침입시도정보 클러스터링의 과도한 일반화 방지를 위한 AOI 알고리즘 개선 방법에 대한 연구

김정태[1O], 서정택[2], 이은영[2], 박응기[2], 이건희[1], 김동규[1]
[1]아주대학교 정보통신전문 대학원, [2]국가 보안기술 연구소
{cool[O], dkkim, icezzoco}@ajou.ac.kr, {seojt, eylee, ekpark}@etri.re.kr

A Study on Modification of Attribute Oriented Induction to Prevent Over-generalization

Jungtae Kim[1O], Jung-Taek Seo[2], Eun-young Lee[2], Eung-ki Park[2], Gunhee Lee[1], Dong-kyoo Kim[1]
[1]Graduate School of Information Communication, Ajou Univ. [2]National Security Research Institute

요 약

현재 침입탐지시스템이 직면하고 있는 문제점중 하나는 침입탐지시스템이 발생시키는 경보의 수가 실제 공격 횟수에 비해 너무 많다는 점이다. 따라서 침입시도 정보관리자의 경보 분석을 도와 줄 수 있는 침입시도 정보 클러스터링 기법들이 최근 소개 되고 있다. AOI를 이용한 기법 또한 그중 하나이다. 하지만 기존 AOI 알고리즘은 속성 값에 대한 과도한 일반화를 발생 시키는 문제점이 있다. 이에 본 논문에서는 AOI 알고리즘의 속성 일반화 단계 수정을 통한 과도한 일반화 방지에 대한 방법을 제시하였다.

1. 서 론

네트워크 기술의 발전과 더불어 이를 이용한 네트워크 기반의 공격들 또한 급격히 증가 하고 있다. 따라서 이러한 공격 위협으로부터 시스템을 안전하게 보호하기 위한 고전적인 암호화 기술과 인증 기술 등의 중요성이 증가 하고 있다. 동시에 침입을 빠르게 파악할 수 있는 침입탐지시스템의 중요성 또 한 증가하고 있다 [1]. 침입탐지시스템에서는 미리 정의된 규칙들에 의해 공격을 탐지하게 되면 경보(alert)를 발생 시키고 이를 시스템 관리자에게 알리게 된다. 하지만 현재 침입탐지시스템이 직면하고 있는 문제점중 하나는 침입탐지 시스템이 발생 시키는 경보의 수가 너무 많다는 점이다[2]. 이는 시스템 관리자가 침입에 대한 빠른 대응을 하지 못하도록 하는 주요한 원인이 되고 있다.

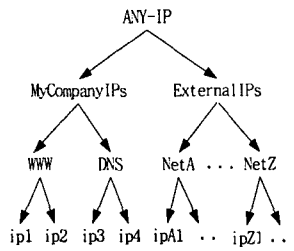
따라서 과도하게 발생한 경보의 분석을 조금 더 쉽게 할 수 있도록 하는 침입탐지시스템 경보 클러스터링 기법들이 최근 소개되고 있다. 속성중심 귀납법을 이용한 방법 또한 그중의 하나이다 [3]. 하지만 기존 속성중심 귀납법은 원본 경보가 과도하게 일반화 되어 원본의 의미를 잃어버리는 과도한 일반화 (over-generalization)문제가 발생한다. 이러한 과도한일반화는 시스템 관리자의 명확한 원인 분석을 어렵게 한다. K. Julisch의 연구에서는 기존 속성중심 귀납법을 침입탐지시스템 경보 클러스터링에 적합하도록 개선함으로써 이 문제에 대해 개선을 시도했다 [1]. 하지만 K. Julisch의 연구 결과에서도 여전히 과도한일반화 문제가 존재한다. 따라서 본 논문에서는 K. Julisch의 알고리즘을 수정하여 경보 클러스터링시 발생하는 과도한일반화를 방지하는 방법을 제안 하고자 한다.

2. 속성중심 귀납법 (Attribute Oriented Induction:AOI)

속성중심 귀납법의 일반적인 원리는 먼저 [그림 1-a]와 같이 관계형 데이터베이스에 클러스터링에 사용할 데이터를 저장한 후 모든 속성(attribute)에 대해 속성내의 서로 다른 값의 개수가 미리 정의된 임계값(threshold) 미만이 될 때까지 [그림 1-b]와 같은 각 속성의 개념 계층도를 이용하여 일반화를 진행하게 된다. 일반화가 진행되는 동안 모든 속성에 대해 동일한 속성 값을 갖는 경보들은 하나로 합쳐지며 하나의 클러스터를 형성하게 된다. 이러한 과정을 통해 최종적으로 [표 1]과 같은 클러스터링 결과를 생성하게 된다.[3]

Src-IP	Dest-IP	Count
ip1	ip4	1000
ip1	ipA1	1
ip1	ipB1	1
...	...	..
ip1	ipZ1	1
ipA1	ip4	1
ipB1	ip4	1
...	...	..
ipZ1	ip4	1

a) 경보에 대한 속성 테이블



b) IP에 대한 개념 계층도

그림 1. 속성 테이블과 개념 계층도

표 1. 경보 속성 테이블에 대한 클러스터링 결과

Src-IP	Dst-IP	Count
MyCompanyIPs	MyCompanyIPs	1000
MycompanyIPs	ExternalIPs	26
ExternalIPs	MyCompanyIPs	26

하지만 단순히 하나의 임계 값에 기반을 두어 속성들을 일반화 할 경우 [표 1]에서 나타난바와 같이 과도한 일반화가 일어나 시스템 관리자의 결과 분석을 더욱 어렵게 만들 가능성이 크다. 왜냐하면 *MyComPanyIPs*에서 *MyCompanyIPs*로 가는 공격을 생각하는 것 보다 ip1에서 ip4로 가는 공격을 생각하는 것이 공격의 분석에 더욱 명확한 의미를 전달하기 때문이다. 이에 K. Julisch

는 기존 알고리즘의 수정을 통해 임계 값 대신 최소크기(min_size)를 클러스터링의 기준으로 사용하여 과도한 일반화 문제에 대한 개선을 시도 하였다. 여기서 최소 크기란 클러스터링을 통해 경보의 군집을 형성하였을 때 하나의 클러스터가 되기 위한 군집의 최소 크기를 의미한다. 최소크기 사용을 통해 얻을 수 있는 가장 큰 장점은 과도한일반화를 방지할 수 있다는 점이다. 예를 들어 최소크기를 26으로 가정하였을 경우 [그림 1-a]에서 속성 값이 ip1, ip4인 경보의 수는 26 이상이므로 하나의 클러스터가 형성된다. 따라서 다음번 클러스터링 단계에서는 클러스터가 형성된 나머지 경보들에 대해 클러스터링을 진행하게 되고 최종적으로는 [표 2]와 같은 결과로 표현된다.

표 2. 개선된 알고리즘을 통한 클러스터링 결과

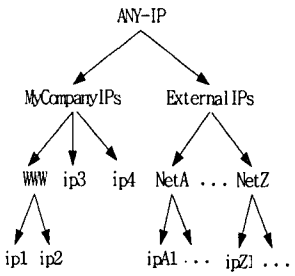
Src-IP	Dst-IP	Count
ip1	ip4	1000
ip1	ExternalIPs	26
ExternalIPs	ip4	26

3. 기존 알고리즘 분석

3.1 기존 알고리즘의 문제점

개선된 AOI 알고리즘이 과도한 일반화문제에 대한 어느 정도의 개선을 가져오기는 하였지만 일반화 계층도의 구성에 따라 발생하는 과도한 일반화 문제는 해결하지 못했다. 예를 들어 [그림 2]와 같은 IP에 대한 개념 계층도와 경보에 대한 속성 테이블과 최소크기 26을 가정해 보자. K. Julisch의 알고리즘을 [그림 2]의 예제와 같이 균형이 맞지 않은 개념 계층도에 그대로 적용할 경우 생성되는 클러스터는 [표 3]과 같다.

Src-IP	Dest-IP	Count
ip1	ip4	10
ip1	ip2	10
ip1	ip3	10



a) 경보에 대한 속성 테이블 b) IP에 대한 개념 계층도

그림 2. 비균형적 개념 계층도와 속성 테이블

결과에서 알 수 있듯이 Dest-IP의 경우 MyCompanyIPs단계에서 일반화 되는 것이 타당함에도 불구하고 ANY-IP로 일반화 되어 있다. 이것은 leaf node의 깊어가 서로 다른데서 기인하는 문제이다. 즉 첫 번째 일반화 단계에서 *ip2→WWW, ip3→MyCompanyIPs, ip4→MyCompanyIPs*로 일반화 되고 두 번째 단계에서 *WWW→MyCompanyIPs, MyCompanyIPs→ANY-IP*, 세 번째 단계에서 *MyCompanyIPs→ANY-IP*로 일반화가 진행됨으로 인해 과도한 일반화 문제가 발생하는 것이다.

표 3. 비균형적 개념 계층도에 대한 클러스터링 결과

Src-IP	Dest-IP	Count
ip1	ANY-IP	30

3.2 고려사항

이것을 해결 하는 방법으로 크게 세 가지를 생각해 볼 수 있다. 첫 번째 방법은 속성에 대한 개념 계층도를 형성할 때 균형 잡힌 개념 계층도만을 만드는 것이다. 하지만 개념 계층도는 속성의 특징에 따라 달라지며, 속성의 특징을 최대한 반영하여야 하기 때문에 정형화된 형태로 규정지어 두는 것은 타당 하지 못하다.

두 번째 방법은 균형 잡히지 않은 개념 계층도를 이용하되 가상의 노드를 이용하여 개념 계층의 균형을 맞추어 주는 방법이다. 하지만 가상의 노드를 이용하게 될 경우 불필요한 오버헤드가 발생한다는 문제점이 있다. 즉 노드가 증가 할 때마다 일반화 단계가 증가 하게 됨으로 노드의 증가 수만큼 데이터베이스의 업데이트 횟수도 증가한다. 이는 데이터의 양이 적을 경우 큰 문제로 작용 하지 않을 수 있지만 데이터의 양이 증가할 경우 큰 문제로 작용 한다. 실제로 약 50,000개의 데이터를 이용하여 실험 하였을 경우 가상의 노드가 하나 증가 할 때 마다 약 4%정도의 클러스터링 시간 증가가 나타났다.

세 번째 방법은 기존 알고리즘의 속성 일반화 과정을 수정하여 과도한 일반화를 방지하는 방법으로 추가적인 오버헤드의 발생 없이도 과도한 일반화를 효과적으로 방지 할 수 있다.

4. 제안하는 알고리즘

제안하는 알고리즘은 기존 알고리즘의 속성 일반화 단계를 [그림 3]과 같이 수정하였다. 제안하는 알고리즘 에서는 일반화를 진행하기 전에 개념계층도 상의 각 노드에 대하여 정지계수(hold count)라는 속성을 부여하게 된다. 이 속성의 값은 각 노드의 서브트리(sub tree)의 깊이에 의해 결정되는 값으로 만약 모든 서브트리의 깊이가 동일하거나 서브트리가 하나인 경우 정지계수는 0으로 설정되고 만약 서브트리간의 깊이가 다르다면 깊이가 가장 깊은 서브트리의 깊이로 설정된다. 예를 들어 [그림 2-b]에서 ExternalIPs의 경우 서브트리간의 깊이가 동일하므로 정지계수는 0으로 설정된다. 반면 MyCompanyIPs의 경우 서브트리간의 깊이가 동일하지 않으므로 정지계수는 서브트리의 깊이가 가장 깊은 2로 설정 된다.

```
1: T := Store log L in table T
2: G_i[H_k] := hold count of attribute value in generalization
              hierarchy G_i
3:   for all alarms a in T do a[count] := 1 // initialize counts
4:   while all a ∈ T: a[count] < min size do {
5:     Use heuristics to select an attribute A_i, i ∈{1..... n}
6:     for all alarms a in T do //generalize attribute A_i
7:       for all attribute value of a[A_i] do
8:         if(G_i[H_k] = 0)
9:           a[A_i][V_k] := father of a[A_i][V_k] in G_i
10:          G_i[H_k] := G_i[H_k] - 1
11:    while identical alarms a, a' exist do
12:      Set a[count] := a[count] + a'[count] and delete
                       a' from T
}
```

그림 3. 경보 클러스터링 알고리즘

이 값은 속성의 일반화 단계에서 사용되는 값으로 만약 개념 계

층도 상의 하나의 노드를 일반화 시키려고 할 경우 먼저 그 노드가 지닌 정지계수 값을 살펴보게 된다. 만약 일반화 시키고자 하는 노드의 정지계수 값이 0이 아닌 경우 서브트리의 일반화가 동일하게 진행되지 않았음을 의미함으로, 서브트리의 일반화가 동일하게 진행될 때 까지 즉 깊이가 깊은 서브트리에서 일반화를 시작한 속성 값의 일반화단계가 현재 노드까지 진행되는 동안 기다리게 된다. 이렇게 정지계수를 두는 이유는 [그림 2-b]의 예에서와 같이 서브트리의 깊이가 서로 다른 경우 깊이가 얕은 노드에서 일반화를 시작한 속성 값이 빠르게 상위 단계로 일반화 되어 가는 현상을 막기 위함이다. 그리고 일반화가 한번씩 진행될 때마다 정지계수를 1씩 감소시킴으로써 깊이가 가장 깊은 노드에서 일반화를 시작한 속성 값의 개념 계층도상의 현재 위치를 정지계수에 반영하게 된다.

[그림 3] 에 제시된 알고리즘을 이용하여 [그림 2]에 대한 클러스터링을 진행할 경우 [표 4]와 같은 클러스터링 결과가 생성된다. 이에서 알 수 있듯이 개선된 알고리즘을 이용 했을 경우 기존 알고리즘을 서브트리의 깊이가 다른 노드가 존재하는 개념 계층도에 적용할 경우 발생하는 과도한 일반화를 방지 할 수 있음을 알 수 있다.

표 4. 개선된 알고리즘을 통한 클러스터링 결과

Src-IP	Dest-IP	Count
ip1	MyCompanyIPs	30

5. 실험결과

제안된 알고리즘의 성능 평가를 위하여 [그림 4-a]와 같은 가상 네트워크에 대해 공격 발생기인 snot을 이용하여 가상 공격을 실시함으로써 얻어진 52,946 개의 경보에 대해 클러스터링을 실시하였다. 실험에서는 클러스터링을 위해 사용한 속성은 경보의 Source IP, Destination IP, Source Port, Destination Port의 네 가지 속성을 사용하였다. 그리고 사용한 개념 계층도는 IP의 경우 [그림 4-b]를 사용 하였고 Port의 경우[그림 4-c]를 사용하였다.

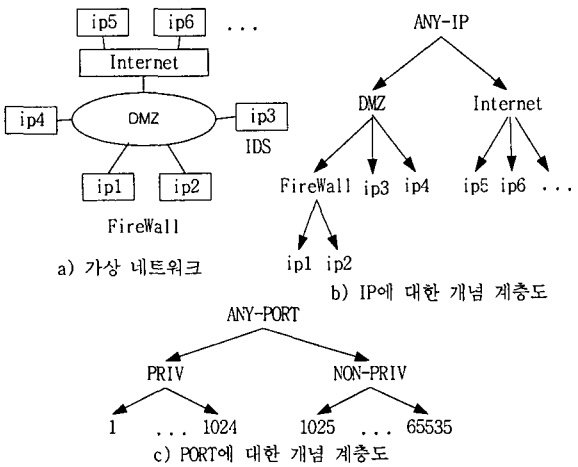


그림 4. 가상 네트워크와 개념 계층도

[표 5]와 [표 6]은 K. Julisch의 알고리즘을 이용했을 경우와 제안된 알고리즘을 이용했을 경우의 결과 비교를 나타낸다. 실험에서는 모든 공격이 외부에서 발생한 상황을 가정 하였으므로 Src-IP가 internet으로 표현된 것을 확인할 수 있다. 결과를 통해 알 수 있듯이 개념 계층도의 불균형이 클러스터링에 영향을 미치지 않는 처음 3개의 클러스터는 두 경우가 동일함을 알 수 있다. 반면 4번째 클러스터의 경우 기존 방식의 경우 *Dest-IP* 속성이 *ANY-IP*단계 까지 일반화 된 반면 제안된 방식의 경우 *DMZ* 단계에서 일반화가 멈춘 것을 확인 할 수 있다. 또한 클러스터내에 포함된 경보의 수가 개선된 방식이 더 적은 것을 확일 할 수 있는데 이는 기존 방식에 비해 불필요한 데이터가 더 적게 포함되었음을 의미한다. 즉 기존 방식의 경우 공격의 목표가 *Internet*인 경보들도 네 번째 클러스터에 포함되었기 때문에 개선된 방식에 비해 서로 연관성이 낮은 경보가 클러스터내에 많이 포함되었음을 알 수 있다.

표 5. K. Julisch의 알고리즘을 이용한 클러스터링 결과

Src-IP	Dest-IP	Src-Port	Dest-Port	Count
Internet	Internet	NON-PRIV	80	9673
Internet	FireWall	NON-PRIV	80	8315
Internet	DMZ	NON-PRIV	80	8235
Internet	ANY-IP	ANY-PORT	*undefined*	9414
Internet	ANY-IP	NON-PRIV	ANY-PORT	8107
Internet	ANY-IP	ANY-PORT	ANY-PORT	7856

표 6. 제안된 방식에 의한 클러스터링 결과

Src-IP	Dest-IP	Src-Port	Dest-Port	Count
Internet	Internet	NON-PRIV	80	9637
Internet	FireWall	NON-PRIV	80	8315
Internet	DMZ	NON-PRIV	80	8235
Internet	DMZ	ANY-PORT	*undefined*	7371
Internet	Internet	ANY-PORT	ANY-PORT	12545

6. 결론

본 연구에서는 기존에 제시된 AOI를 응용한 침입탐지시스템의 경보 클러스터링 기법에서의 과도한 일반화 문제를 개선하기 위한 방안을 제시 하였다. 실험 결과를 통해 알 수 있듯이 개선된 알고리즘이 기존 알고리즘에 비해 과도한 일반화 문제를 더 효율적으로 개선하였음을 알 수 있다. 실험에서는 비균형적 개념 계층도가 하나인 경우를 가정 하여 실험 하였지만 속성에 대한 개념 계층도의 불균형성이 증가하면 증가 할수록 개선된 알고리즘은 더 효율적일 것으로 판단된다. 하지만 현재의 알고리즘은 트리 형태의 개념 계층도만을 가정하고 있으므로 추후 그래프 형태의 비균형적 개념 계층도 또한 수용할 수 있도록 알고리즘의 개선이 필요하다.

7. 참고 문헌

[1] K. Julisch, Clustering intrusion detection alarms to support root cause analysis, ACM Transactions on Information and System Security, 6(4), pp. 443-471, 2002
[2] S. Axelsson, The Base-Rate Fallacy and the Difficulty of Intrusion Detection, *ACM Transactions on* Information and System Security, 3(3), pp. 186-205, 2000
[3] J. Han, Y. Cai, Data-Driven Discovery of Quantitative Rules in Relational Databases. IEEE Transactions on Knowledge and Data Engineering , 5(1), pp. 29-40, 1993