효율적인 침입탐지를 위한 네트워크 정보와 시스템 콜 정보융합 방법개발

문규원† 김은주° 류정우 김명원
(주)인우기술†, 숭실대학교 컴퓨터학과
blue7786@ssu.ac.kr°, mkim@comp.soongsil.ac.kr

Data Fusion of Network and System Call Data For Efficient Intrusion Detection

Kyuwon Moon† Eun Ju kim° Joung Woo Ryu Myungwon Kim
Inwoo Tech Inc.†, Dept. of Computing, Soongsil University

요 약

최근 인터넷, 인트라넷과 같은 통신 기술 발전에 따라 거의 모든 시스템이 서로 연결되었고, 사용자들은 손쉽게 정보를 공유할 수 있게 되었다. 따라서 시스템 침입을 통한 데이터의 변형과 인증 받지 않은 접근과 같은 컴퓨터 범죄가 급속도로 증가하고 있다. 그러므로 이러한 컴퓨터 범죄를 막기 위한 침입 탐지 기술 개발은 매우 중요하다. 전통적인 침입 탐지 모델은 단지 네트워크 패킷 데이터만을 사용하고 있으며, 침입탐지 시스템의 성능을 높이기 위해 서로 다른 분류 알고리즘을 결합하는 방법을 사용해왔다. 그러나 이러한 모델은 일반적으로 성능향상에 있어서 제한적이다. 본 논문에서는 침입탐지 시스템의 성능을 개선하기 위해 네트워크 데이터와 시스템 콜 데이터를 융합하는 방법을 제안하였으며, 데이터 융합 모델로서 Multi-Layer Perceptron (MLP)를 사용하였다. 그리고 DARPA 에서 생성한 네트워크 데이터와 본 논문에서 가상으로 생성한 시스템 콜 데이터를 함께 결합하여 모델을 생성 한 뒤 실험을 수행하였다. 본 논문에서의 실험결과로, 단순히 네트워크 데이터만을 사용한 모델에 비해 시스템 콜 데이터를 함께 결합한 모델이 훨씬 더 높은 인식률을 보인다는 것을 확인할 수 있다.

1. 서 론

최근 인터넷, 인트라넷 등과 같은 정보 기술의 활용으로 인해 예전과는 달리 거의 모든 시스템이 개방 환경과 상호연결성에 노출되게 되었다. 이로 인해 얻게 되는 이익도 많은 반면 시스템 침입에 의한 정보의 유출과 파괴 같은 역기능도 기하급수적으로 늘고 있다. 질적, 양적으로 증가해 가는 컴퓨터 범죄를 막기 위한 국내의 기술 수준은 아주 미약하다고 볼 수 있다. 침입 탐지 시스템의 구현에는 침입유형의 다양성과 유동성, 침입행위의 자동성, 방대한 양의 데이터 조작과 같은 어려움이 존재한다. 따라서 이러한 어려움을 극복하기 위해서는 시스템이 능동적으로 불법 침입의 유형이나 패턴을 예측함으로써 날로 다양해지고 자동화되어 가는 침입 행위에 적극적으로 대응해 나갈 수 있도록 하는 것이 필요하다. 또한, 현재 연구되는 침입탐지 모델은 네트워크 정보와 사용자에 의한 정보를 분리하여, 각각의 모델을 생성한 뒤 침입에 대한 탐지를 수행하고 있다. 이러한 정보를 이용하여 생성된 모델에 의한 침지 탐지는 성능을 향상시키는데 한계를 가지게 된다. 따라서 이러한 문제점을 해결하기 위해서는 기존 데이터에 새로운 정보를 융합함으로서 정보의 상호보완을 통해 성능을 향상시키는 방법이 필요하다. 본 논문에서는 네트워크와 시스템 콜 시퀀스 정보를 함께 융합하여 모델을 생성하는 방법을 통해 기존 네트워크 정보만을 사용한 모델의 정보를 시스템 콜 시퀀스 정보를 통해 보완하는 방법을 제안하고자 한다.

2. 관련 연구

2.1 침입탐지 시스템의 개요

침입이란 정보 접근, 정보 조작 및 시스템 무기력화 등에 대한 고의적이고 불법적인 잠재가능성이나 자원의 무결성(integrity), 기밀성(confidentiality) 및 가용성(availability) 을 저해하는 일련의 행위를 말한다. 그리고 침입탐지란 침입을 시도하거나 침입 행위가 일어나고 있거나 침입이 발생한 것을 확인하는 절차를 말한다. 이러한 침입탐지 시스템은 오용(Misuse) 탐지와 비정상(Anomaly) 탐지의 두 가지로 나눌 수 있다[1].

2.2 기존 침입탐지 방법

2.1.1 다중 인식기 기반의 오용탐지

다중 인식기를 이루고 있는 기본 인식기는 MLP, RBFN, k-NN 과 C4.5 등의 다양한 인식기가 사용된다[2][3]. 인식기는 데이터 특성에 가장 적절한 인식기를 선택하도록 한다. 다중 인식기를 생성하기 위해서는 부스팅(Boosting) 또는 배깅(Bagging)에 의한 방법을 사용하며, 각 인식기에서 생성된 결과를 융합하는 방법으로는 다수결 의사 결정(Majority Voting) 또는 각 인식기의 확신도(Belief Value) 등의 방법을 이용한다[2][4]. 이러한 오용탐지 모델은 기존에 이미 발생한 패턴에 대한 탐지가 용이한 반면 새로운 공격에 대한 패턴에 대해서는 탐지가 어렵다는 문제점을 가지고 있다.

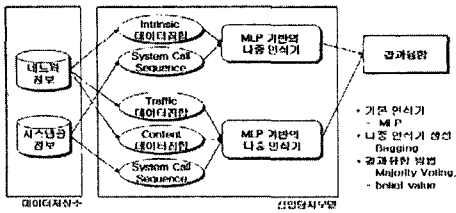
2.1.2 사용자 프로파일 정보를 이용한 비정상탐지

비정상탐지는 사용자에 의해 자주 사용되는 시스템 콜 정보나 시스템의 자원 정보 등을 이용하여 사용자가 사용하는 패턴을 이용하여 사용자 프로파일 정보를 구축한 뒤 새로운 정보가 들어온 경우 기존 사용자 프로파일의 정보를 이용하여 비정상 사용인지 아닌지를 판단하게 된다[5]. 이러한 탐지의 경우는 새로운 탐지 유형을 잘 판단할 수는 있지만 너무 잦은 경고를 발생하게 되는 문제점을 가지고 있다.

3. 연구내용

3.1 네트워크 정보와 시스템 콜 정보의 융합

본 논문에서는 네트워크 정보와 시스템 콜 정보를 융합함으로써 침입탐지 시스템의 성능을 높이고자 한다. 본 논문에서 제안하고자 하는 목표 시스템은 다음과 같다.



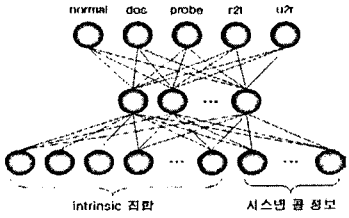
[그림3-1] 네트워크 정보와 시스템 콜 정보의 융합 모델

네트워크 정보는 한 사용자가 특정 서버에 접속한 후부터 접속을 완료한 시점까지의 패킷 정보를 말하며, 시스템 콜 정보는 사용자가 서버 접속 기간 동안 사용한 모든 명령어들의 집합을 말한다. 이러한 두 가지 정보를 융합함으로서 네트워크 정보에 의해 잘못 분류되는 정보를 보완할 수 있다.

[표 3-1] 시스템 콜 시퀀스 정보 융합 효과

Intrinsic	traffic	content	system call	class
지속시간:10시간 수신량:100M	...	...	send,send,send, send,send	dos
지속시간:10시간 수신량:100M	...	...	ftp,open,send, get,close	normal

3.1.1 Intrinsic 과 시스템 콜 데이터 집합의 학습방법



[그림 3-2] Intrinsic 과 시스템 콜 데이터집합에 대한 모델

모델을 생성하기 위해 신경망 모델인 MLP(Multi-Layer Perceptron)를 사용하였다. 히든노드의 수는 입력노드수의 1/3 이며 출력노드의 수는 5개(normal, dos, probe, r2l, u2r)로 설정하였다. 모델을 학습시키기 방법은 다음과 같다. 예를 들어 다음과 같은 데이터가 존재 한다고 가정한 경우,

[표 3-2] Intrinsic 과 시스템 콜 데이터 집합

Intrinsic	시스템 콜 시퀀스	클래스
0,tcp,http,SF,237,392,0,0	open,read,write,writ e,close	u2r

시스템 콜의 시퀀스 정보를 신경망에 적용시키기 위해서 다음과 같이 데이터를 변형하여 입력데이터 수를 늘리게 된다. 콜 시퀀스의 윈도우 사이즈를 3으로 설정한 경우 다음과 같은 데이터가 새로 생성된다.

[표 3-3] Intrinsic 과 시스템 콜 데이터의 변형된 입력 값

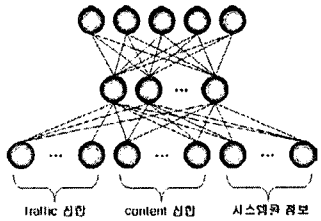
Intrinsic	시스템 콜 시퀀스	클래스
0,tcp,http,SF,237,392,0,0	open,read,write	u2r
0,tcp,http,SF,237,392,0,0	read,write,write	u2r
0,tcp,http,SF,237,392,0,0	write,write,close	u2r

생성한 모델을 평가가 하기 위해서는 다음과 같은 방법을 이용한다. 먼저, 모델을 평가하기 위한 데이터가 [표3-2]와 같이 존재한다고 가정할 경우 모델을 학습시키기 위한 방법과 동일한 방법으로 입력 데이터를 [표3-3]와 같이 변형하여, 생성된 모델의 입력 정보로서 사용한다. 생성된 정보를 통해 다음과 같은 방법을 적용하여, 입력 데이터에 대한 최종 클래스를 설정하게 된다. 즉, [표3-5]와 같이 각각의 입력데이터에 대해 가중치를 서로 다르게 적용한 후 각 클래스의 출력 값들을 더하여 가장 높은 출력 값을 갖는 클래스를 최종 클래스로 설정한다.

[표 3-4] 신경망 출력 값

	normal	dos	probe	r2l	u2r	가중치
입력1	0.1	0.4	0.2	0.1	0.3	1/3
입력2	0.1	0.2	0.1	0.3	0.7	2/3
입력3	0.1	0.1	0.1	0.1	0.8	3/3
합계	0.187	0.319	0.209	0.319	1.327	

3.2 Traffic, Content 및 시스템 콜 데이터 집합의 학습방법



[그림 3-3] traffic, content 및 시스템 콜 데이터 집합에 대한 모델

모델을 생성하기 위해 같은 방법으로 신경망 모델인 MLP 를 사용하였다. 데이터에 대한 모델은 [그림3-3]과 같으며, 모델의 학습과 평가 방법은 3.1.1과 동일하다.

3.2 가상 시스템 콜 정보의 생성

본 논문에서 필요로 하는 시스템 콜 정보가 존재하지 않기 때문에 가상으로 시스템 콜 정보를 생성함으로서 네트워크 정보에 대한 새로운 추가 정보로써 활용하였다. 가상 시스템 콜 정보를 생성하기 위해 다음과 같은 단계를 적용하였다.

[단계1] DARPA에서 제공한 데이터[6]와 속성정보[7]를 이용해서 각 클래스에 대해 normal(400), dos(400), probe(400), r2l(400), u2r(50) 를 추출하여 총 1650 개로 구성된 샘플 데이터를 생성한다.

[단계2] 가상 데이터에서 발생 할 수 있는 시스템 콜을 20개로 정의하였다. 사용한 시스템 콜 정보는 다음과 같다.

[표 3-5] 시스템 콜 종류

open(0), read(1), write(2), exit(3), kill(4), rmdir(5), mkdir(6), connect(7), access(8), nmap(9), sendto(10), sned(11), reboot(12), wait(13), execve(14), unmask(15), moune(16), readlink(17), ftp(18), fork(19)

[단계3] C4.5 에 대한 분기종료 조건으로서는 해당 노드에서의 최소 데이터 포함비율과 주 클래스의 비율을 적용하여 규칙을 생성하였다. 생성된 규칙의 형태는 다음과 같다.

[표 3-6] C4.5 에 의해 생성된 규칙

조건	결론	신뢰도(%)
(src_bytes<=28.0), (logged_in=0), (serror_rate>0.71)	class=dos	96.01

[단계4] 각 공격 유형별로 발생 가능한 콜 시퀀스 정보가 존재하지만 이 정보를 정확하게 알 수 없으므로, 본 논문에서는 임의적으로 각 클래스 별로 발생 가능한 콜 시퀀스 정보를 정의하였다.

[단계5] 단계 3에서 생성된 각 규칙에 대해 주어진 클래스와 다른 클래스의 규칙을 추가로 생성한 후 각 클래스에 대한 콜 시퀀스 정보를 추가한 새로운 규칙리스트를 생성한다.

[표 3-7 초기규칙

조건	결론
(count<=90.0), (src_bytes<=18.0)	class=probe

[표 3-8] 콜 시퀀스 추가와 함께 확장된 규칙

조건	결론
(count<=90.0), (src_bytes<=18.0), **(calls=0:1:2:3:4)**	class=probe
(count<=90.0), (src_bytes<=18.0), **(calls=3:4:5:6:7)**	class=dos

[단계6] 클래스에 대한 충돌이 발생하는 비율을 고려하여 여러 데이터를 생성한다. 클래스의 충돌이란 콜 시퀀스 정보를 제외한 네트워크 정보는 같지만 클래스 정보가 다른 경우를 말한다.

4. 실험 및 결과

4.1 실험방법

실험을 위해 두 가지의 신경망 모델을 생성하였다. 한 모델은 {Intrinsic, 시스템 콜 시퀀스} 데이터에 대한 신경망 모델이며, 다른 한 모델은 {content, traffic, 시스템 콜 시퀀스} 데이터에 대한 신경망 모델이다. 모델을 생성하는데 있어 단일 인식기에 비해 다중 인식기를 사용하는 경우의 성능에서 효율적이므로, 다중 인식기를 생성하기 위해 배깅(Bagging)에 의한 방법을 사용하였다[2][4]. 두 모델에 대해 각각 3개씩의 인식기를 생성하도록 하여, 총 6개의 모델을 생성하여 학습을 수행하였다.

4.2 실험결과

실제 데이터에서는 네트워크 정보는 같지만 시스템 콜 시퀀스 정보에 의해 클래스가 달라질 수 있기 때문에 콜 시퀀스 정보의 사용에 따라 사용하지 않은 모델에 비해 보다 나은 성능을 기대할 수 있다.

• 첫 번째 실험으로서 콜 시퀀스에 대한 윈도우 크기의 변화에 따른 비교를 수행하였다.

[표4-1] 윈도우 크기에 따른 시간 및 평균 인식률

	call size	window size	평균학습 시간(초)	평균 인식률(%)
test_01	7	2	271	92.35
test_02	7	3	335	93.76
test_03	7	4	190	99.65
test_04	7	5	66	99.76
test_05	7	6	35	100.00
test_06	7	7	15	100.00

실험 결과를 살펴보면, 윈도우 사이즈가 증가함에 따라 인식률이 증가하는 것을 볼 수 있다. 윈도우 사이즈가 작은 경우는 각 데이터에 대한 클래스를 결정하기 위한 콜 시퀀스 정보가 충돌이 일어나는 경우가 많기 때문에 성능이 낮아지게 된다. 반면에 윈도우 사이즈가 커진 경우 학습된 데이터에 종속적으로 학습이 되므로, 새로운 데이터에 대해서는 인식률이 떨어지게 된다. 따라서 본 논문에서는 이러한 사항들을 고려하여 콜 사이즈를 7로 설정하고 윈도우 사이즈를 4로 설정하여 다음 실험을 수행하였다.

• 두 번째 실험으로서 콜 시퀀스 정보를 제외한 후, 클래스의 충돌 비율을 고려하여 실험을 하였다.

[표4-2] 콜 시퀀스를 제외한 후의 실험 결과

	0.00	0.1(84)	0.2(170)	0.3(254)	0.4(340)	0.5(424)
평균 인식률	92.12	85.65	81.88	76.24	71.53	67.41

실험 결과를 보면, 클래스의 충돌 비율을 정차 늘려감에 따라 평균인식률이 낮아지는 것을 볼 수 있다. 이는 콜 시퀀스 정보가 없기 때문에 클래스가 충돌되는 경우를 학습할 수 없기 때문이다.

• 세 번째 실험으로서 콜 시퀀스 정보를 포함한 후, 클래스의 충돌 비율을 고려하여 실험을 하였다.

[표4-3] 콜 시퀀스를 포함한 후의 실험 결과

	0.00	0.1(84)	0.2(170)	0.3(254)	0.4(340)	0.5(424)
평균 인식률	100.00	98.47	99.18	99.76	99.53	99.65

실험 결과를 보면, 클래스의 충돌 비율을 정차 늘려가더라도 평균인식률에는 큰 변화가 없는 것을 볼 수 있다. 이는 네트워크 정보만으로는 구분이 불가능 했던 정보들에 대해 콜 시퀀스 정보를 활용함으로서 이들을 구분할 수 있게 되었기 때문이다.

5. 결론 및 향후연구

본 논문에서는 효율적인 침입탐지를 위해 기존 네트워크 정보와 사용자가 입력한 시스템 콜 시퀀스 정보를 융합함으로서 네트워크 정보만을 이용한 경우의 잘못 인식되는 정보를 보완해 주는 방법을 제안하였다. 네트워크 정보만을 이용하는 경우는 사용자에 의한 사용자의 패턴을 고려하지 못하기 때문에 침입이 아닌 경우를 침입으로 분류하는 문제점을 가지게 되지만, 논문에서 제안한 방법으로 사용자가 입력된 시스템 콜 정보를 이용함으로서 이러한 문제점을 해결함과 동시에 전체적으로 좀 더 높은 인식률을 보이게 되는 것을 실험을 통해 확인하였다. 시스템 콜 데이터가 존재 하지 않기 때문에 본 논문에서는 이러한 데이터를 가상으로 생성하는 방법을 제안하여 모델의 성능을 평가하였다. 본 논문에서의 가상 콜 시퀀스 정보는 순차정보 안에 불필요한 콜 데이터가 없는 경우로 제한하였다. 하지만 실제 데이터 상에서는 시퀀스 정보에 불필요한 정보가 포함된 경우가 발생할 수 있다. 이러한 데이터에서의 유용한 정보를 추출하기 위해 순차패턴과 같은 방법을 사용할 수 있을 것이다.

6. 참고문헌

[1] Yuebin Bai, Hidetsune Kobayashi, Intrusion Detection Systems:Technology and Development. Advanced Information Networking and Applications, 2003 IEEE Proceedings, 17th International Conference on, 27-29 March 2003 Page(s): 710 -715.

[2] Giorgio Giacinto and Fabio Roli, Intrusion Detection in Computer Networks by Multiple Classifier Systems, Pattern Recognition, 2002 IEEE Proceedings.

[3] Partrick Verlinde, Gerard Chollet, Comparing decision fusion paradigms using k-NN based classifiers, decision trees and logistic regression in a multi-modal identity verification application. in Proceedings AVBPA'99, Washington D.C., USA, March 1999, pp. 188-193.

[4] Ruta D, Gabrys B. An Overview of Classifier Fusion Methods. Computing and Information Systems 7(1):1-10, University of Paisley, 2000.

[5] Henry S.Teng, Kaihu Chen, Stephen C-Y Lu, Adaptive Real-time Anomaly Detection Using Inductively Generated Sequential Patterns. 1990 IEEE Computer Society Symposium on, 7-9 May 1990 Page(s): 278 -284.

[6] Lee, W. and Stolfo, S. J., A Framework for Constructing Features and Models for Intrusion Detection Systems. ACM Transactions on Information and System Security, vol. 3, November, 2000.

[7] UCI KDD Intrusion DataSet, http://kdd.ics.uci.edu/databases/kddcup99/kddcup99.html