SOM 기반의 효율적인 침입 분류 체계에 관한 연구

최진우*, 우종우*
*국민대학교 컴퓨터학부
e-mail : cwwoo@kookmin.ac.kr

A Study on an Effective Intrusion Classification Mechanism based on SOM

Jin-woo Choi *, Chong-woo Woo *
*School of Computer Science, Kookmin University

요 약

최근 침입의 형태는 기존 공격자의 직접적인 시스템 침입 및 악의적 행위들의 행사와는 달리 침입 자동화 도구들을 사용하는 형태로 변모해 가고 있다. 알려지지 않은 공격의 유형 또한 변형된 이들 도구들의 사용이 대부분이다. 이들 공격도구들 대부분은 기존 형태에서 크게 벗어나지 않으며, 침입 도구의 산출물 또한 공통적인 형태로 존재한다. 본 논문에서는 알려지지 않은 다양한 공격 유형 또한 기존 유사한 공격군으로 분류하기 위한 침입 분석 알고리즘으로 SOM(self-Organizing Maps)을 적용하고, 침입 구체화 분석 단계에서 공격도구들의 패턴을 정형화한 지식베이스를 기반으로 분석하는 시스템을 제안한다.

1. 서론

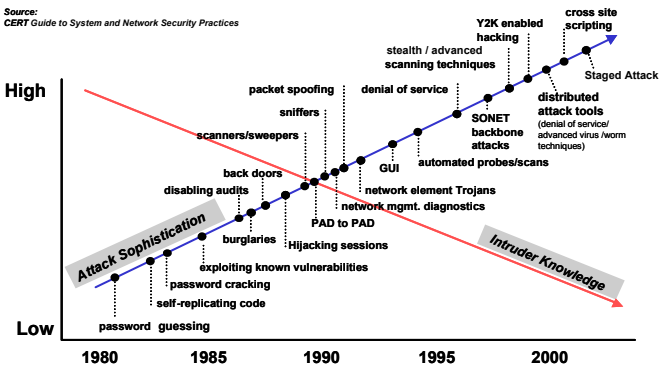
최근 ITU(International Telecommunication Union)의 조사에 따르면 우리나라의 가정 사용자의 점유율 58%로써 광대역 사용국의 선두국으로 보고되었다. 이는 다른 나라로부터의 공격 시 상당한 매력을 느끼게 하는 요인으로서 대부분의 공격 시작점으로의 이용을 제공하게 된다는 점이다. 이러한 광대역 기반 사용자들의 빠른 증가는 다른 나라들에 의한 광대역 사용을 용이하게 함으로써, 우리나라는 10,000 명당 가장 많은 공격 수치를 포함하고 있는 공격 랭킹 1위라는 오명을 남기고 있다[1]. 이러한 수치는 순수 우리나라에서 발생된 공격 개시 뿐만이 아니라, 침입 경유지로 이용된 사실 모두를 포함하고 있다. 즉 우리나라의 기관 및 개인 호스트들이 어떠한 형태로든 공격의 중간 경유지로서 활용되고 있음을 의미한다[2].

위의 사실과 같이 많은 공격 회수의 증가는 인터넷에서 쉽게 획득할 수 있는 공개된 공격도구의 사용 및 변조된 공격도구들로부터 이루어진 것들이 대부분이다. 본 논문에서는 SOM을 이용한 몇몇 구분된 공격군으로부터, 사용된 공격도구들의 패턴을 지식베이스로 구축하고, 이를 기반으로 구체화 분석을 수행하는 시스템을 제안한다.

2. 관련 연구

침입의 형태를 분석하고 이에 대한 적절한 대응을 위해서는 현재의 침입 경향 파악이 선행되어야 한다[3].

2.1 침입 행위의 경향



[그림 1] 최근 침입 행위의 경향

[그림 1]과 같이 1980년대 초반 공격자들의 개별적인 전문 지식을 사용한 시스템 공격들이 이루어졌다. 대부분의 공격이 공격자들의 수작업에 의한 직접적인

명령어들에 의해 사용됨으로써 전문 지식이 요구되는 반면, 그 정교함은 낮은 수준이었다. 그러나 이후 자동화된 도구의 사용, 그리고 시스템 취약점 발견을 위한 빠른 스캐닝 도구들이 사용됨으로써 공격은 손쉽게 감행되어져 왔다. 그럼으로써 이전과는 상이하게 공격을 위한 전문 지식은 낮아진 반면, 공격의 정교함은 매우 높은 수준이 되어왔다[4].

2.2 자가 조직화 신경망(SOM)

초기의 경쟁 학습 알고리즘은 단순 경쟁 학습 알고리즘으로서 항상 승자 뉴런만을 학습 시키므로 초기 가중치 벡터들의 분포에 따라 전혀 학습이 이루어지지 않는 출력 뉴런들이 생기는 문제점이 발생한다. 이를 해결하기 위한 다양한 접근들이 있었으며, 이 중 가장 대표적인 경쟁 학습이 Kohonen의 자가 조직화 신경망(Self-Organizing Map)이다[5]. 자가 조직화 알고리즘은 승자 뉴런 뿐만 아니라 위상적으로 이웃한 뉴런들도 함께 학습시킨다. 그 결과 비슷한 입력 패턴들은 인접한 출력 뉴런들의 기하학적인 관계로써 형성된다. SOM의 학습과정은 다음 단계들을 수행한다.

- 1 단계: 연결 가중치를 초기화 한다. N 개의 입력과 M 개의 출력을 연결하는 가중치들은 아주 작은 임의의 값으로 설정한다.
- 2 단계: 새로운 입력 패턴을 입력 뉴런에 제시한다.
- 3 단계: 입력 벡터와 모든 출력 뉴런들과의 거리를 계산한다.

$$d_j = \sum_{i=0}^{N-1} \left[X_i(t) - W_{ij}(t) \right] \quad \text{[식 2-1]}$$

  $X_i(t)$는 시간 t에서 뉴런 i로의 입력, $W_{ij}(t)$는 시간 t에서 입력 뉴런 i로부터의 출력 뉴런 j로의 가중치
- 4 단계: 최소 거리를 가지는 승자 뉴런 c를 구한다.

$$c = \min_j d_j \quad \text{[식 2-2]}$$

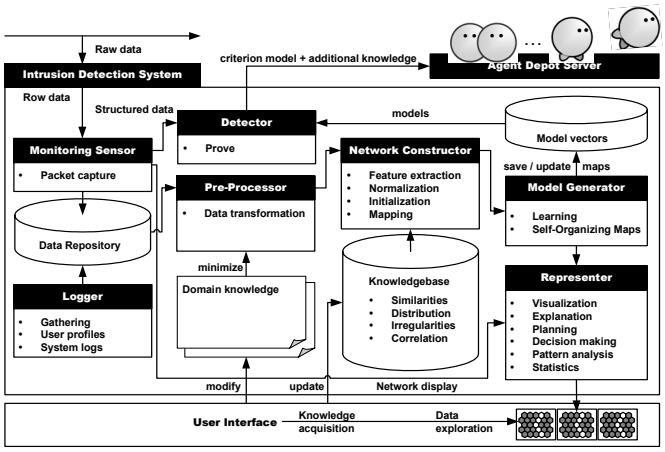
- 5 단계: 승자 뉴런 c와 이웃 출력 뉴런의 연결 가중치들을 갱신한다.

  $W_{ij}(t+1) = W_{ij}(t) + \alpha(t)\left[X_i(t) - W_{ij}(t) \right] \; for \; j \in NE_c(t)$ [식 2-3]

  $\alpha(t)$는 0과 1 사이의 값을 가지는 학습률로서 시간에 따라 감소하는 함수이며, $NE_c(t)$는 승자 뉴런 c로부터 일정한 범위 내에 있는 출력 뉴런을 포함한다.
- 6 단계: 2 단계의 새로운 입력 벡터를 처리한다.
- 7 단계: 지정된 학습 회수까지 2 단계부터 6 단계의 과정을 반복 수행한다.

3. 시스템 설계

호스트 상의 침입 탐지 시스템은 모니터링 센서, 디텍터, 전처리기, 네트워크 생성기, 모델 생성기, 리프리젠터, 6 개의 핵심 모듈로 구성된다[그림 2].



[그림 2] 침입 탐지 시스템 구조

- 네트워크 모니터링 센서는 *tcpdump*를 사용하여 의미있는 형태의 산출물을 획득한 후, 이를 구조화된 형태로 변환한다. 변환된 패킷 데이터들은 추후 오프-라인 학습을 위한 입력으로 사용되기 위하여 데이터 저장소에 저장된다.
- 전처리기는 데이터 저장소로부터의 데이터 선택 또는 정제를 위한 바로 이전 단계에 해당한다. 데이터 저장소의 데이터들은 그 구조가 이질적인 형태들로 구성되어 있다. 그러므로 이러한 이질적인 구조의 데이터들을 신경망의 입력으로 사용하기 위하여 알맞은 형태로 변환하거나 재배열하는 과정이 필요하다. 본 설계에서는 영역지식 참조의 최소화를 목표로 하고 있지만, 몇몇 단계에서는 전문가의 조정이 필요하며, 현재 단계에서는 학습에 필요한 데이터들의 양, 데이터들의 길이, 또는 연대기적 정렬에 관련된 단순한 지식만을 참조한다.
- 네트워크 생성기는 신경망의 속성을 결정하는 기능을 담당한다.
- 모델 생성기는 침입 탐지 시스템의 핵심 모듈로써, 생성된 모델은 신경망의 학습 후 창출되는 산출물이 된다. 본 설계에서는 비교사 학습인 SOM(Self-Organizing Maps)을 사용한 모델 벡터들의 생성을 제안한다. 뿐만 아니라, 신경망의 형성 시 지식베이스를 참조하여 신경망 속성의 결정이 가능하도록 함으로써, 다양한 학습 엔진의 탑재가 가능하도록 그 적응력을 고려한다.
- 리프리젠터는 학습되어진 모델 벡터의 가시화를 담당한다. 가시화 작업은 SOM이 제공하는 대표적인 특징들 중의 하나로써, SOM에 의해 형성된 맵은 인간으로 하여금 이해 가능한 형태로 표현된다. 이렇게 함으로써 보안 전문가들과의 협력을 통해 새롭게 해석되어, 보다 효과적인 새로운 모델의 생성이 가능하다.
- 디텍터는 모니터링 센서들로부터의 구조화된 데이터를 수신한 후, 이를 판별하기 위하여 모델 생성기에 의해 생성된 모델 벡터들과의 유사성 척도를 기준으로 평가하게 된다.

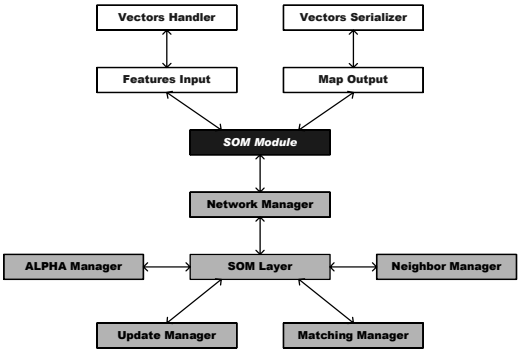
본 시스템에서 침입탐지 알고리즘으로 제안하는 SOM의 적용 시의 장점으로는 다음과 같다.

첫째, 침입 탐지 시스템의 기반 프레임워크를 결정할 때 학습 능력은 매우 중요한 것이다. 신경망 학습은 대표적으로 교사 학습, 그리고 비교사 학습으로 구분된다. 교사 학습은 알려져 있는 문제에 대하여 학습 데이터들의 레이블링 작업이 수반되어야 하는 반면, 비교사 학습 시에는 레이블이 필요로 하지 않으므로 알려져 있지 않은 문제에 대해서도 특징들 간의 관계만을 가지고 학습이 가능하다는 장점을 제공한다.

둘째, 비교사 학습의 SOM은 패턴 분석 및 클러스터링, 양쪽 모두에게 적응성을 제공한다. 또한 U-matrix를 이용한 클러스터 가시화가 가능함으로 비록 알려져 있지 않은 문제에서도 직관적인 이해가 가능하다는 장점을 제공한다.

3.1 침입탐지를 위한 자가 조직화 신경망(SOM)

SOM 프레임워크의 구조는 모듈 기반으로 설계한다. 구성 모듈은 [그림 3]과 같이 6 개의 모듈로 구성된다. SOM 모듈의 상위는 입력과 출력을 의미한다.



[그림 3] SOM 프레임워크

- SOM 모듈(SOM Module): SOM의 핵심 엔진.
- 네트워크 매니저(Network Manager): 맵을 구성하는 뉴런들을 제어한다. 학습된 맵의 클러스터링 가시화를 수행하는 U-matrix visualizer 모듈을 내부적으로 포함한다.
- SOM 레이어(SOM Layer): 네트워크 매니저로부터 유입되는 입력을 수신하고, 이를 학습한다.
- ALPHA 매니저(ALPHA Manager): 현재 학습률(α)을 조절한다.
- 이웃 매니저(Neighbor Manager): 이웃함수를 가지고 적절한 반경 내의 이웃 뉴런들을 선택한다.
- 매칭 매니저(Matching manager): 입력과 맵 상의 뉴런들의 BMU를 선택한다.
- 업데이트 매니저(Update Manage): 현재 뉴런들의 가중치 업데이트를 담당한다.

(가) 정규화

본 논문에서 학습을 위해 사용하는 KddCup'99[6] 데이터 집합은 수치형 데이터와 기호형 데이터들의 혼합으로 구성되어 있다.

KddCup'99 데이터 집합 내의 기호형 데이터들은 개별적 TCP 연결의 기본 특징들에 관한 부분으로 "*protocol_type*", "*service*", 그리고 "*flag*" 특징들에서 표현되어지고 있다. 또한 수치형 데이터로써 연속적인 값을 표현하는 "src_bytes", "dst_bytes"는 수치 분포의 범위가 매우 크다. 만일 그러한 각각의 벡터를 별도의 변환과정 없이 SOM의 학습에 사용한다면, 높은 분산을 가지는 차원이 맵의 조직화에 다수를 점하는 경향을 나타내게 될 것이다. 이러한 문제를 해결하기 위하여 SOM의 학습을 위해 결정되어진 특징 벡터들에 대하여 유클리드 거리의 개념을 내포하는 l_2 정규화를 수행한다. 총 길이 n의 입력 데이터 X에 대하여 d 차원의 연속형 실수값인 경우 다음을 만족한다.

$$X = \{X_i\}_{i=1}^{n}, \quad X_i^R \in R^d \qquad [식\ 3\text{-}1]$$

위를 만족하는 경우 l_2 정규화는 다음과 같다.

$$the\ length\ of\ X_i^R \ = \ l_i = \sqrt{\sum_{j=1}^{d} X_{ij}^2} \qquad [식\ 3\text{-}1]$$

$$n_i = \frac{x_i}{l_i} \qquad [식\ 3\text{-}1]$$

$$\langle x_1, x_2, \cdots, x_d \rangle \rightarrow \langle n_1, n_2, \cdots, n_d \rangle \quad n_i : normalized\ vector;\ range[0,1]$$

연속형 실수값 뿐만 아니라, 기호형 데이터들도 침입탐지를 위한 중요한 정보를 내포하고 있을 수 있으며, 또는 시험에 의해 클래스를 구성하는 새로운 특징들 간의 관계를 획득할 수도 있다.

본 논문에서는 두 가지 타입의 데이터를 함께 학습에 참여시킴으로써 보다 효과적인 클러스터링의 성능을 기대할 수 있도록 설계한다. 기호형 데이터의 값들은 임의 이산 수치값을 부여하고 위와 유사한 과정을 수행한 후, SOM 학습을 위한 입력으로 사용한다.

(나) SOM 학습

Algorithm SOM
Input: Set of N dimension vector, X
Output: Subsets of input data. (M subsets)

begin
Randomly initialize $W_i = (w_{i1}, w_{i2}, w_{i3}, ..., w_{in})$ for each node
for ($t = 0$; unless a topping condition is reached; Increase t)
for (for all input data)
for ($i = 0$ to M)
Compute $D_i = \|X_t - W_i^{(t)}\|$
endfor
Find the winner $j = i$ such that $D_i(t)$ is minimum for overall i
Update the winner j (and its neighbors)
endfor
endfor
end

[그림 4] SOM의 학습 알고리즘

① 연결 가중치 벡터들의 초기값을 임의로 할당한다.
② 임의 연결 가중치를 할당 후, 입력 벡터와 유사성을 측정한다. 본 논문에서는 일반적인 유클리드 거리를 사용한다.
③ 승자 뉴런을 발견하면 연결 가중치 갱신을 위해 *normalized vector sum*을 사용한다. 갱신 시 가우시안 함수를 사용한다.
④ 이를 모든 입력 벡터에 대하여 순환 적용하기 위해 ②-③을 반복한다.

침입 탐지 결과는 크게 4 가지로 구분된다. 이러한 구분은 KddCup'99의 4 개의 공격군(DoS, R2L, U2R, Probing)으로 구성되기 때문이다. 이들 개별적인 공격군들에 대한 학습된 모델을 기준 모델(criterion model)로 결정하고, 이들에 해당하는 에이전트들이 각각의 공격군들을 대표한다. 이러한 공격군을 대표하는 에이전트들이 최초 피침입 시스템으로 파견된다. 본 설계에서는 공격군에 대하여 대표하는 에이전트들에 의해 침입 구체화 분석 단계를 수행한다. 이렇게 함으로써 학습된 모델에 의한 의사양성 및 의사음성의 문제점들을 감소시킬 수 있다.

3.2 침입 분석 단계

침입 분석 단계는 침입 구체화 분석 단계와 침입 경유지/발원지 분석 단계로 구분된다.

가) 침입 구체화 분석 단계

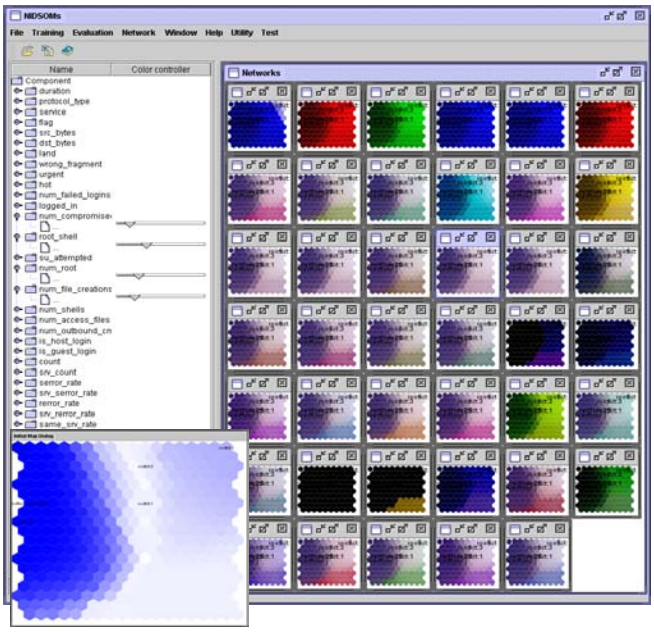
구체화 분석 작업이 필요한 이유는 대부분의 침입은 단일 공격에 의해서 일어나지 않기 때문이다. 즉 U2R 만의 공격에 의해 침입이 이루어지는 것이 아니라, R2L 공격과 병행하여 시도된다. 예를 들어, R2L의 "guess_passwd", "dictionary" 공격들이 선행되어 사용자의 패스워드를 알아내어 적법한 로그인한 뒤, 루트킷/백도어를 다운로드하고, 이를 이용하여 루트의 권한을 획득하기 위한 U2R 공격이 이루어지는 것이다. 그러므로 이들을 구체적으로 분석할 수 있는 작업이 침입 구체화 분석 단계이다.

나) 침입 경유지/발원지 분석 단계

침입 구체화 분석 단계가 완수되면 그 결과는 공격군에 포함된 구체적인 침입 유형이 될 것이며, 이 정보를 이용하여 이전 침입 경유 시스템의 위치 정보를 발견하고, 침입 발원 시스템에 도달할 때까지 순환한다. 예를 들어 U2R 공격군에서 대부분의 루트킷이 사용하는 특수 목적의 파일로부터 공격자의 접속 정보를 획득할 수 있다. 루트킷은 이를 위한 용도의 몇몇 파일들을 디폴트로 생성한다. 그러므로 이들의 발견을 위한 관련 정보를 지식화 하여 작업 에이전트들의 개별적인 고유 지식으로 활용하며, 이들을 분석할 수 있는 절차 지식들을 설계한다.

4. 시스템 구현

본 논문에서는 신뢰성 있는 실험을 위하여 훈련 데이터 집합으로 KddCup'99 데이터를 사용하였으며, 침입 탐지 분석 알고리즘으로는 비교사 학습인 SOM을 이용하여 개발하였다. [그림 5]와 같이 SOM의 학습 결과와 클러스터링 된 결과로부터 "color controller" 인터페이스를 사용하여 각 특징들의 관계에 관한 새로운 지식을 도출해 볼 수 있다.



[그림 5] 자가 조직화 신경망의 학습

5. 결론

본 시스템의 장점은 다음과 같다. 첫째, SOM에 의해 학습된 맵을 최종 가시화 형태로 생성함으로써, 전문가들에 의해 새롭게 획득한 지식으로 지식베이스 갱신이 가능하다. 둘째, 분석 결과를 기반으로 학습에 영향을 미치지 않는 사소하고, 불필요한 특징들을 배제할 수 있도록 함으로써, 신경망의 재학습 시 최적화를 도모할 수 있다.

참고문헌

[1] Symantec Internet Security Threat Report, Volume 3, Feb. 2003. available at http://www.symantec.com/region/hu/huresc/download/2003_02_03SymantecInternetSecurityThreatReport.pdf.
[2] Korea Computer Emergency Response Team Coordination Center. available at http://www.certcc.or.kr/statistics/hack/hack.htm.
[3] CERT Coordination Center, "Overview of Attack Trends," Carnegie Mellon University, April 8, 2002. available at http://www.cert.org/archive/pdf/attacktrends.pdf.
[4] Julia H. Allen, "CERT Guide to System and Network Security Practices," Addison-Wesley, 2001.
[5] T. Kohonen, Self-Organizing Maps, Springer-Verlag, Berlin, 1995.
[6] KddCup'99. available at http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html.