위협수위 기반 AOI 클러스터링 기법

김순동[O] 서정택* 김도환* 이도훈* 김동규 채송화
아주대학교 정보통신전문대학원, *국가보안기술연구소
{sdkim[O], dkkim, portula}@ajou.ac.kr, *{seojt, dkim, dohoon}@etri.re.kr

Threat Level Based Attribute Oriented Induction

Soon-Dong Kim[O], Jung-Taek Seo*, Do-Whan Kim*, Do-Hoon Lee*, Dong-Kyoo Kim, Song-Hwa Chae
GSIC at Ajou University, *NSRI(National Security Research Institute)

요 약

지난 10년간 네트워크 기반의 컴퓨터 공격은 급격히 증가했으며 이에 따라 보안 기술도 발달하게 되었다. 침입탐지시스템은 컴퓨터 보안 기술로써 발전되어 왔으나 과도한 침입시도정보의 발생과 그 대부분이 긍정오류(false positive)를 발생시킴으로써 실제로 관리하는데 많은 어려움을 준다. 이러한 문제에 대안으로 여러 연구들이 진행되어 왔으며, 침입시도정보의 축약을 통한 관리적 측면에서의 효율을 높이는 연구도 진행되고 있다. 그러한 연구들의 한 방법으로서 속성중심귀납법(Attribute Oriented Induction, 이하 AOI)은 침입시도정보를 속성정보에 기반 하여 의미 있는 묶음으로 클러스터링 하는 방식이다. 본 논문은 이 방식에서의 문제점을 분석하였으며 그 해결책으로써 본 논문에서는 위협수위 기반 AOI 클러스터링 기법을 제시하였다.

1. 서 론

지난 10 년 동안 네트워크 기반의 컴퓨터 공격은 급격히 증가했다 [1]. 이에 따라 암호학 혹은 인증과 같은 기존의 컴퓨터 보안기술들의 중요성이 더욱 강조되었으며, 새로운 접근 방식으로 침입을 탐지 하는 방법이 대두되었다. 침입탐지 시스템은 시스템의 이벤트들을 모니터하고 분석함으로서 보안위배사항에 대한 탐지를 하며 해당 침입정보에 대한 경보(이하 침입시도정보)를 발생시킨다. 이전에 많은 연구들에서 이러한 침입탐지시스템들이 하루에 수천 개의 침입시도정보를 발생시키고 이들 중 대다수가 긍정오류(false positive)라고 보고 된 바 있다 [2]. 이러한 잘못된 공격들에 대한 침입시도정보의 발생은 실제 공격을 가려내는 것을 아주 어렵게 만든다. 많은 공격들 중에 유효하고 치명적인 공격을 일일이 관리자가 가려내는 일은 많은 시간과 노력을 필요로 하며, 잘못된 판단의 원인이 되기도 한다. 자동화된 방식들이 존재하기도 하지만 아직까지 완전한 해결책으로 보기엔 어렵다.

침입시도정보의 축약의 관점에서 클러스터링은 동일한 침입시도정보를 하나로 묶어서 관리하여 관리자의 수고를 덜어줌과 동시에 클러스터의 생성과 그 분포를 분석함으로서 현재 목적 시스템에 시도되는 공격에 대해서 분석하는 것에 용이하게 사용될 수 있다. 본 논문에서는 침입시도정보를 축약하는 방법 중 하나인 AOI 이라는 기존의 클러스터링 방법에 대하여 분석하고 좀더 발전된 AOI기법에 대해서 제안한다.

본 논문은 다음과 같이 구성하였다. 2장에서는 전통적인 AOI 기법과 최근에 수정된 방식에 대해서 분석하게 될 것이다. 3장에서는 본 논문에서 제안하는 방식에 대해서 기술하고 4장에서는 실험과 결과를 보여줄 것이다. 5장에서는 결론으로 본 논문을 마무리 하겠다.

2. AOI기법

AOI기법은 침입탐지시스템에서 발생하는 침입시도정보를 클러스터링 방법을 통해서 축약하는 방식이다. 최초의 AOI 기법[3] 은 데이터베이스 분야에서 데이터를 요약하는데 이용하기 위해서 소개되었다. 2.1에서 전통적인 AOI기법들을 알아보고 그 문제점들을 알아보겠다.

2.1 전통적인 AOI알고리즘

AOI는 관계형 데이터베이스의 테이블 상에서 반복적으로 속성 값을 보다 추상화된 값으로 치환하는 연산을 한다. 이때 보다 추상적인 값은 일반화 구조(generalization hierarchy)에서 상위 노드의 값이 된다. 이런 방식으로 반복하면서 거대한 관계형 데이터베이스 테이블은 간결하고 추상화된 요약 테이블이 된다. 그림 1은 IP속성의 일반화 구조를 표현한 예이다.

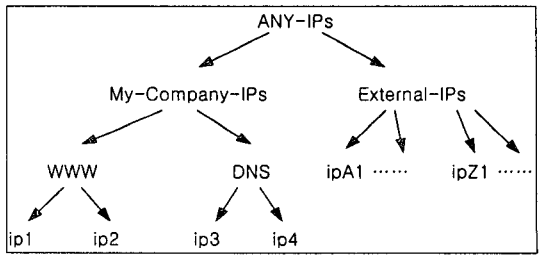

그림 1. IP속성의 일반화 구조 예

```
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 A_i;
4:     for all alarms a in T do        // Generalize A_i
5:         a.A_i := father of a.A_i in H_i;
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 전통적인 AOI 알고리즘

AOI 알고리즘은 그림 2에 나타내었다. 그림 1은 침입시도정보의 속성 관계형 테이블 T의 일반화 구조 H를 나타낸다. C는 해당 속성을 갖는 침입시도정보의 개수를 나타내며, 속성은 $\{A_1,\ldots,A_n,C\}$로 나타낸다. 알고리즘의 첫 번째 단계는 모든 침입시도정보의 개수를 1로 설정하는 것이다. 이어서 2~8번 루프가 반복된다. 3번째 단계에서 특정 속성 A_i를 선택하고 4, 5 번째 단계를 통해 특정 속성 일반화 구조 H상의 상위 속성 값으로 일반화 할 것인지 결정하고 시행한다. 6, 7단계에서는 서로 다른 두 침입시도정보들의 속성 값이 공통의 상위 속성 값으로 치환되면서 하나로 통합되게 된다. 이때 두 침입시도정보의 발생 횟수를 합쳐서 그 값을 일반화된 C로 가지게 된다. 이러한 반복을 통해서 일반화된 침입시도정보들이 하나의 클러스터를 이루게 되고 만들어진 클러스터의 숫자가 구분 임계 값보다 작아지게 되면 반복을 중단한다.

2.2 수정된 AOI 알고리즘

전통적인 AOI알고리즘은 결국 임계값에 의해 클러스터 수가 결정되며, 클러스터의 수가 임계값보다 작아질 때까지 일반화를 계속 하게 된다. 결과적으로 침입시도정보의 자세한 정보를 전달하지 못하고 과도한 일반화를 하게 될 우려가 있다. 예를 들어 동일한 공격에 의해 발생된 동일한 침입시도정보가 1000 개이고 다른 침입시도정보들은 모두 서로 다른 100개의 공격의 의한 것이었다고 하여도 구분 임계값이 30 이면 1000개의 동일한 침입시도정보가 다른 침입시도정보와 동일하게 일반화 된다. K. Julisch는 이러한 문제점을 이미 지적한 바 있으며 그에 대한 해결책으로 구분 임계값을 없애고 최소 크기 값을 도입함으로서 해결하였다. 기존의 알고리즘에서 하나의 클러스터가 최소 크기값을 넘기게 되면 그 침입시도정보(혹은 일반화된 침입시도정보) 들은 하나의 클러스터로서 취급되게 되고 더 이상 일반화를 반복하지 않는다. 일반화를 하게 될 속성을 선택하는 방식도 수정을 하였으며 또한 하나의 클러스터를 생성한 침입시도정보들은 기존의 테이블에서 분리한 후 다시 남은 침입시도정보들을 다시 클러스터링 하는 방식으로 알고리즘을 수정하였다 [4].

3. AOI 클러스터링 방법의 수정제안

K. Julisch는 기존의 방식의 문제점을 지적한 바 있으며 그에 대한 대안을 내놓았다. 하지만 좀더 살펴보면 고려해야 할 점들이 있다. 특히 최소 크기 값은 AOI 알고리즘의 수행 결과에 가장 크게 영향을 미치는 요소이고 이에 대한 고찰이 필요하다. 최소 크기 값이 크면 침입시도정보가 지나치게 일반화 되어 나타날 것이며 값이 너무 작을 경우는 지나치게 상세한 정보를 그대로 반영하여 본연의 목적을 달성할 수 없게 된다. 기존의 논문에서 획일화된 최소 크기 값을 적용하여 알고리즘을 수행하였으며 결과적으로 치명적인 침입시도정보를 간과 할 수 있다는 문제점이 발생한다. 예를 들어 어떤 시스템에 치명적인 침입시도정보가 발생했고 이 침입시도정보는 기존의 발생한 침입시도들과는 공통점이 거의 없고 발생 횟수도 적다고 가정하자. 이럴 경우 이 침입시도정보는 가장 상위 속성값으로 일반화되어 나타나고 관리자는 이 공격에 대해 많은 신경을 쓰기 어려울 것이다. 보안 위협 수위가 높은 공격들은 좀더 관리자가 주위를 기울여 줄 수 있도록 클러스터링 되는 것이 더욱 옳은 방식일 것이다. 따라서 기존 AOI 알고리즘은 소수의 치명적인 공격에 대해서 비의도적으로 은닉할 수 있는 여지가 있다고 할 수 있다.

이러한 문제점을 해결하기 위해서 본 논문에서는 위협 수위기반 AOI 클러스터링을 제안한다. 이 방식은 침입시도정보가 시스템에 미칠 수 있는 피해의 정도에 따른 가중치를 설정하고 그 정보들을 이용하여 클러스터를 생성하는 방식이다. 기본적인 알고리즘은 K. Julisch의 방식을 그대로 따르지만 최소 크기와 침입시도정보의 개수를 비교하는 것이 아니라 각각의 침입시도정보가 가지는 가중치의 환산 값을 합산한 값과 최소 크기 값을 비교하여 클러스터를 생성한다는 점이 다르다. K. Julisch는 최초의 모든 침입시도정보의 개수를 1로 설정하였지만 본 논문에서는 침입시도정보의 가중치를 반영하여 최초의 개수를 책정하였다. 예를들어 침입시도정보가 가중치가 높다면 1번 발생한 것이 1.5번 혹은 2번 발생한 것으로 계산하는 방식이다. 이런 방식을 도입할 경우, 침입시도 정보의 개수가 상대적으로 적다고 하여도 부여된 가중치로 환산하기 때문에 치명적인 공격들에 무게를 싣게 된다.

4. 실험 및 결과

실험을 위해 그림 1과 같은 일반화 구조를 가지는 가상의 네트워크를 구성하였다. 본 논문에서는 침입탐지시스템으로써 네트워크 기반의 침입탐지시스템인 Snort[5]를 이용하여 구현하였으며, Snort에 알려진 공격들을 무작위로 발생시키고 Snort에서 발생한 침입시도정보들을 이용하여 테스트 하였다. 각각의 공격들은 공격 유형을 가지고 있으며 우선순위가 높은 공격을 1로, 가장 위협 수위가 낮은 것을 5로 설정하였다. 합산 시 부여되는 가중치 환산 값은 표 1과 같이 설정하였다. 이 값은 실험에 의해 관리자가 조정할 수 있다. 각각의 침입시도정보들은 이 환산 값들의 합을 클러스터의 최소 크기와 비교를 하게 된다. 현재 공격 클래스와 ip주소, 포트를 일반화 구조로 생성하였으며 각각은 그림 1, 3에 나타나 있다. 침입시도정보의 시그니쳐 식별자(Sid)는 상위 한 단

계만 존재하며 그 값은 'ANY'이다. 최소 크기 값은 100으로 하였다.

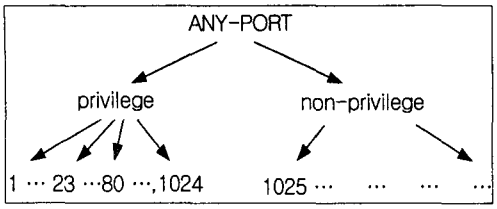

그림 3 포트 일반화 구조

표 1. 가중치에 따른 환산값

가중치	1	2	3	4	5
환산값	5	4	3	2	1

표 2. K. Julisch 방식 결과 - 최소크기 100

Sid	source IP	Target IP	Source Port	Dest Port	Count
7	External-IPs	ip1	undefined	undefined	162
1	External-IPs	ip1	non-privilege	80	596
5	External-IPs	ip1	non-privilege	80	138
ANY	External-IPs	ip1	undefined	undefined	147
ANY	External-IPs	ip1	non-privilege	privilege	103
ANY	External-IPs	ip1	non-privilege	non-privilege	135
ANY	ANY-IPS	ANY-IPs	ANY-PORT	ANY-PORT	112

표 3. 위협수위 기반 AOI 결과1 - 최소크기 100

Sid	source IP	Target IP	Source Port	Dest Port	Count
7	External-IPs	ip1	undefined	undefined	648
8	External-IPs	ip1	undefined	undefined	177
1	External-IPs	ip1	non-privilege	80	1788
5	External-IPs	ip1	non-privilege	80	414
ANY	External-IPs	ip1	undefined	undefined	333
ANY	External-IPs	ip1	non-privilege	80	115
ANY	External-IPs	ip1	non-privilege	privilege	288
ANY	External-IPs	ip1	non-privilege	non-privilege	485
ANY	External-IPs	ip1	ANY-PORT	ANY-PORT	189

표 2는 K. Julisch가 제안한 방법으로 인해 AOI를 수행한 결과이고 표 3은 본 논문에서 제안하는 방법으로 클러스터링 한 결과이다. 최소 크기는 100으로 하였다. 또한 표 4는 제안하는 방법으로 최소 크기를 300으로 하여 실험한 결과이다. 세 경우 모두 source IP는 External-IPs로 나타났으며 대부분의 공격이 외부에서 들어왔기 때문이다. 또한 Target IP가 항상 ip1인 것은 현재 실험에 사용된 snort가 ip1에 설치되어 동작하기 때문에 ip1에 발생하는 침입시도정보만을 모아서 테스트했기 때문이다. WWW혹은 My-Company의 게이트웨이 등지에 설치된 침입탐지시스템에서 실험하였다면 Target IP속성은 더욱 다양한 값을 나타낼 것이다.

보이는 바와 같이 본 논문에서 제안하는 방식이 2개 더 많은 클러스터를 생성하였다. Sid가 8인 클러스터는 가중치가 3인 공격이었다. 기존의 AOI방식으로는 생성되지 않을 클러스터가 새로 생성된 것이다. 새로 생성된 클러스터는 본래 최소크기 보다 적은 숫자의 침입시도이지만 위협정도가 큰 공격이었으므로 구분해 내는 것이 합당하다고 할 수 있다.

표 4에서는 클러스터의 최소크기를 300으로 설정하였다. 이 경우 K. Julisch 방식보다 적은수의 클러스터를 생성하게 되었다. 이는 가중치가 낮은 공격들이 하나의 클러스터를 생성하지 못하게 되어서 발생한 결과이다.

표 4. 위협수위기반 AOI 결과2 - 최소 크기 300

Sid	source IP	Target IP	Source Port	Dest Port	Count
7	External-IPs	ip1	undefined	privilege	648
1	External-IPs	ip1	non-privilege	80	7188
5	External-IPs	ip1	non-privilege	80	414
ANY	External-IPs	ip1	undefined	undefined	510
ANY	External-IPs	ip1	non-privilege	privilege	403
ANY	External-IPs	ip1	non-privilege	non-privilege	485

5. 결론 및 향후과제

본 논문에서는 침입시도정보를 축약하고 관리하기 편하게 하는 방법으로 AOI를 살펴보았고 좀더 개선된 방안인 위협수위 기반 AOI를 제안하였다. 앞에서 살펴본 바와 같이 위협수위가 낮은 공격은 클러스터를 생성하는데 더 많은 수의 침입시도정보가 필요하게 되었고 위협수이가 높은 공격들은 좀더 적은 수의 침입시도정보로 클러스터를 생성할 수 있었다. 이 방식은 위협수위가 높은 공격에는 좀더 가중치를 두는 방법을 통하여 좀더 실제적인 위협에 대처할 수 있도록 하는데 도움을 줄 것이다. 위협수위가 낮은 침입시도정보에 대해서 등한시해도 된다는 것은 아니지만 한정된 시간에 효과적인 침입시도정보를 관리하기위한 한 방법이 될 것이다.

앞으로 침입시도정보 내의 다른 속성들을 이용하여 일반화 구조를 생성하여 보다 공격의 속성을 잘 반영된 AOI 알고리즘을 구현 할 수 있을 것으로 예상된다. 또한 평상시 발생되는 침입시도정보의 양에 따라 차등화 된 최소 크기의 설정을 통해 침입경향에 변화에 민감히 대처할 수 있도록 하는 방안도 생각해볼 과제이다.

참고문헌

[1] J. Allen, A. Christie, W. Fithen, J. McHugh, J. Pickel, and E. Stoner. State of the Practice of Intrusion Detection Technologies. Technical report, Carnegie Mellon University, January 2000. http://www.cert.org/archive/pdf/99tr028.pdf.

[2] E. Bloedorn, B. Hill, A. Christiansen, C. Skorupka, L. Talboot, and J. Tivel. Data Mining for Improving Intrusion Detection, 2000. http://www.mitre.org/support/papers/tech_papers99_00/

[3] J. Han, Y. Cai, and N. Cercone. Data-Driven Discovery of Quantitative Rules in Relational Databases. IEEE Transactions on Knowledge and Data Engineering, 5(1):29.40, 1993.

[4] K. Julisch and M. Dacier. Mining Intrusion Detection Alarms for Actionable Knowledge. Proc. 8th ACM International Conference on Knowledge Discovery and Data Mining, Edmonton, July 2002.

[5] Snort. http://www.snort.org