트래픽 분석에 의한 n-차원 벡터 기법을 사용한 침입 탐지

신수복*, 김원일**, 예홍진*
*아주대학교 정보통신전문대학원
**세종대학교 전자정보공학
e-mail : watermel@ajou.ac.kr

Intrusion Detection Method that uses N-demension Vector Technique by Traffic Analysis

Soo-bok Shin*, *Won-il Kim*, Hong-jin Yeh *
*Graduate School of Information Communication, Ajou University
**College of Electronics and Information Engineering, Sejoug University

요 약

인터넷의 급속한 발달과 그 중요성이 날로 증가하면서 인터넷은 우리생활의 일부가 되었다. 따라서 네트워크에서 환경에서의 침입은 빠르게 증가하고 있으며 그 피해 또한 커지고 있다. 서비스 거부공격, 분산서비스거부 공격, 웜 등은 악의적인 의도로 호스트나 네트워크의 자원을 고갈 시키는 등 심각한 문제를 초래하고 있다. 또한 중요한 서버의 공격으로 인해 시스템이 다운되면 서비스를 하지 못하게 됨으로 사용자에게 불편을 초래할 뿐만 아니라 막대한 비용의 손실도 발생한다. 따라서 본 논문에서는 벡터를 이용하여 지역 네트워크망으로 들어오는 트래픽의 비정상 증가를 분석함으로 침입을 탐지하고 위험수위를 결정하는 방법을 제안한다.

1. 서론

오늘날 인터넷의 발달과 인터넷 사용자가 급증하면서 인터넷의 중요성이 증가하였다. 또한 네트워크를 이용한 산업이 커지고 네트워크를 통한 정보 유통이 중요해 짐으로 네트워크의 침입으로 인해 발생되는 손실도 매우 크다. 따라서 인터넷 침입을 탐지하고 차단하는 것은 중요하다. 또한 요즘은 침입의 탐지에 그치지 않고 침입을 예방하려는 많은 연구가 진행되고 있다.

인터넷 침해에 대응하기 위한 방법으로 침입차단 시스템, 침입예방 시스템, 침입탐지 시스템을 들 수 있다. 침입차단 시스템은 침입이 발생하지 않도록 네트워크의 출입구를 제어하는 기능을 수행한다. 인증을 받지 않은 외부의 접근 시도는 차단할 수 있으나, 이미 인증된 사용자나 이를 가장한 침입에 의한 공격에는 취약하다. 특히 내부인 혹은 허가된 외부인에 의해 발생하는 침입은 대처하기 힘든 단점이 있다. 침입탐지 시스템은 정보시스템 또는 네트워크로부터 보안 관련 정보들을 수집, 분석하여 침입 또는 오용을 탐지할 뿐만 아니라 침입에 대한 적절한 기능을 포함하는 시스템이다[1]. 침입예방 시스템은 Packet header + content field 를 검사하여 침입 혹은 유해정보 여부를 자체적으로 판단하는 시스템이다[2].

본 논문에서는 침입탐지 시스템을 위한 침입탐지 기법으로 기존의 침입탐지 방법과는 다른 n 차원 벡터 기법을 이용하여 특정 요소에 대한 침입탐지 기법을 제안한다. IP 헤더와 TCP 헤더에서의 일부 파라미터만을 벡터 요소로 선택하였다.

본 논문의 구성은 2 장에서 IDS 의 정의, 기존의 방법, 관련연구와 침입탐지를 위한 가정 기술하고 3 장에서 본 논문에서 제안하는 탐지 방법, 4 장에서는 실험 및 결과 5 장에서는 결론으로 맺을 것이다.

2. 관련연구 및 가정

일반 IDS 란 Intrusion Detection System 의 약자로 탐지 대상 시스템이나 네트워크를 감시하여 비인가 되거나

비정상적인 행동을 탐지하여 구별해 낸다. 여기서 침입이란 자원의 기밀성, 무결성, 가용성을 훼손하는 제반행위로써 기밀성은 자원으로의 접근은 합법적인 권한을 가진 사람만이 가능함을 무결성은 자원이 훼손되거나 변경되지 않도록 함을 가용성은 합법적 권한을 가진 사용자는 언제나 자원의 접근이 가능함을 나타낸다[3]. 정상적인 트래픽으로부터 이후 트래픽을 예측하는 방법 Seasonal ARIMA 모형[7], 웨이블릿 분석을 사용[8], 퍼지-자기회기 모형[9], 지수평활법[5]이 있다. 트래픽 분석으로 flow 기반의 인터넷응용 트래픽 특성분석 연구도 있었다. 이는 flow Grouping mothod 를 이용하여 응용 프로그램 별로 분류하고, flow 기반으로 이들 응용 프로그램을 다양한 각도에서 분석하였다. [6]

본 논문에서 정상적인 트래픽으로부터 정상적인 트래픽 데이터를 가공하여 측정하고자 하는 시점에서의 침입을 탐지하려 한다. 본 논문에서 침입탐지를 효과적으로 하기 위한 몇 가지 가정이 필요하다. 트래픽의 양은 시간대 또는 요일과 같은 변수에 민감하기 때문에 본 논문에서 제시한 벡터의 각 요소의 평균값과 분산값은 측정하고자 하는 시점에 가장 적절한 값이라 가정한다. 트래픽 양과 같이 시간에 따라 변동하는 데이터의 이상을 탐지하는 기법으로 시계열 분석이 있다. 이 분석은 분산분석을 이용하여 월별요인, 요일요인, 시간요인을 파악하였다.[4] 또한 침입탐지의 목적은 지역 네트워크망의 보호를 목적으로 하며, 지역 네트워크망으로 들어오는 패킷에 대한 것이다. 침입탐지 모니터링을 위한 위치는 지역 네워워크 망의 방화벽을 통과한 패킷에 대한 침입탐지 모니터링을 한다. 또한 인트라넷의 중요서버를 보호 대상으로 하여 패킷의 목적지 주소가 중요서버인 것들에 대한 침입탐지 기법을 적용한 것이다. 아래의 그림은 침입탐지 모니터링을 위한 위치를 나타낸다.

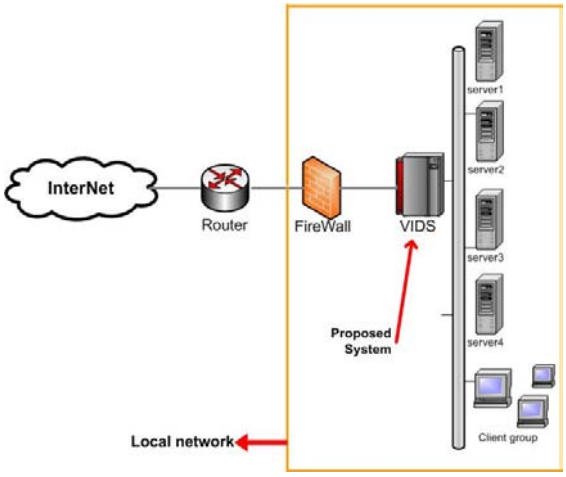

그림 1. 제안하는 시스템의 위치

3. 제안하는 침입탐지 기법

3.1 탐지를 위한 벡터요소

네트워크 상의 침입을 탐지하기 위한 정보들은 많이 있다. 그 중에서 많이 사용하고 있는 정보는 네트워크를 통과하는 IP 패킷에 대한 정보이다. 이러한 IP 패킷에 대한 정보는 IP 계층에서 쉽게 얻을 수 있다. IP 계층에서 얻을 수 있는 정보는 프로토콜의 종류 (UDP|TCP|ICMP), source IP address, destination IP address, source port number, destination port number, 트래픽양 등이 있다. 또한 시스코의 Netflow 나 SNMP(Simple Network Management Protocol)의 MIB 정보를 활용할 수 있다.

그러나 공격의 형태는 주로 특정 프로토콜과 포트를 통하여 이루어 지고 공격이 발생하였을 때, 공격과 관련된 트래픽 양이 급격히 증가한다. 따라서 실시간 침입탐지를 위해 IP layer 에서 쉽게 얻을 수 있는 특정 요소들을 벡터 파라미터로 선정하였다. 첫째로 단위시간 동안의 패킷의 양이다. 공격이 행해질 때 전체 트래픽이 증가하며, 트래픽의 증가는 네트워크의 성능저하를 초래하기도 한다. 따라서 전체 패킷량의 변화는 침입탐지의 중요한 요소이다. 둘째로, 일정시간당 각 프로토콜의 패킷량이다. 많은 공격들이 특정 프로토콜을 이용한다. 예로, Smurf attack 은 ICMP 프로토콜을 이용하고, Fraggle attack 은 UDP 프로토콜을 이용하여 공격한다. 이외에도 알려진 많은 공격들 역시 특정 프로토콜을 이용하여 공격한다. 따라서 프로토콜도 침입탐지를 위한 중요한 파라미터이다. 일반적으로 일정시간당 패킷의 수를 살펴보면 TCP 패킷이 90% 이상을 차지하고 UDP, ICMP 순서이다. 특히 ICMP 패킷의 수는 거의 없는 것으로 나타났다.[18] 셋째는 일정시간당 포트별 패킷량이다. 웜의 대부분은 특정 포트를 이용해 공격을 한다. 예로 Nimda 는 69 번 포트를 CodeRed 는 80 번 포트, SQL Slammer 는 1434 포트, Witty 는 source 4000 포트, Sasser 은 445, 5554, 9996 을 이용한다. 따라서 본 논문에서 중요한 벡터 성분은 (평균패킷량, 프로토콜별 평균패킷량, i_포트 평균패킷량, j 포트 평균패킷량, …)이다. 또한 추가되는 포트넘버의 수에 따라서 차수가 늘어난다.

3.2 필요한 벡터요소 구하기

■ 요소의 정규화

아래의 식과 같이 3 가지 요소의 값에 대한 정규화가 필요하다.

$$m_t = \frac{\sum(Paket/unit\_time)}{Number\_of\_unit\_time} \quad (1)$$

$$m_p = \frac{\sum((TCP|UDP|ICMP)Pakets/unit\_time)}{Number\_of\_unit\_time} \quad (2)$$

$$m_{n(i)} = \frac{\sum((i\_Port)Pakets/unit\_time)}{Number\_of\_unit\_time} \quad (3)$$

$$\sigma_t^2 = \frac{\sum(X_i - m_t)^2}{Number\_of\_unit\_time} \quad (4)$$

. 각 용어의 의미는 다음과 같다

• $Number\_of\_unit\_time$:단위시간당 패킷량
• $Paket/unit\_time$: 단위시간당 총 패킷량
• $(TCP|UDP|ICMP)Pakets/unit\_time$:단위시간당 각 프로토콜의 패킷량
• $(i\_Port)Pakets/unit\_time$: 단위시간당 포트별 패킷량
• $X_i - m_t$: 편차

σ_p^2 값과 $\sigma_{n(i)}^2$ 값은 식 4)에 의해 구할 수 있다.

위의 식에서 구한 평균 패킷량과 프로토콜별 평균 패킷량, 포트별 평균 패킷량은 데이터의 양이 많이 질수록 모두 정규분포를 따르게 되는데 이러한 정규분포는 표준화에 의해 평균이 0 표준편차가 1 인 표준정규분포로 된다. 이 표준정규분포의 확률을 이용하여 벡터요소의 값들을 0 에서 1 사이의 값들로 변환할 수 있다

일반적으로 일정시간당 패킷의 수를 살펴보면 TCP 패킷이 90% 이상을 차지하고 그 다음이 UDP, ICMP 이다. 특히 ICMP 패킷의 수는 거의 없는 것으로 나타났다.(참조) 그러므로 UDP, ICMP 패킷은 백터요소를 (평균패킷량, 프로토콜별 평균패킷량) 인 2 차 벡터를 생성하고, TCP 를 이용한 공격은 포트번호가 중요한 특성 파라미터 값이므로 (평균패킷량, 프로토콜별 평균패킷량, i_포트 평균패킷량, j 포트 평균패킷량, …) 와 같이 추가되는 포트에 따라 일반적으로 n 차원 벡터 요소까지 확장 할 수 있다.

■ 표준화 과정

아래의 과정에 따라 요소를 표준화 할 수 있다.

단계 1) 단위시간당 분석하기 위한 요소를 선택

$$\vec{R} = (T, P, N(i), N(j), \ldots) \qquad (5)$$

단계 2) 각 요소의 표준화

$$z_t = \frac{T - m_t}{\sigma_t} \qquad (6)$$

모든 요소에 식 6)을 적용하면 $z_{(t|p|n(i)|n(j)|\ldots)}$ 의 값을 구할 수 있다. 벡터의 모든 요소들이 0 에서 1 사이의 값으로 표준화된다.

단계 3) 값들을 확률값으로 변환

$$P(Z \le z_{(t|p|n(i)|n(j)|\ldots)}) \qquad (7)$$

$$\vec{r} = (p_t, p_p, p_{n(i)}, p_{n(j)} \ldots) \qquad (8)$$

$P(Z \le z_{(t|p|n(i)|n(j)|\ldots)})$ 의 값은 그림 2 에서 어두운 부분으을 나타낸다.

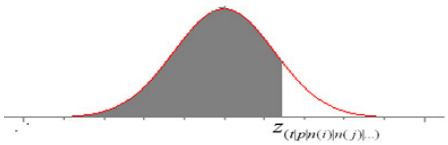

그림 2. 표준정규분포 곡선

3.3 제안하는 벡터를 이용한 침입탐지 기법

표준화된 벡터 요소들로 벡터를 생성하면 생성된 값은 벡터 요소들에 의해 n 차원 벡터공간에 표현된다. 이처럼 벡터공간에 표현된 값들은 각 축에 정사영을 하면 그 벡터요소의 값만 남고 나머지 요소들은 모두 0 의 값이 된다. 예를 들면, 총 패킷량축에 정사영을 하면 $\vec{r_t} = (p_t, 0, 0, 0 \ldots)$ 가 된다.

이처럼 각 요소에 대해 정사영 벡터를 구하고 r 과 각 축에 대한 정사영 벡터가 이루는 각을 $\theta_{(t|p|n(i)|n(j)|..)}$ 라 하면

$$\cos\theta_{(t|p|n(i)|n(j)|..)} = \frac{\| \vec{r}_{(t|p|n(i)|n(j)|...)} \|}{\| \vec{r} \|} \qquad (9)$$

$\| \vec{r} \|$ 와 $\| \vec{r}_{(t|p|n(i)|n(j)|...)} \|$ 의 값은 벡터의 크기 $(\| \vec{r} \| = \sqrt{p_t^2 + p_p^2 + p_{n(i)}^2 + p_{n(j)}^2 + \cdots})$

$\theta_{(t|p|n(i)|n(j)|..)}$: 벡터와 각축이 이루는 각도.

$$\| \vec{r'} \| = \sqrt{\frac{2(p_t^2 + p_p^2 + p_{n(i)}^2 + p_{n(j)}^2 + \cdots)}{n}} \qquad (10)$$

식 9)에 의해 전체 요소에 대한 특정 요소의 변화량을 관찰할 수 있다. 식 10)은 n 차원의 경우 아래의 그림 3 을 적용할 수 있도록 벡터크기의 변환한 값을 나타낸다.

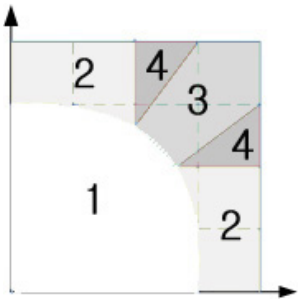

그림 3. 경고 영역의 2차원 구분

$\| \vec{r} \|$ 와 $\cos\theta_{(t|p|n(i)|n(j)|..)}$ 값으로 트랙픽 이상유무를 결정한다. 이상유무의 경고는 아래와 같이 경고레벨을 결정한다.

• Level 1 $(\| \vec{r} \| \le 0.7)$: 1 의 영역을 나타내며 네트워크의 성능에 영향을 미치지 않는다.
• Level 2 $(\| \vec{r} \| > 0.7$ and, p_t or $p_p \le 0.5)$: 2 의 영역을 나타내며 거의 발생하지 않는 영역이다.
• Level 3 $(\| \vec{r} \| > 0.7$, $0.6 \le \cos\theta_{(t|p|n(i)|n(j)|..)} \le 0.8)$: 3 의 영역을 나타내며 총 패킷량과 특정프로토콜 패킷량이 모두 증가하였으므로 정상적인 증가로 판단하나, 트래픽양이 많아졌으므로 트래픽 조절이 필요하다.
• Level 4 ($\| \vec{r} \| > 0.7$ and p_t and p_p >0.5 and $\cos\theta_{(t|p|n(i)|n(j)|..)} < 0.6$ or $\cos\theta_{(t|p|n(i)|n(j)|..)} > 0.8$) : 비정상적

인 트래픽 증가로 총패킷양보다 특정패킷량의 증가가 두드러짐으로 공격으로 간주하고 최고의 경고를 한다.

4. 예제 시나리오와 분석

4.1. 2-차원 패킷분석

UDP 패킷의 비정상적 증가를 가정한다.

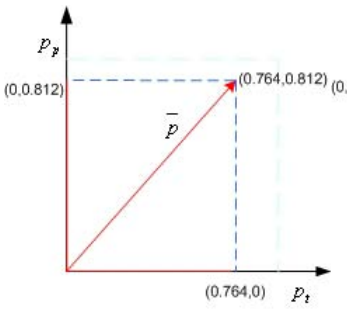

그림. 4 정상적인 벡터

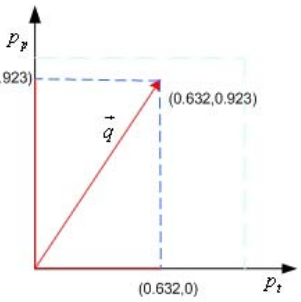

그림. 5. UDP를 이용한 공격

위의 그림 4 에서 $\vec{p} = (0.764, 0.812)$, $\overrightarrow{p_p} = (0, 0.812)$, $\|\vec{p}\| \approx 1.115$, $COS\theta_P \approx 0.684$ 이다. $\|\vec{p}\| > 0.7$ 이므로 패킷의 양이 증가하였으나, $0.6 \le COS\theta_P \le 0.8$ 이므로 전체 패킷량의 증가에 따라 UDP 패킷량도 증가하였으므로 정상적인 패킷의 증가량으로 간주고 Level 3 의 경고를 한다.

그림 5 의 경우는 UDP 프로토콜을 이용하여 공격할 때의 벡터이다. $\vec{q} = (0.632, 0.923)$, $\overrightarrow{q_P} = (0, 0.923)$, $\|\vec{q}\| \approx 1.118$, $COS\theta_P \approx 0.826$. $\|\vec{q}\| > 0.7$ 로 패킷량의 증가를 알수 있고, $COS\theta_P > 0.8$ 로 우리는 비정상적 패킷의 증가하였음을 알수 있다. 따라서 비정상적인 UDP 패킷량의 증가로 파악하고 Level 4 의 경고를 한다.

4.2. 3-차원 패킷분석

80 포트 패킷의 비정상적 증가를 가정한다.

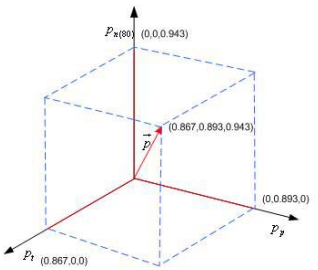

그림. 6. 3차원 정상벡터

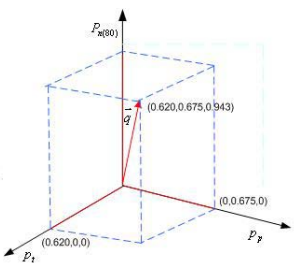

그림.**7.** 80 port 공격

그림 6 에서, $\vec{p} = (0.867, 0.893, 0.943)$, $\vec{p}_{n(80)} = (0, 0, 0.943)$, $\|\vec{p}'\| \approx 1.275$, $COS\theta_{n(80)} \approx 0.740$ 이다. $\|\vec{p}'\| > 0.7$ 에 의해 패킷량이 증가함을 파악할 수 있고, $0.6 \le COS\theta_{n(80)} \le 0.8$ 에 의해 정상적인 패킷의 증가 하였음을 알 수 있다. 따라서, 정상적인 패킷의 증가로 간주하고 Level 3 의 경고를 한다.

그림 7 에서 $\vec{q} = (0.620, 0.675, 0.943)$, $\vec{q}_{n(80)} = (0, 0, 0.943)$, $\|\vec{q}'\| \approx 1.153$, $COS\theta_{n(80)} \approx 0.817$ 이다. $\|\vec{q}'\| > 0.7$ 에 의해 트래픽의 증가를 확인하고, Level 4 의 경고를 한다.

5. 결론

본 논문에서 제안하는 n 차원 벡터 기법은 전체 트래픽양의 변화와 특정 프로토콜의 패킷량의 변화 또 특정 포트번호의 패킷량의 변화를 벡터 요소로 침입을 탐지하였다. 이 방법의 이점은 연관성있는 요소들을 벡터요소에 포함함으로써 그 요소들에 관한 관계를 파악할 수 있고 각 요소들의 축에 정사형을 함으로 각 요소에 대한 이상유무도 확인 할 수 있었다.

따라서 본 논문에서 제안하는 n 차원 벡터기법은 트래픽의 양을 요소로 하는 침입탐지 뿐만 아니라 다른 분야에도 적용할 수 있다.

참고문헌

[1] 1. Dorothy E. Dennig, “An intrusion-detection model”, IEEE Transactions on Software Engineering, Feb.1980

[2] Kyujin Cho “Intrusion Prevention System(IPS) : The role and definition that see though real and the future of security”. http://www.secuinfo.com/ ips/ips_notice.htm, 2004

[3] Charles P.Pfleeger, Shari Lawrence Pfleeger: Security in Computing, 3nd ed. PRENTICE HALL. pp.9-19, 2003

[4] J. L. Hellerstein, F. Zhang, P. Shahabuddin, “A statistical approach to predictive detection”, Computer Networks, 2001

[5] “Fast Detection Scheme for Broadband Network Using Traffic Analysis”, 2003

[6] Mungsub Kim, Youngjun Won, Hungjo Lee, Wonki Hong, “Flow-basde Internet Application Traffic Characteristic Analysis”, KNOM Review, Vol. 7, No.1, 2004

[7] Y. Shu, M. Yu, J. Liu: Yang and O.W.W, “Wireless traffic modeling and prediction using seasonal ARIMA models”, Communications, 2003. ICC ’03 IEEE International Conference on, v.3, May. 2003

[8] P. Barford, J. Kline, D. Plonka and A.Ron, “A Signal Analysis of Network Traffic Anomalies”, IMW’02, Nov. 2000

[9] B. Chen, S. Peng and K Wang, “Traffic Modeling, Prediction, and Congestion Control for High-Speed Networks: A Fuzzy AR Approach”, IEEE Transactions on Fuzzy System, Oct 2000