방법으로 <그림 2-2>와 같이 근본적인 원인을 제거하는 것이 아니라, 침입탐지 엔진에 적용함으로써 오탐 알람 발생을 막을 수 있으며 비용적인 측면에서 물리적인 방법보다 저렴하다. 이는 선택의 문제이며 우선순위와 환경요소에 따라 달라질 것 이다.

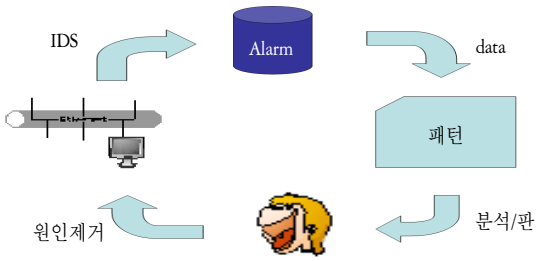

그림 2-2. 필터링 기법

침입탐지의 오탐율을 줄이기 위한 알람 로그 분석으로는 통계적 기법, 시각화 기법(Information Visualization), 데이터마이닝 기법 등이 연구되고 있다(Cuppens, 2001; Julisch, 2001, Sherif, 2002, Erbacher *et al*., 2002).

일반적으로 로그 메시지 생성 기능은 모든 보안과 관련된 이벤트(event)에 대하여 다량의 로그 메시지를 생성하기 때문에, 자연적으로 로그 데이터양의 증가는 분석 효율의 저하를 초래한다. 예를 들어 보안과 잠재적으로 관련 있는 파일에 접근할 때 생성되는 로그 데이터의 경우나 무수히 많은 합법적인 파일 접근을 침입으로 판단하여 기록하는 경우에는 악의적인 침입이라 할 수 있는 하나의 이벤트(event)를 찾아내는 것이 매우 어렵고 소모적인 작업이 된다. 그러나 대량의 유사한 알람 로그 데이터를 축약할 수 있다면 감시 대상 시스템들로부터 수집된 감사 데이터의 분석을 위한 부담을 줄일 수 있으며, 분석효율을 증가시킬 수 있다(한국정보보호진흥원, 2001).

본 논문에서는 오탐 알람의 원인이 되는 근본 원인 요소들을 구별해내기 위하여 AOI에 기반을 둔 클러스터링 기법을 사용하였다.

2.2 AOI(Attribute-Oriented Induction)를 이용한 알람 분석

AOI는 Concept Hierarchy로 정의된 속성 값들을 이용하는 데이터베이스 지식발견 기법(Knowledge Discovery in Database)으로, 데이터 서브셋(subset)의 속성치 들을 일반화 시키는 기법이다. 이를 통해서 일반화된 관계식으로 압축함에 따라 데이터베이스의 특성들을 쉽게 파악할 수 있다(Han and Fu, 1996).

AOI 알고리즘은 <그림 2-3>과 같다.

```
1: for all alarms a in T do a.C := 1;// Init counts
2: while table T is not abstract enough do {
3:       Select an alarm attribute Ai;
4:       for all alarm a in T do // Generalize Ai
5:             a.Ai := faher of a.Ai in Hi;
6:       while identical alarms a,a' exist do //Merge
7:             Set a.C := a.C+a'.C and delete a' from T;
8: }
```

그림 2-3. AOI 알고리즘

T는 입력할 관계테이블을 말하며, A1, A2, …, An은 테이블의 속성(attribute)을 가리킨다. H는 일반화시킨 계층구도를, d는 임계값을 나타낸다. C는 카운트 수를 나타낸다. 처음 초기 단계로 모든 알람의 C 속성 값으로 1을 지정하고 계속해서, 2단계에서 8단계까지 반복 루프를 돌게 되는데, 이때 3단계에서 적당한 속성 Ai를 고르게 되며, 4, 5단계에서 계속해서 일반화시킨 계층구도 H를 이용하여 Ai의 속성 값들을 더 일반화된 값으로 치환해간다. 그로 인해 이전에 다른 알람이 똑같은 알람으로 변화하게 된다면 하나의 알람을 삭제를 하고 나머지 하나의 카운트 C 값으로 통합하고 이러한 단계를 임계치 값까지 계속 반복한다.

Julisch and Dacier(2002)는 IP 주소, 포트 번호, 알람 타입 등의 범주를 갖는 속성을 이용해 알람데이터를 분석하여 AOI의 적용가능성을 제시하였다. 예를 들어, AOI를 이용하여 관계데이터베이스 테이블에서 계속해서 좀 더 추상적인 값으로 속성의 값들을 치환해 나간다. 더 추상적인 값이란 사용자가 정의하여 생성된 계층구도를 이용하는 것으로, <그림 2-4>에서와 같이 각 IP들은 어느 한 네트워크에 속하게 되고, 포트 번호 역시 어느 특정 범위에 속하게 된다. 각각의 계층 트리를 이용하여 관련 데이터베이스 테이블의 속성 값들을 계속해서 더 추상적인 값으로 일반화시킴으로 해서, 이전에는 달랐던 알람이 동일한 값의 속성이 되고, 그로 인해 하나로 합쳐진다. 이런 방식으로 대용량의 데이터베이스 테이블들을 간략히 축약하면서도 높은 요약된 정보를 갖는 결과물을 생성해 낼 수 있게 된다.

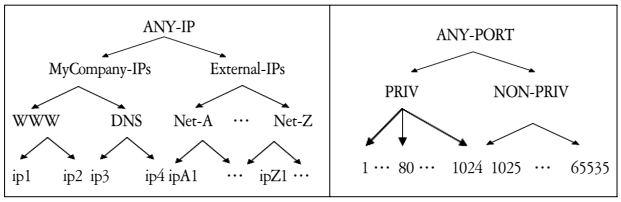

그림 2-4. 계층화 시킨 IP와 Port 번호

이와 같은 과정을 거쳐 산출된 결과는 <그림 2-5>와 같이 전체적 알람 로그에 대한 요약을 보이고 있다. 이를 근거로 보안 전문가가 알람의 근본 원인을 찾아내게 된다.

그러나 AOI 알고리즘의 결과에는 많은 양의 데이터 집합 가운데서 예외 데이터들이 많이 포함될 가능성이 있다는 문제가 발생한다. 이러한 문제는 예를 들어, <그림 2-4>의 ip1에 관련된 알람데이터가 존재하지 않더라도, AOI 알고리즘은 계속해서 보다 큰 개념으로 추상화 시켜 나가기 때문에, 이를 포함하는 결과 값으로 나타나게 된다.

다른 문제점으로는, AOI를 적용하여 나온 결과가 단순히 반복적으로 발생하는 오탐인지, 한 순간에만 이루어진 공격인지 구분하기가 어렵다는 점이다. 대부분의 경우, 오탐일 가능성은 높지만, 이에 대한 정확한 판단이 필요하다.

AT	Src-Port	Src-IP	Dst-Port	Dst-IP	Time	Contest	Size
1	NON-PRIV	EXTERN	80	ip4	any	see text	54310
1	NON-PRIV	EXTERN	80	ip5	any	see text	54013
1	NON-PRIV	FIREWALL	80	EXTERN	any	any	17830
2	NON-PRIV	FIREWALL	21	EXTERN	any	any	6439
2	NON-PRIV	EXTERN	21	WWW/FTP	any	any	4181
3	undefined	ip6	undefined	ip1	WORKDAY	undefined	4581
3	undefined	ip6	undefined	ip2	WORKDAY	undefined	3708
4	NON-PRIV	ip1		any	any	undefined	761
4	NON-PRIV	ip2		any	any	undefined	663
4	NON-PRIV	FIREWALL		any	any	undefined	253
5	undefined	EXTERN	undefined	ip4	any	undefined	823
5	undefined	EXTERN	undefined	ip5	any	undefined	711
6	undefined	ip7	undefined	FIREWALL	END-OF MONTH, TUESDAY	undefined	861

Alarm Type(AT): 1≡“www TIS View Source Attack”; 2≡“FTP SYST Command Attempt”; 3≡“IP Fragment Attack”; 4≡“TCP SYN Host Sweep”; 5≡“Fragmented ICMP Yraffic”; 6≡“UnKnown Protocol Field IP Packer”;

그림 2-5. AOI 알고리즘을 이용한 분석 결과

3. AOI에 기반을 둔 개선된 알람 분석 방법 제안

본 연구에서는 제 2장에서 언급된 YOGI 방법의 문제점을 해결하기 위한 방안을 제시하고 실험으로 타당성을 보이고자 한다.

3.1 서비스 제공 시스템과 클라이언트 시스템의 분리

일반적인 인트라 네트워크 트래픽은 같은 종류의 OS를 사용하는 개인 PC 사용자에 의해 대부분 발생한다. 이로 인해 새로운 바이러스나 해킹기법의 출현 시에 상대적 파급효과가 크며, 개인 PC 의 사양과 성능이 좋아지면서, 개인 PC 가 또 다른 2차 공격을 위한 경유지 역할을 할 수도 있기 때문에, 최근에는 해킹의 대상이 서버에서 개인 PC로 변화하고 있는 실정이다 (한국정보보호진흥원, 2004). 따라서 개인 사용자용 시스템과 서비스 제공 시스템을 분리한 후에, 각각의 클래스 군의 분석이 필요로 한다. 대부분의 기업체의 정보 시스템 사용현황을 보면, 서비스 시스템에 비해 개인 사용자용 시스템의 수가 월등히 많기 때문에, 이와 같은 구분 관리는 서비스 제공 정보 시스템의 보호뿐만이 아니라, 개인 사용자 PC의 안전까지 도모할 수 있을 것이다. 또한, 시스템분리를 통해 외부에서의 공격이 아닌 내부자 공격 발생 시, 쉽게 근원지를 찾을 수 있다는 장점이 있으며 발생하는 네트워크 트래픽 과다 발생의 원인을 제거하여 안전한 네트워크 환경을 유지 관리 할 수 있다. 예를 들어, 인사과, 총무과, 영업과가 있는 회사의 경우 인사과의 사용자중 한사람이 감염된 파일을 다운로드 받아 개인 사용자 컴퓨터에 Worm 바이러스가 감염되었다고 할 때, 그 감염된 컴퓨터가 시발점이 되어 같은 인사과의 컴퓨터에 공격 트래픽을 계속 해서 발생시킬 것이고, 점점 다른 과로 확산을 시키려고 시도를 하게 된다. 이와 같은 상황에서 침입탐지시스템의 알람을 AOI의 속성을 부서별로 분류를 해놓고 적용한다면, 인사과에서 다른 과로 알람 로그가 계속 해서 발생한다는 것을 알 수 있어, 네트워크 관리자는 비교적 용이하게 근원지를 알아내어 조치를 취할 수 있을 것이다.

3.2 시간에 따른 알람의 발생 빈도 확인

AOI알고리즘은 미리 지정된 임계값까지 연산을 하여 결과물을 낸다. 그러나 결과 값은 Attribute 값들이 계속해서 추상화된 형태의 값이기 때문에, 경우에 따라서는 예외적인 결과가 포함될 가능성이 있다. 예를 들어, 시간 속성을 다음과 같이 적용을 하면, 다음 <그림 3-1>과 <그림 3-2>에서 보는 바와 같이 다른 형태의 공격 알람이 같은 것으로 처리될 가능성이 있다. <그림 3-2>에서 왼편 그래프는 인위적인 형태의 공격일

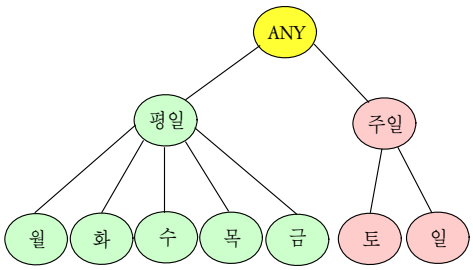

그림 3-1. 시간 속성 계층도

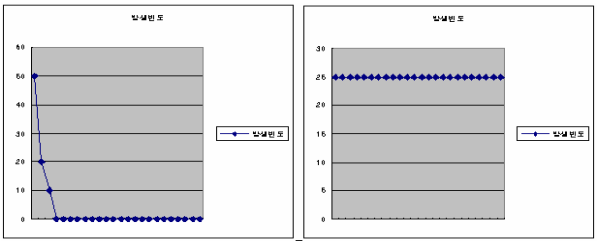

그림 3-2. 시간에 따른 발생 빈도

가능성이 오른편의 그래프 보다 높다. 단기간동안 어떠한 공격 행위가 존재하고 그 이후부터는 발생하지 않았기 때문에, 이 경우, 공격시도를 인위적으로 했다고 볼 수 있다. 반면에 오른편 그래프의 경우 어떤 종류의 알람인지 모르지만 시간 흐름에 따라 일정한 반복이 되풀이 되고 있기 때문에, 실제의 공격이라고 보기 보다는, 환경 설정의 오류 또는 자동화된 단순한 공격이라고 볼 수 있다.

이와 같이 AOI 기법은 적용할 속성 값을 어떻게 추상화 시켜놓았는가에 따라 위의 정보가 똑같은 것으로 간주되어지기도 하고, 다른 것으로 간주되어지기도 한다. 이런 문제점의 해결을 위해서, 시간변화에 따른 알람 발생 빈도를 조사함으로써, AOI 알고리즘의 문제점을 보완할 수 있다.

3.3 실시간 분석을 위한 슬라이딩 윈도우

알람로그는 이벤트(event)가 발생한 후에 생성 되는 것이기 때문에 실시간으로 알람로그를 분석한다는 것은 가능하지 않다. 그러나 어떠한 공격이나 침입이 발생 후, 빠른 시간 내에 탐지하는 것이 필요하기 때문에 본 연구에서는 <그림 3-3>에서와 같이 알람로그의 일정량을 정하여 순차적으로 알고리즘에 적용하는 슬라이딩 윈도우 방식의 활용가능성을 제안하였다.

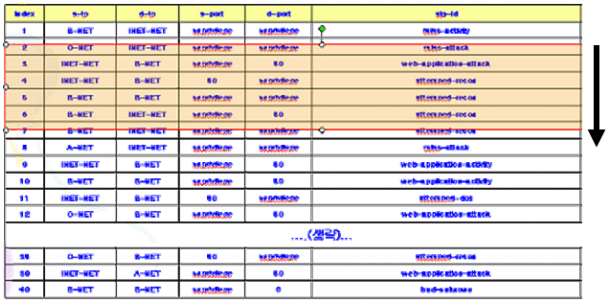

그림 3-3. 슬라이딩윈도우를 이용한 분석

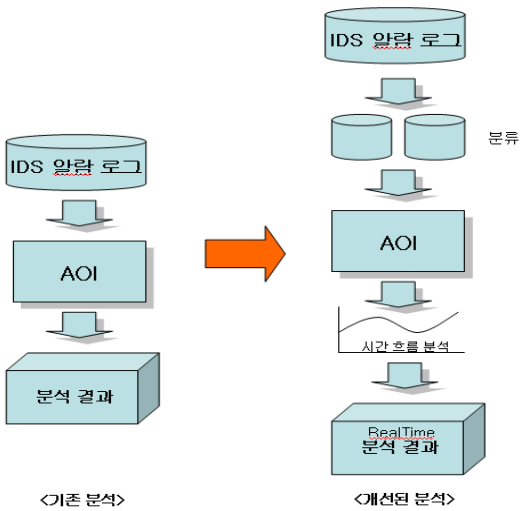

그림 3-4. 제안된 알람로그 분석절차

본 논문에서는 이와 같이 세 가지의 제안사항에 대하여 실험을 하였다. 첫 번째 제안은 적용할 데이터 셋의 개인 정보 보호를 위해 본 연구에서는 다루지 않고 나머지 두 가지의 제안에 대하여 실험을 실시하였다.

제안된 알람로그분석 절차는 <그림 3-4>와 같다.

4. 실험 결과

실험은 편의상, 적은 량의 데이터를 이용한 축소실험과 현업에서 확보한 대용량의 데이터를 이용한 확대실험등 2가지의 다른 환경의 실험을 실시하여 제안된 방법의 현실적인 활용가능성을 제시하였다.

4.1 축소실험

4.1.1 실험자료

실험 자료는 A대학교에 설치한 네트워크 침입탐지시스템인 SNORT(www.snort.org) 시스템에서 6개월간 발생한 알람로그를 사용하였다. 총 데이터 수는 92,991건으로 AOI 기법을 적용하기 위한 속성 계층도는 <그림 4-1>과 같은 IP에 대해서만 적용을 하였다. 실험 데이터 및 결과물에 대해서는 정보보안를 위해 임의로 알파벳이나 가나다 등으로 대치를 하였다.

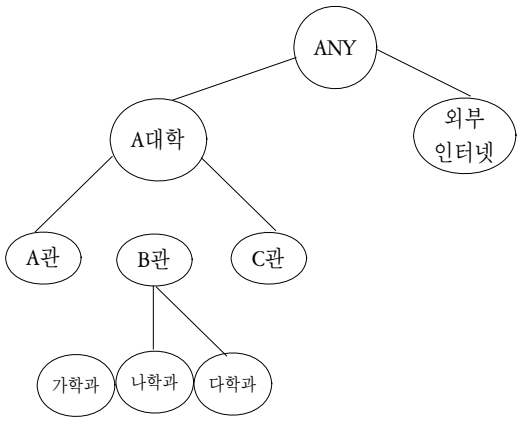

그림 4-1. 적용 IP 계층도(1)

본 실험에서는 단순히 시발지 IP 주소와 목적지 IP 주소에 대해서만 적용을 하였지만, 로그 분석의 목표에 따라 속성 계층도를 달리 적용할 수 있다. 다음 장에서 다루는 확대실험에서는 네트워크 구조가 아닌 분석을 원하는 대역에 대해서만 속성 계층도를 달리하여 실험 적용을 하였다. 알고리즘에 적용할 알람 데이터양은 일주일 분량으로 하였는데, 이는 입력 데이터양의 조절과 실시간 적용을 위해 적용 알고리즘 데이터양을 일주일분량으로 한 것이다. 본 실험에서는 실험결과의 변화도를 보여주기 위해, 하루(24시간)간격으로 슬라이딩윈도우를 설정하였다. 알고리즘을 구현하기 위해서 사용한 시스템

환경을 듀얼라틴 Celeron-1.3G, 512memory의 시스템, windows 2000 운영체제, JAVA 및 python으로 구성을 하였다.

4.1.2 실험 결과

실험결과 중에서 1주차부터 8주차 결과물만을 <표 1>,<그림 4-2>에 나타내었다. 기존의 AOI를 이용하여 알람을 클러스터링 한 경우, <표 1>에서 2주차 자료의 경우에는, "나 학과 ==> A관"과 "외부 ==> 나 학과"의 경우가 잘못된 판단 로그로 인식된다. 즉, 다른 알람들에 비해 카운트수가 큰 두 개의 알람이 오탐 알람일 가능성이 매우 크다고 인식을 하게 된다.

그러나 단순히 카운트만을 사용하여 판단을 하는 것에서 끝나지 않고, 제안한 바와 같이 집중된 알람 결과에 대해 시간변

표 1. 실험결과(1)

1주차 ip : 나 학과 ==> A 관 count : 707 ip : C 관 ==> 나 학과 count : 108 ip : 외부 ==> 나 학과 count : 375 ip : 다 학과 ==> 외부 count : 960 ip : 나 학과 ==> 나 학과 count : 39 ip : 나 학과 ==> 외부 count : 87 ip : 나 학과 ==> C 관 count : 2 ip : 가 학과 ==> 나 학과 count : 2 ip : 다 학과 ==> 나 학과 count : 32 ip : C 관 ==> 가 학과 count : 1 ip : 외부 ==> 다 학과 count : 1 ip : 외부 ==> 가 학과 count : 3 Alarms Size =12	2주차 ip : 나 학과 ==> A 관 count : 813 ip : 외부 ==> 나 학과 count : 235 ip : C 관 ==> 나 학과 count : 40 ip : 나 학과 ==> 나 학과 count : 1 ip : 나 학과 ==> 외부 count : 17 ip : 외부 ==> 가 학과 count : 4 ip : 외부 ==> C 관 count : 9 Alarms Size =7
3주차 ip : 나 학과 ==> A 관 count : 983 ip : 외부 ==> 나 학과 count : 249 ip : 외부 ==> 가 학과 count : 1 ip : 다 학과 ==> 외부 count : 240 ip : 나 학과 ==> 나 학과 count : 72 ip : 나 학과 ==> 외부 count : 44 ip : 나 학과 ==> 가 학과 count : 12 ip : 나 학과 ==> C 관 count : 7 ip : C 관 ==> 나 학과 count : 18 ip : 가 학과 ==> 나 학과 count : 3 Alarms Size =10	4주차 ip : 나 학과 ==> A 관 count : 553 ip : 외부 ==> 나 학과 count : 205 ip : 나 학과 ==> 나 학과 count : 8 ip : 나 학과 ==> 외부 count : 102 ip : C 관 ==> 나 학과 count : 64 ip : 외부 ==> 가 학과 count : 12 ip : 외부 ==> C 관 count : 6 ip : 다 학과 ==> 외부 count : 620 ip : 외부 ==> 외부 count : 2 ip : 가 학과 ==> 외부 count : 74 ip : 가 학과 ==> 나 학과 count : 3 Alarms Size =11
5주차 ip : 나 학과 ==> A 관 count : 253 ip : 외부 ==> 나 학과 count : 422 ip : 외부 ==> 가 학과 count : 2 ip : C 관 ==> 나 학과 count : 60 ip : C 관 ==> 가 학과 count : 1 ip : 나 학과 ==> 외부 count : 745 ip : 다 학과 ==> 외부 count : 320 ip : 나 학과 ==> 나 학과 count : 24 ip : 외부 ==> 외부 count : 110 ip : 외부 ==> C 관 count : 19 ip : 가 학과 ==> 나 학과 count : 2 ip : 나 학과 ==> 가 학과 count : 1 Alarms Size =12	6주차 ip : 나 학과 ==> 외부 count : 2005 ip : 외부 ==> 나 학과 count : 140 ip : 나 학과 ==> A 관 count : 23 ip : 나 학과 ==> 가 학과 count : 250 ip : 나 학과 ==> 나 학과 count : 109 ip : 나 학과 ==> 다 학과 count : 65 ip : 가 학과 ==> 나 학과 count : 30 ip : 다 학과 ==> 나 학과 count : 197 ip : 나 학과 ==> C 관 count : 254 ip : 다 학과 ==> 외부 count : 124 ip : 외부 ==> 외부 count : 90 ip : C 관 ==> 나 학과 count : 2 ip : 외부 ==> 가 학과 count : 3 ip : 외부 ==> C 관 count : 5 Alarms Size =14
7주차 ip : 다 학과 ==> 나 학과 count : 304 ip : 나 학과 ==> 외부 count : 336 ip : 외부 ==> 나 학과 count : 268 ip : 외부 ==> 가 학과 count : 13 ip : 나 학과 ==> 나 학과 count : 31 ip : 다 학과 ==> 외부 count : 47 ip : 외부 ==> C 관 count : 1 ip : 가 학과 ==> 나 학과 count : 14 ip : 가 학과 ==> 외부 count : 107 ip : 외부 ==> 외부 count : 1 Alarms Size =10	8주차 ip : 다 학과 ==> 나 학과 count : 309 ip : 나 학과 ==> 외부 count : 775 ip : 외부 ==> 가 학과 count : 9 ip : 외부 ==> 나 학과 count : 387 ip : 외부 ==> 다 학과 count : 1 ip : 외부 ==> C 관 count : 4 ip : 외부 ==> 외부 count : 30 ip : 나 학과 ==> 나 학과 count : 51 ip : 가 학과 ==> 나 학과 count : 3 ip : 다 학과 ==> 외부 count : 78 ip : C 관 ==> 나 학과 count : 36 ip : 가 학과 ==> 외부 count : 26 ip : C 관 ==> 가 학과 count : 2 ip : A 관 ==> 가 학과 count : 3 ip : A 관 ==> 나 학과 count : 2 ip : A 관 ==> 외부 count : 1 Alarms Size =16

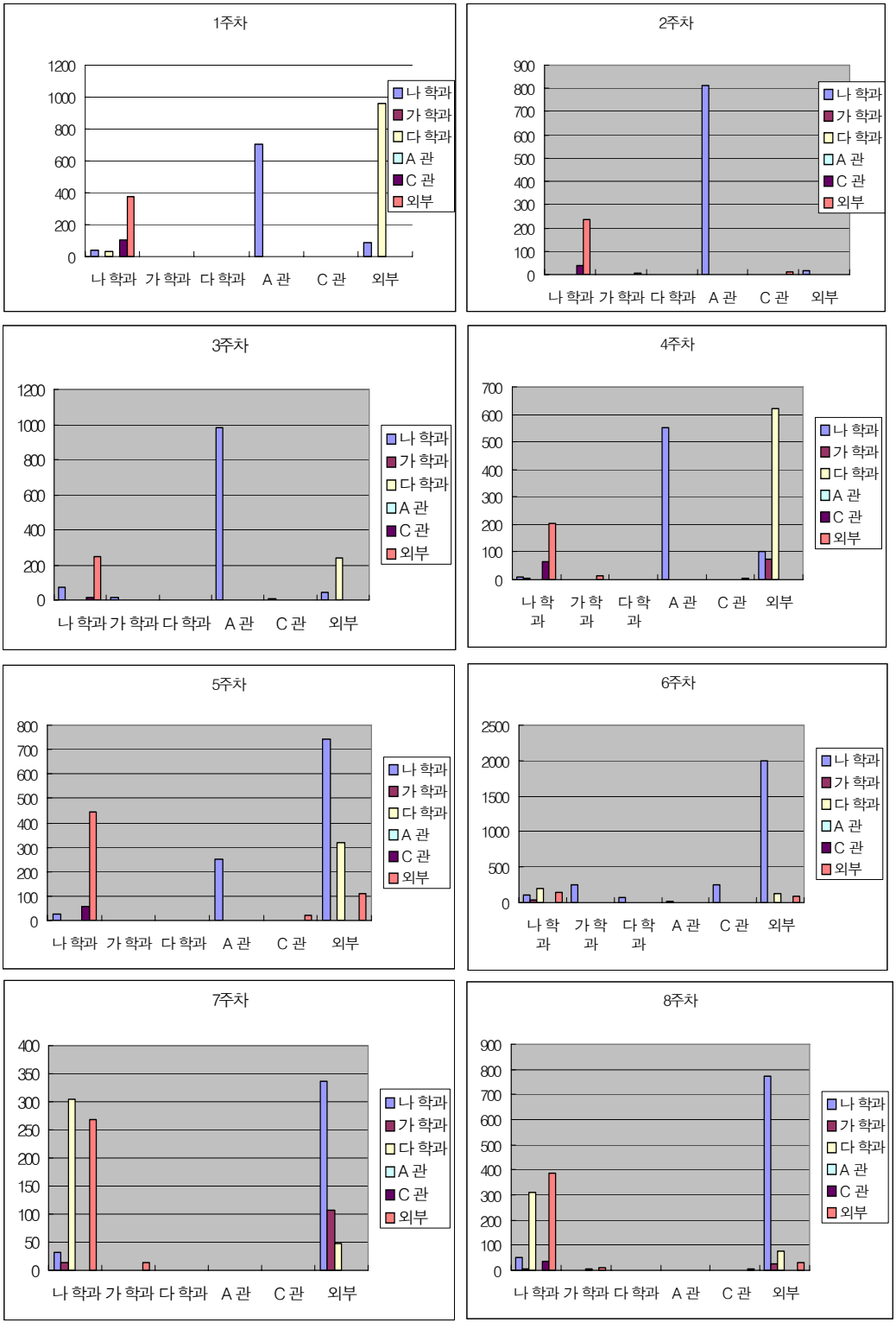

그림 4-2. 실험결과(2)

화에 따른 알람 발생 빈도를 조사해 봄으로써 더욱 정확한 분석을 할 수가 있었다. 예를 들어, <표 1>의 2주차 알람 데이터 중 첫 번째와 두 번째 알람 로그를 시간차로 분석을 해보면, <그림 4-3>과 같다.

<그림 4-3>에서 좌측 그래프들은 일별로 발생한 알람 빈도수를 그래프화 시킨 것이고, 우측 그래프들은 18일에 해당하는 알람 빈도수를 그래프화 시킨 것이다. 이 그래프에서 "나 학과 ==> A 관"알람의 경우는 반복적으로 알람이 발생하는 것을

알 수 있다. 따라서 이 알람은 반복적이며 지속적인 잘못된 판단에 근거해 발생한 알람로그임을 알 수 있다. 반면에 “외부 ==> 나 학과” 알람의 경우는 시간의 변화에 따라 알람의 종류가 다르거나, 매우 불규칙적으로 변화하는 알람이므로 보다 자세한 분석이 필요함을 알 수 있다.

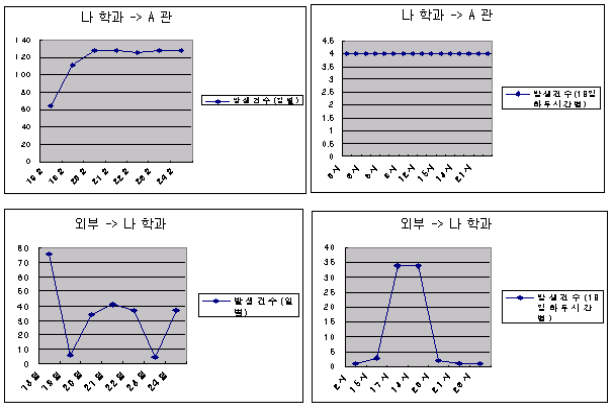

그림 4-3. 시간에 따른 변화

이와 같이 AOI 기법을 이용하여 생성된 결과물을 가지고 시변화에 따른 알람 로그 발생 빈도를 확인함으로써 보다 정확한 분석이 가능하게 된다.

마지막으로 실시간 분석이 가능한지 여부를 위해서 일정한 량의 알람 데이터를 일정 시간 간격으로 해서 슬라이딩 윈도우 방식으로 적용을 해본 결과 지정하는 시간 간격에 따라 다르지만, 대체로 부드럽게 표현이 가능하였다.

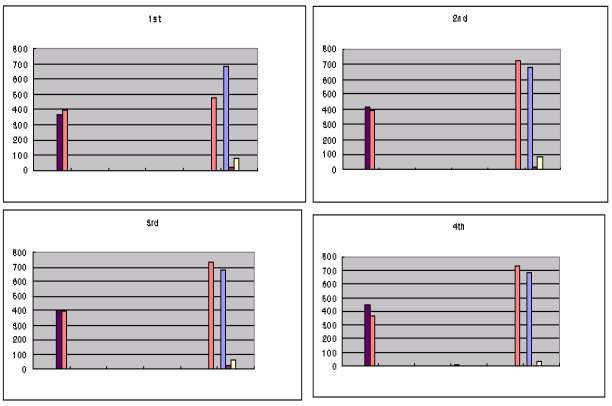

그림 4-4. 실시간 분석

4.2 확대실험

4.2.1 실험자료

두 번째 실험은 모 보안연구소에서 제공한 현업에서의 알람 데이터를 이용하여 실험을 하였다. 현업에서는 하루에도 약 50만 건 이상이 되는 엄청난 데이터가 쌓이고 있다. 이를 처리하기 위해서는 시스템의 성능 및 감시 네트워크 구조의 이해가 필요하다. 하지만, 보안상의 이유로 실제 적용된 곳의 시스템 구성을 알 수가 없었기 때문에, 본 연구에서는 간단히 기본 네트워크 IP 대역을 나누어, 관심이 있는 네트워크를 임의로 구성하여 알고리즘을 적용하였다. 본 실험에서도 역시 시발지 IP 주소와 목적지 IP 주소에 대해서만 적용을 하였지만, 이는 로그 분석의 목표에 따라 속성은 달리 적용할 수 있다.

이번 실험에서는 앞에서 수행한 축소실험과 달리 나무(tree) 형태의 속성 계층을 만들지 않고 단순히 관심 있는 네트워크에 대역에 대해서만 각각 그룹으로 묶어서 실험을 하였다.

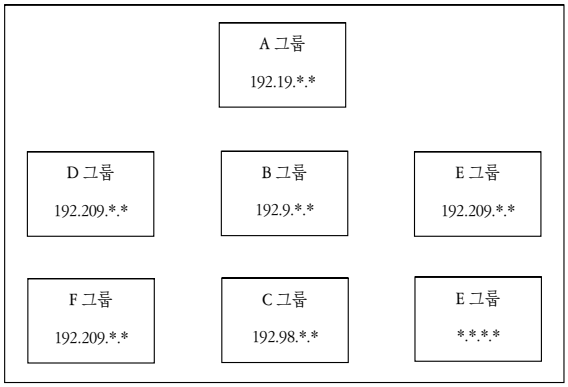

그림 4-5. 적용 IP 계층도(2)

이렇게 함으로써 관심 있는 네트워크에 대해서 중점적으로 알람 로그 데이터를 분석할 수 있게 된다. 이는 상황에 따라 속성 계층도를 필요에 맞게 수정할 수 있기 때문에 향후 실제 적용이 가능한 방법이다.

4.2.2 실험 결과

실험 결과 약 500,000건의 알람 데이터가 <그림 4-6>과 같은 형태로 약 330여 개의 축약된 클러스터를 얻을 수 있었다. AOI 알고리즘은 어느 정도의 임계치까지 계속해서 알고리즘을 수행하느냐에 따라, 결과물이 합쳐지기도 하고 분리되기도

```
[Hacker=192.98.8.99], [B]          ------>      318
[Hacker=192.9.7.39], [C]           ------>      37
[Hacker=192.98.8.100], [A]         ------>      403
[Hacker=192.98.8.70], [A]          ------>      413
[Hacker=192.98.8.101], [A]         ------>      436
[Hacker=192.9.7.33], [C]           ------>      495
[Hacker=192.98.8.87], [B]          ------>      641
[Hacker=192.9.7.34], [C]           ------>      644
[Hacker=192.98.8.113], [A]         ------>      674
[Hacker=192.98.8.115], [A]         ------>      678
[Hacker=192.98.8.70], [B]          ------>      687
[Hacker=192.98.8.120], [A]         ------>      698
[Hacker=192.98.8.88], [B]          ------>      698
[Hacker=210.124.9.125], [F]        ------>      703
[Hacker=192.98.8.86], [B]          ------>      716
[Hacker=192.98.8.90], [B]          ------>      731
[Hacker=192.98.8.114], [A]         ------>      737
[Hacker=192.98.8.111], [A]         ------>      756
[Hacker=192.19.7.50], [C]          ------>      767
[Hacker=192.9.7.30], [C]           ------>      791
[Hacker=192.98.8.89], [B]          ------>      798
[Hacker=192.98.8.87], [A]          ------>      833
[Hacker=192.98.8.99], [A]          ------>      838
[Hacker=192.98.8.89], [A]          ------>      851
[Hacker=192.98.8.90], [A]          ------>      862
[Hacker=192.98.8.86], [A]          ------>      864
[Hacker=192.98.8.88], [A]          ------>      873
[Hacker=192.98.8.118], [A]         ------>      931
[Hacker=192.98.8.117], [A]         ------>      949
[Hacker=192.98.8.210], [C]         ------>      1086
[Hacker=192.98.8.120], [B]         ------>      1493
[Hacker=192.98.8.114], [B]         ------>      1494
[Hacker=192.98.8.115], [B]         ------>      1602
[Hacker=192.98.8.116], [D]         ------>      1741
[Hacker=192.98.8.111], [B]         ------>      1763
```

그림 4-6. 적용 결과물(2)

하므로 임계치의 결정은 상황에 따라 운영자가 적절하게 조절을 하게 된다.

첫 번째의 축소실험과 달리 대용량의 현업의 데이터를 사용할 때는, 알고리즘 적용을 위한 전처리 작업이 더욱 많은 시간을 할애하였으며, 노이즈를 제거하는 데 많은 노력을 기울였다. 그 외의 모든 실험 환경이나 설정들은 앞에서 수행한 축소실험과 같다.

출력 데이터는 공격자의 주소에서 목적지 주소로 향한 정보를 담고 있는 알람들의 출력물과 그 알람의 개수를 알려주고 있다. 결과물 중에 "[Hacker = 192.98.8.219], [C] -----> 15948"이라는 결과물은 192.98.8.219의 시발지에서 C 그룹을 향한 공격 알람이 감지되었음을 나타내며, 그 수가 15948건이 된다는 것을 나타낸다.

더 세밀한 분석을 위해 시간에 따른 분석을 한 결과 <그림 4-7>과 같은 결과를 얻을 수 있었다.

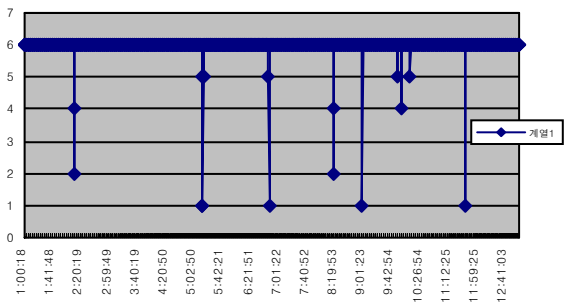

그림 4-7. 시간 흐름에 따른 변화

다른 알람들에 비해서 유일하게 그래프는 시간의 흐름에 따라 거의 차이가 없이 일정한 량의 알람을 일으키고 있음을 나타내고 있다. 이를 원본 데이터를 통해 확인을 한 결과 모두 일정한 스캐닝을 시도한 것을 나타났다. 이는 실제 공격 시도에 대한 알람 로그 데이터라 생각을 할 수도 있겠지만, 그 보다는 네트워크 환경이나 다른 요건으로 인해서 일상적으로 발생하는 알람일 가능성이 크다. 따라서 이 경우 알람발생 원인을 규명을 해볼 필요가 있음을 알 수 있다.

결론적으로 제안된 방법은 기존의 AOI 기법보다 정확한 판단의 근거로 활용 할 수 있는데, 이는 보다 일반적인 개념으로 그룹핑을 시키는 과정에서 생길 수 있는 데이터의 합침 현상으로 인해, 결과 데이터에 불필요한 정보까지 포함을 할 수 있는 문제점을 시간 흐름에 따른 분석을 통해서 보완을 할 수 있었기 때문이다.

5. 결론 및 향후 연구 과제

본 논문에서는 지금까지 연구된 침입탐지시스템에 대해 소개하였고, 기존의 침입탐지시스템의 알람로그들의 특성 및 분석기법으로는 어떠한 것들이 있는지와 기존의 분석기법의 문제점이 무엇인지 알아보고, 그에 대한 해결책을 제안하였다.

연구를 통해서 우리는 대량의 원본 알람 로그 데이터로부터 소량의 알람으로 축약을 시킬 수 있었으며, 보다 정확한 분석을 위해서 적극적으로 시간흐름에 따른 빈도를 조사하여 기존보다 오탐 알람을 쉽게 발견할 수 있었다. 이는 다시 결과 알람 데이터의 수를 더욱 줄이는데 기여를 하게 되며, 결과적으로 침입탐지 알람을 분석하는 전문가의 수고를 덜어 주게 되었다. 이를 통해 실제적인 공격에 더욱 집중을 할 수 있게 되기 때문에, 네트워크 보안에 크게 기여할 수 있었다. 또한 알고리즘에 적용할 속성 계층도를 다양한 네트워크 환경에 따라 적당히 변형하여 적용함으로 해서 다양한 시각에서의 분석이 가능하게 되었다. 이러한 분석 기법은 단순한 로그 분석 분야뿐만이 아니라, 침입탐지시스템 엔진의 커스터 마이징등 다양한 분야에 쓰일 수 있을 것이다.

하지만, 아직 완전한 자동화 시스템을 구축할 수는 없었고, 관리자의 수고를 덜어주는 역할 정도에 지나지 않는다. 향우에는 보다 자동화된 시스템을 개발할 필요성이 있으며, 분석 시스템을 통해서 발견된 오탐 알람의 근본 원인을 찾아내어, 물리적으로 해결을 하거나, 상황에 따라 침입탐지시스템의 탐지 룰을 수정하는 등의 정책적인 문제가 남아 있다.

모든 시스템은 로그를 남기게 되어있다. 그 중에서 침입탐지시스템의 로그 기록은, 단순한 네트워크 트래픽 기록이 아니라, 공격 현상을 탐지한 공격 트래픽 기록이다. 따라서 이벤트(event) 발생 후에 감사 자료로 사용될 수 있으며, 새로운 형태의 공격도 발견해 낼 수 있다.

네트워크의 특성상 모든 시스템에서 똑같은 형태의 알람이 발생하지는 않는다. 따라서 보안 전문가의 능력에 따라 안전한 네트워크 구축이 좌우되는 현시점에서, 보안 전문가의 능력을 최대한 발휘할 수 있도록 도와주는 침입탐지시스템 알람 로그 분석기법이 꼭 필요하다. 본 논문은 단순히 침입탐지시스템의 알람로그를 분석하기 위한 기법을 제안하였지만, 단순히 한가지의 기법을 이용해서 로그 데이터를 분석하는 것보다 다양한 기법을 동시에 적용하여 다각도의 시점에서 알람 로그 데이터를 분석할 필요가 있다. 특히 요즘은 통합 보안 시스템의 역할이 증대하고 있는 시점에서, 기존의 모든 보안 시스템을 통합해서 관리를 할 필요가 있다. 이런 상황에서는 각 보안 시스템 각각에 가장 적합한 로그 분석 기법을 개발을 할 필요가 있다. 모든 정보 시스템은 관리가 가장 중요하다. 정보보안에 있어서 완성이란 개념은 존재하지 않는다. 계속적이고 꾸준한 관리만이 진정한 보안에 이룩하는 것임을 잊지 않아야 할 것이다.

참고문헌

Berry, Michael, J. A. and Linoff, G. (1999), Mastering Data Mining, John Wiley

& Sons.
Bloedorn, Eric (2000), Data Mining for Improving Intrusion Detection, MITRE.
Clifton, Chris and Gengo, Gary (2000), Developing Custom Intrusion Detection Filters using Data Mining, Proceedings of MILCOM 2000, 440-443.
Cuppens, Frederic (2001), Managing Alerts in a Multi-Intrusion Detection Environment, Proceedings of the 17th ACSAC 2001.
Ellis, J., Hayes, E., Marella, J. and Willke, B. (2002), State of the Practice of Intrusion Detection Technologies, Technical Report, SEI, Carnegie Mellon University.
Erbacher, R. F., Walker, K. L. and Frincke, D. A. (2002), Intrusion and Misuse Detection in Large-Scale Systems, IEEE computer Graphics and Applications, 38-48.
Han, J. and Fu, Y. (1996), Exploration of the Power of Attribute-Oriented Induction in Data, Advances in Knowledge Discovery and Data Mining.
Han, J. and Kamber, M. (2001), Data Mining Concepts and Techniques, Morgan Kaufmann.
Han, J., Cai, Y. and Cercone, N. (1992), Knowledge Discovery in Databases: An Attribute-Oriented Approach, Proceedings of the 18th International Conference on Very Large Databases, 547-559.
Julisch, K. (2000), Dealing with False Positives in Intrusion Detection, In Extended Abstract, the 3rd Workshop on Recent Advances in Intrusion Detection (RAID), (http://www.raid-symposium.org/raid2000/program.html).
Julisch, K. (2001), Mining Alarm Clusters to Improve Alarm Handling Efficiency, Proceedings of the 17th Computer Security Applications Conference, 12-21.
Julisch, K. (2002), Clustering Intrusion Detection Alarms to Support Root Cause Analysis, ACM, 2(3), 111-138.
Julisch, K. and Dacier, M. (2002), Mining Intrusion Detection Alarms for Actionable Knowledge, Proceedings of 8th SIGKDD, 366-375.
Korea Information Security Agency (2001), Intrusion Detection System Estimate Standard, Report, 2001-12.
Korea Information Security Agency (2003), 2003 Information Security Industry Survey, Report, 2003-12.
Korea Information Security Agency (2004), Hacking/Virus Statics and analysis , Report, 2004-12.
Magbag, Sheilla D. (2004), A Survey of Misuse Intrusion Detection, (Seminar bstract). UPLB-ICSwebpage (http://www.ics.uplb.edu.ph/node/143).
Ministry of Information And Communication Republic of Korea (2005), Diving in IT 2005 Numerically, Report, 2005-12.
Sherif, J. S. (2002), Intrusion Detection: Systems and Models, Proceedings of the 7th IEEE International workshop, 115-133.

정 인 철
시립인천대학교 학사
동국대학교 산업공학과 석사
현재: 동국대학교 산업공학과 박사과정
관심분야: 데이터마이닝, Knowledge Discovery, Semantic Web

권 영 식
서울대학교 산업공학 학사
한국과학기술원 공학석사
한국과학기술원 공학박사
현재: 동국대학교 산업시스템공학과 교수
관심분야: 데이터마이닝, Business Intelligence, IT서비스