한국산학기술학회논문지
Vol. 11, No. 11 pp. 4553-4562, 2010

비정상행위 탐지 알고리즘 구현 및 성능 최적화 방안

신대철[1], 김홍윤[1*]
[1]한서대학교 전자컴퓨터통신공학부

Implementation of abnormal behavior detection Algorithm and Optimizing the performance of Algorithm

Dae-Cheol Shin[1] and Hong-yoon Kim[1*]
[1]**Division of Eletronics and Computer, Hanseo University**

요 약 네트워크의 발달과 더불어 보안에 대한 중요성이 부각되면서 많은 침입탐지시스템이 개발되고 있다. 침입에 대한 다양한 침투기법을 미리 파악하여 패턴화시킴으로써 침입을 탐지하는 오용행위탐지와 알려진 침입뿐만 아니라 알려지지 않은 침입이나 비정상행위 탐지를 위한 비정상행위탐지 등이 그것이다. 현재 비정상행위탐지를 위한 통계적 방법 및 비정상적인 행위의 추출과 예측 가능한 패턴 생성을 위한 다양한 알고리즘 등이 연구되고 있다. 본 연구에서는 데이터 마이닝의 클러스터링 및 연관규칙을 사용하여 두 모델에 따른 탐지영역을 분석하여 대규모 네트워크에서의 침입탐지 시스템을 설계하는데 도움을 주고자 한다.

Abstract With developing networks, information security is going to be important and therefore lots of intrusion detection system has been developed. Intrusion detection system has abilities to detect abnormal behavior and unknown intrusions also it can detect intrusions by using patterns studied from various penetration methods. Various algorithms are studying now such as the statistical method for detecting abnormal behavior, extracting abnormal behavior, and developing patterns that can be expected. Etc. This study using clustering of data mining and association rule analyzes detecting areas based on two models and helps design detection system which detecting abnormal behavior, unknown attack, misuse attack in a large network.

Key Words : Abnormal behavior, Algorithm, Clustering, Association

1. 서론

최근 네트워크로 연결된 시스템의 가용성, 기밀성, 무결성 등을 해치는 침입을 탐지하기 위해 많은 연구가 진행 중에 있다[1-14]. 알려진 공격에 대한 패턴 일치 여부를 통해 침입여부를 판정하는 오용행위탐지시스템과 정상행위 프로파일링을 통한 알려진 침입뿐만 아니라 알려지지 않은 침입과 비정상행위에 대해 판정을 하는 비정상행위탐지시스템 등이 있다. 이러한 침입탐지시스템을 개발하는데 있어서 오용행위탐지시스템은 패턴의 유/무에 따라 시스템의 성능을 좌우할 뿐이지만 비정상행위탐지시스템의 경우 어떠한 모델을 사용하고 판정기준이 되는 판정요소를 어떻게 선택하며 그 수는 과연 얼마나 할 것인가? 하는 등의 전반적인 고려사항을 토대로 시스템의 성능을 좌우 할 수 있다. 본 연구에서는 비정상행위탐지시스템을 개발하는데 이용하는 통계적 방법, 베이지안 통계학, HMM(Hidden Markerv Model), 신경망(Neural Network), 데이터 마이닝(Data Mining) 중 데이터 마이닝 기법의 클러스터링과 연관규칙을 이용하여 알고리즘 특성에 따른 시스템의 탐지영역을 분석해봄으로써 비정상행위탐지시스템 설계에 있어서의 고려요소와 나아가 오용행위와 비정상행위의 탐지영역을 분석해 보고자 한다.

먼저, 2장에서는 네트워크기반침입탐지시스템 개발에 있어서 데이터 마이닝에 대한 기존의 연구가 어떻게 진행되고 있는지에 대하여 알아보고 3장에서는 기존의 연구를 바탕으로 클러스터링과 연관규칙에 대한 실질적인

본 논문은 한서대학교 전자컴퓨터통신공학부 연구과제로 수행되었음.
*교신저자 : 김홍윤(hykim@hanseo.ac.kr)
접수일 10년 10월 07일 수정일 10년 10월 30일 게재확정일 10년 11월 19일

방법론에 대해 서술하고 4장에서는 본 연구에서 개발한 두 비정상행위탐지모델과 오용행위에 대한 탐지영역 분석을 위한 실험을 통해 5장에서는 결론 및 추후 연구 방향에 대해서 서술한다.

2. 관련연구 동향

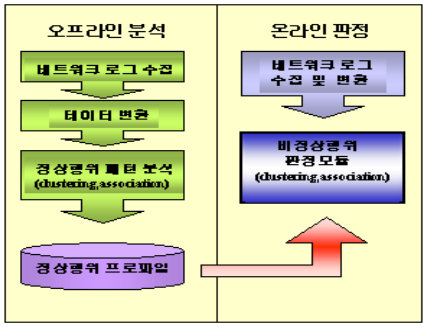
데이터 마이닝은 대용량의 실제 데이터로부터 묵시적(implicit)이고, 미리 알려져 있지는 않지만(previously unknown) 잠재적으로 유용한 정보를 발견하는 작업으로 정의된다. 즉, 불확실한 데이터나 잘못 획득된 데이터까지 포함하는 데이터로부터 예상하지 않은 새로운 정보를 추출하는 것을 의미하며, 또한 이러한 데이터를 활용하여 의사결정, 성능향상과 같은 목적으로 명시적(explicit)이 아닌 데이터로부터 행위에 대한 유용한 특성패턴을 발견하는 것이다[20-29]. 데이터 마이닝은 결과에 대한 유용성과 불확실성을 정량화 할 수 있어야 하며, 수행 결과는 수많은 패턴 및 새로운 정보를 얻게 되는데 이로부터 유용한 패턴을 식별할 수 있는 척도와 예측을 위해 얻어진 결과는 통계적 불확실성이 반드시 포함되므로 불확실성에 대한 척도가 필요하다. 데이터 마이닝을 사용한 예로는 JAM(Java Agent for Meta-Learning over Distributed Database)이 있다[20]. 데이터 마이닝은 탐사하고자 하는 데이터의 형태에 따라 분류할 수 있는데, 클러스터링(data clustering), 분류규칙(classification rule), 연관규칙(association rule)등의 주요한 방법을 나열 할 수 있다. 네트워크상의 시간을 포함하는 순차적 사건에 대하여 데이터 마이닝의 연속패턴에 의한 연관규칙의 방법을 이용해서 네트워크 특성을 분류하고 규칙을 생성한다. 예를 들면, 네트워크 행위 N에 대한 임의시간에 발생한 사건 S1이 존재한다면, 동일한 네트워크 사용자에 의해 이후 시간에 발생하는 사건 S2가 존재하는 새로운 정보를 추출할 수 있다.

즉, S1=>S2, 단, S1과 S2는 서로소(disjoint)이다. 이러한 연관규칙 방법은 "S1에 속한 사건의 발생은 동시 혹은 일정한 시간 간격 사이에 S2에 속한 사건 항목들이 발생한다."는 해석을 할 수 있다. 분류규칙은 몇 개의 중복되지 않은 그룹으로 분류했을 때, 각 그룹의 특성을 대표할 수 있는 특정 패턴을 추출하여 아직 알려지지 않은 사건들에 대한 예측을 하는 것을 목적으로 한다.

클러스터링 방법은 발생빈도가 적은 데이터가 주기적으로 발생되었다면 중요한 의미를 가지고 있을 수 있지만 기존의 방법에서는 서로 관련이 없는 데이터를 하나의 단위로 처리함으로써 의미 있는 데이터의 손실과 비정상행위에 대한 부정확한 판단을 할 수 있다. 따라서 네트워크 이용에 관련한 행위를 지지율 등의 판정요소에 의한 클러스터링 방법을 이용하여 네트워크상의 정상행위 및 비정상행위를 탐지하도록 한다.

3. 구현

네트워크기반침입탐지시스템은 오프라인 분석단계와 온라인 판정단계로 나눌 수 있다. 오프라인 분석단계에서는 네트워크상에서 전송되는 데이터를 수집/분석하여 재구성함으로써 네트워크상의 패킷 정보를 추출하고, 클러스터링과 연관규칙을 이용하여 고유한 정상행위 패턴을 찾아서 최적화 된 프로파일을 생성한다. 온라인 판정 단계에서는 오프라인 단계에서 학습된 정상행위를 근거로 실시간으로 비정상행위를 판정하는 단계이다. 그림 1은 네트워크기반침입탐지시스템의 전체구조이다.



[그림 1] 전체구조

3.1 설계

네트워크기반 침입탐지시스템을 설계하는데 있어서 가장 우선적으로 해야 할 것은 네트워크 행위에 있어서의 방대한 데이터로부터 반복적인 행위에 대해서 대표가 될 수 있는 정상행위를 추출하기 위한 판정요소를 선정하는 것이다. 판정요소 선택의 적합여부에 따라 시스템의 성능에 많은 영향을 끼칠 수 있기 때문이다.

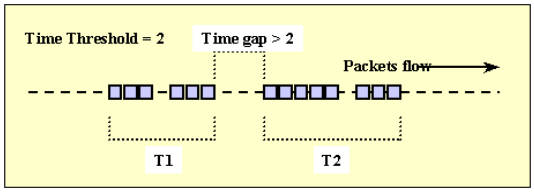
3.1.1 신뢰도

판정요소를 선택하는데 있어서 가장 일반적인 선택은 네트워크 데이터로부터의 IP헤더 정보라고 볼 수 있다. IP헤더로부터 얻을 수 있는 정보로는 duration, protocol type, data length, flag, source/destination address, port등이다[18,19,20,25,29]. 성능과 판정요소의 관계는 판정요소가 너무 많으면 온라인 판정시 네트워크 처리 속도가

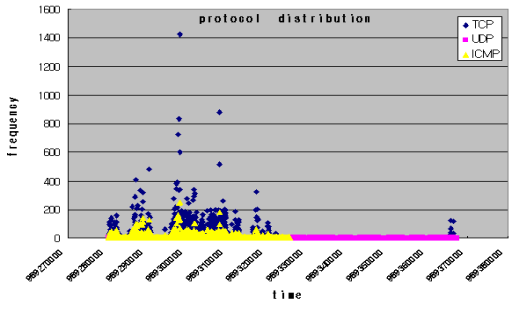
저하 될 수 있을 뿐만 아니라 정상행위를 학습하는데 많은 시간을 낭비 할 수 있으며 판정요소가 너무 적으면 정상행위에 대한 묘사성이 떨어질 수 있다. 이러한 판정요소 선택은 성능면에서 아주 중요한 요소로 많은 연구가 필요하다고 할 수 있다.

3.1.2 판정요소 분석단위

네트워크 행위로부터 판정요소를 추출하기 위해서는 일정단위의 트랜잭션이 필요하다. 트랜잭션은 분석단위 및 판정단위가 되는 조건으로 기존의 통계적인 방법에서는 사용자의 행위에 관계없이 일정시간 간격을 가지는 고정길이 트랜잭션을 사용하였다[16]. 본 연구에서는 이러한 사용자 대신 네트워크에서 일어나는 행위에 대해서 가변길이 트랜잭션을 기준으로 삼았다. 일반적으로 네트워크상의 행위 트랜잭션은 수시로 일어나기 때문에 의미있는 단위의 트랜잭션을 만들기 위해서는 가변길이 트랜잭션이 적합하다고 볼 수 있다. 또한 고정길이 트랜잭션을 사용하는 경우 탐지시간에 있어 고정길이 트랜잭션 크기만큼의 탐지 시간이 지연되는 반면 가변 길이 트랜잭션을 사용하는 경우 실시간 탐지가 가능하다는 장점을 가지고 있다. 그림 2는 특정 IP로부터 n개의 패킷이 전송되어 올 때 패킷 간의 시간 간격이 주어진 임계치(Threshold)를 넘지 않는 다면 n개의 패킷은 하나의 트랜잭션으로 묶일 수 있음을 보여주고 있다. 그림 3은 네트워크에서 전송되는 데이터를 받아 트랜잭션 내의 정보를 시간의 흐름에 따라 프로토콜별로 나타나는 분포와 빈도를 나타내주는 그래프이다.



[그림 2] 가변길이 트랜잭션



[그림 3] 판정요소 분포 및 빈도

3.1.3 클러스터링

대용량의 사건들이 기록되어 있는 데이터베이스에서 유사 작업군 탐색 기법을 클러스터링[21-24]이라고 한다. 작업 단위가 트랜잭션 단위이기 때문에 침입탐지 환경에서는 트랜잭션 정보가 네트워크 정상행위 패턴 생성에 있어서 상당히 중요하다. 기존의 데이터 마이닝 관련 연구[20,21,25-27]에서와 마찬가지로 본 연구에서도 지지율을 이용하여 사용자의 네트워크 행위에 대한 정상행위 패턴을 생성하도록 하였다. 그 결과 침입 탐지 환경에서 데이터를 보다 더 정확하고 효과적으로 모델링 할 수 있었다. 일반적으로 정상행위 패턴을 생성하기 위해서는 일정 기간 내에 존재하는 트랜잭션들을 분석해야 하며, 각 트랜잭션에는 다양한 행위가 포함될 수 있으며 클러스터링을 수행하기 위해 이러한 행위들은 판정 요소에 의해서 수치 값으로 표현된다. 즉, i번째 트랜잭션에서 j번째 행위를 $a_{i,j}$라 하면, 판정요소 m_k에 대한 행위 값은 $m_k(a_{i,j})$와 같이 표현될 수 있다. 또한, 전체 트랜잭션에서 판정요소 m_k에 대한 데이터의 집합을 M_k와 같이 표현될 수 있으며 이를 이용하여 실질적인 클러스터링이 수행된다. 집합 M_k에 대한 클러스터링 수행 시 클러스터 분류를 위해서 사용자의 특정 행위 $a_{s,t}$에 유사한 행위 집합 ($N_\lambda(a_{i,j})$)을 다음과 같이 표현될 수 있다.

$$N_\lambda(a_{i,j}) = \{ x \mid m_k(a_{i,j}) - \lambda \leq x \leq m_k(a_{i,j}) + \lambda, x \in M_k \}$$

※ λ : 클러스터링 범위

여기에서 M_k내에서 인접한 두 행위를 $a_{s,t}$와 $a_{s,t+1}$이라 했을 때, 두 행위가 동일한 클러스터에 포함되기 위한 조건은 다음과 같다.

① $N_\lambda(a_{s,t}) \cap N_\lambda(a_{s,t+1}) \neq \varnothing$

② $\sup(N_\lambda(a_{s,t}))$, $\sup(N_\lambda(a_{s,t+1})) \geq$ min_support

※ $\sup(N_\lambda)$: N_λ의 지지율

조건①은 클러스터링 범위에 대해서 두 집합 $N_\lambda(a_{s,t})$와 $N_\lambda(a_{s,t+1})$에서 가장 인접한 두 행위 값 간의 차가 λ이하임을 나타낸다. 즉, 인접한 두 행위 값이 클러스터링 범위 내에 존재하게 되므로 같은 클러스터로 묶일 수 있으며, 그렇지 않으면 서로 다른 클러스터로 분리

된다. 조건②에서는 두 집합 $N_\lambda(a_{s,t})$와 $N_\lambda(a_{s,t+1})$가 클러스터링 수행을 위해서 주어진 최소 지지율(min-support)을 만족하는지의 여부를 판별하게 된다. 여기에서 N_λ의 지지율을 계산하기 위해서 N_λ^i를 i번째 트랜잭션 Ti에 포함되는 행위 집합이라 정의할 수 있다. 따라서 sup(N_λ)은 다음과 같이 계산된다.

$$\text{sup}(N_\lambda) = \sum_{i=1}^{n} I(T_i, N^{i_\lambda})$$

I(T_i, N_λ^i) : if T_i, N_λ^i then 1, otherwise 0

```
Find_Normal_Activity(D(Measure ID), MinSupport, )

Nλ (a): 행위 값 a의 클러스터링 반경 내에 존재하는 데이타 집합

Sort by UserID as major key and MeasureID as minor key;
for all data a in D(Measure ID) {
   if (a is unclassified) {

      retrieve Nλ (a);

      if (sup(Nλ (a)) < MinSupport) set noise to a and return;

      else{set new cluster-id to all data in Nλ (a);

            push all data from Nλ (a) onto stack;
            while(not stack.empty()) {
               current = stack.pop();

               retrieve Nλ (current);

               if (sup(Nλ (current)) >= MinSupport) {

select all data unclassified/marked as noise  in Nλ (current);
                     stack.top = -1
           set current cluster-id to these data;
           push the unclassified data onto stack; }
               }
       }
   }
```

[그림 4] 클러스터링 알고리즘 구조

클러스터링 알고리즘은 그림 4와 같다. 알고리즘 FNA는 판정요소(Measure_ID), 최소 지지율(Min_Support), 클러스터링 범위(λ)가 주어졌을 때 유사 작업군을 탐색하게 된다. 탐색된 작업군이 주어진 최소 지지율 이상의 트랜잭션들을 포함하고 있다면 클러스터로 생성되고 그렇지 않을 경우에는 잡음(noise)으로 처리되어 클러스터에 포함시키지 않는다. 알고리즘 FNA의 상세한 수행 과정은 다음과 같다.

[단계1] 클러스터링을 위해서 주어진 데이터를 오름차순으로 정렬한다.

[단계2] 클러스터링은 정렬된 데이터의 맨 처음 데이터부터 시작하며, 데이터의 유효 반경 안에 존재하는 데이터들의 집합 N_λ을 구한다.

[단계3] N_λ내에 존재하는 모든 데이터들에 대해서 지지율을 구하고 이를 최소 지지율과 비교한다. 이때 최소 지지율보다 작으면 이 데이터의 상태는 Noise로 설정된다. 만일 최소 지지율 이상이면 이 데이터에 새로운 클러스터 ID를 부여하고 N_λ에 포함되어 있는 데이터들을 스택(stack)에 넣고 스택이 빌 때까지 다음 단계를 수행한다. 만일 스택이 비어있으면 2단계를 다시 수행한다.

[단계4] 스택의 맨 상단으로부터 데이터를 읽어들여 이 데이터에 대한 N_λ을 구한다. 만일N_λ의 지지율이 최소 지지율보다 크면 N_λ내에 존재하는 unclassified /noise 데이터에 현재의 클러스터 ID를 부여한다.

[단계5] 스택을 초기화 한다. N_λ내에서 이전에 잡음(noise)이나 클러스터 ID가 부여되지 않은 새로운 데이터들을 스택에 넣는다. 이 과정을 모든 데이터가 접근이 될 때까지 반복한다.

최소 지지율이 지지율에 영향을 주는 반면 클러스터링의 범위는 실질적으로 유사한 작업 그룹을 묶는 정도에 영향을 준다. 만일 클러스터링 범위가 너무 좁게 설정되면 클러스터가 만들어지지 않을 수 있으며 너무 크게 설정되면 생성된 클러스터의 정확도가 떨어지게 된다. 따라서 최적의 클러스터링 범위를 설정하는 것은 생성될 클러스터의 개수와 클러스터의 정확도에 상당히 많은 영향을 미치게 된다. 이를 위해 실험을 통해서 최적의 클러스터링 범위를 찾도록 하였다.

3.1.4 연관규칙

대용량의 사건들이 기록되어 있는 데이터베이스에서 자주 발생되는 item간의 상호 연관성 탐색 기법을 연관규칙이라고 한다[28]. 연관규칙을 이용하여 네트워크의 정상적인 행위 패턴을 추출함으로써 새로운 행동에 대한 비정상행위도를 파악할 수 있다. 하지만 기존의 연관 규칙을 그대로 이용하였을 때 네트워크 행위에 대해서 정확하게 모델링할 수 없는 단점을 가지고 있다. 즉, 네트워크상으로부터 수신되는 패킷들은 의미관계를 가지는 하나의 트랜잭션으로 묶을 수 있다. 따라서, 정상행위 패턴을 생성하기 위해서는 패킷 내 뿐만 아니라 패킷 간의 연관성을 탐사하는 알고리즘이 필요하다.

패킷 내 및 패킷 간의 연관성을 탐사하는 연관규칙 알고리즘은 다음과 같이 크게 네 가지 과정으로 나누어진다.

[1단계] 기존의 Apriori 알고리즘을 이용하여 패킷내의 연관성을 탐사한다.

[2단계] 탐사된 연관 패턴의 ID를 로그에 매핑시켜서 새로운 로그를 만든다.

[3단계] 매핑된 로그를 다시 변형된 Apriori알고리즘을 이용 패킷간의 연관성을 탐사한다.

[4단계] 생성된 패턴간의 포함관계가 존재하지 않는 최대 연관 집합을 구한다.

위의 4가지 단계를 고려하여 네트워크 로그 데이터로부터 연관성을 찾는 과정은 그림 5와 같다.

```
[단계1] get unique item set from log;
        sort unique item set;
        get large 1-item set;
          while there is no more large itemset
begin
            generate candidate n-itemset;
            prune nonnecessary itemset;
            generate large n-itemsets;
           end
[단계2] map itemset id to log;

[단계3] while there is no more large association
begin
        generate candidate n-association;
        prune nonnecessary association;
        generate large n-association;
        end
[4단계] generate maximal associations;
```

[그림 5] 연관규칙 알고리즘

[1단계]에서 Apriori알고리즘을 이용하여 패킷내에 존재하는 item들간에 연관성을 탐사한다. 먼저 로그로부터 유일한 item 집합을 탐색한다. 탐색된 item 집합은 오름차순으로 정렬하고 각 item에 대해서 빈발 1-item집합을 구한다. 빈발 1-item집합으로부터 item의 길이가 2인 후보 2-item집합을 생성하게 되며 후보 2-item집합을 이용하여 빈발 2-item집합을 구하게 된다. 이러한 과정은 새로운 빈발 item집합이 탐사되지 않을 때까지 반복된다.

[2단계]에서는 [1단계]에서 생성된 빈발 item집합을 로그에 적용하여 패킷 간 연관성을 찾기 위한 새로운 로그를 생성하게 된다. 이때 생성된 빈발 item집합에 각각 유일한 식별자를 부여하게 된다. 이것은 [3단계]에서 빈발 association을 구하기 위한 item으로 이용된다. 한편, 빈발 association을 구하기 위한 로그를 생성하기 위해서 각 로그에 정합되는 빈발 item집합의 식별자를 새로운 로그에 추가된다. 이 과정에서 빈발 1-association을 생성한다.

[3단계]에서는 두 번째 단계에서 생성된 로그를 이용하여 빈발 association을 구하게 된다. 이때 사용되는 알고리즘은 빈발 item 집합에서와 마찬가지로 Apriori 알고리즘을 이용한다.

[4단계]에서는 생성된 빈발 association에 대해서 서로간에 포함관계가 없는 최대 빈발 association 집합을 구하게 된다. 최대 빈발 association 집합을 구하기 위해서, 먼저 [3단계]에서 생성된 association 집합을 패턴 크기의 내림차순으로 정렬한다. 정렬된 association 집합에서 맨 첫 번째 association을 maximal-set에 입력하고 나머지 association에 대해서 maximal-set에 포함되면 삭제하고, 그렇지 않으면 maximal-set에 입력한다.

3.2 판정

3.2.1 클러스터링

온라인 트랜잭션에 대한 비정상 행위 판정은 정상행위 프로파일을 이용하여 온라인에서 발생되는 사용자의 행위정도에 대한 차이가 높을 경우 비정상 행위도가 높아지게 된다. 클러스터링을 이용한 정상행위 프로파일에는 클러스터링 결과로 생성된 각 판정 요소의 클러스터 리스트를 유지한다. 정상행위 프로파일에 포함되는 클러스터는 판정요소(M), 클러스터 평균값(C), 클러스터 분산(V), 클러스터 지지율(S)로 구성된다. 판정요소는 비정상 행위도에 대한 판정에 참여하는 항목(Category)이며, 클러스터 평균값은 클러스터 내에 존재하는 데이터들의 평균값이다. 클러스터 분산은 클러스터 내에 존재하는 값들의 분포를 나타내며, 클러스터 지지율은 클러스터 내에 존재하는 트랜잭션의 수이다.

하나의 판정요소에 대해서, 클러스터가 두 개 이상 생성됐을 때 온라인 데이터의 비정상행위도를 판정할 경우 어떤 클러스터와 비정상 행위를 판정할 것인지를 결정해야 한다. 이를 위해서는 [정의]와 같이 두 클러스터 사이의 중심점을 구하여 온라인 데이터와 가까운 위치에 존재하는 클러스터와 비정상행위도를 계산하게 된다.

[정의] 두 클러스터 사이의 중심점

클러스터 사이의 중심점은 두 클러스터의 정규 분포함수가 교차하는 지점으로 정의할 수 있다. 즉, 클러스터 A와 B의 평균과 분산이 각각 μ_A, δ_A^2, μ_B, δ_B^2 와 같다면 클러스터 사이의 중심점(x)은 근의 공식을 이용하여 다음과 같이 구할 수 있다. 이 때 x는 두 클러스터의 평균 μ_A, μ_B사이에 존재해야 한다. 만일 클러스터 사이의 중심점 x가 두 클러스터의 평균 μ_A, μ_B사이에 존재하지

않는다면 온라인 판정시 두 클러스터 모두와 판정을 수행하도록 한다.

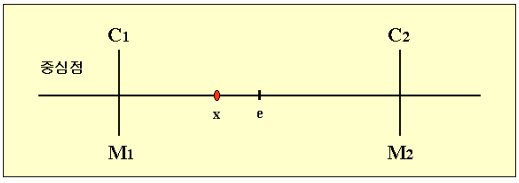
$f_A(X) = f_B(X)$ (f_A와 f_B는 정규 분포 함수)

$$\frac{1}{\delta_A\sqrt{2\pi}}e^{-\frac{1}{2}\frac{(x-\mu_A)^2}{\delta_A^2}} = \frac{1}{\delta_B\sqrt{2\pi}}e^{-\frac{1}{2}\frac{(x-\mu_B)^2}{\delta_B^2}}$$

※ $X = \frac{-b \pm \sqrt{b^2-4ac}}{2a}$

※ $a = (\delta_B^2 - \delta_A^2)$, $b = -2*(\delta_B^2\mu_A - \delta_A^2\mu_B)$,

$c = \delta_B^2\mu_A - \delta_A^2\mu_B - 2*\delta_A^2*\delta_B^2*\log\frac{\delta_A}{\delta_B}$



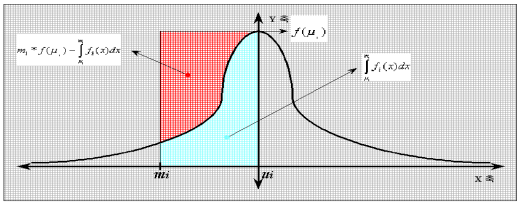
X > e : C2 와 비교
X < e : C1 과 비교
X = e : C1 또는 C2 선택

[그림 6] 클러스터 사이의 중심점

그림 6에서 보는 바와 같이 상대적인 중심점 x가 절대적인 중심점 e보다 큰 경우 클러스터 C2와 비교를 하고 e보다 작은 경우 C1과 비교를 한다. 만약, x가 e와 같은 경우는 두 클러스터 C1 또는 C2와 비정상행위도를 비교한다. 한편, 온라인 트랜잭션 T = { m_1, m_2 ..., m_t}에 대한 판정을 수행하기 위해서 트랜잭션 T의 각 판정요소에 대한 값들과 정합되는 클러스터 집합(CS)을 [정의]를 이용하여 다음과 같이 구해야 한다.

CS = {(C_1, support(C_1)), (C_2, support(C_2))..(C_t, support(C_t))}

여기에서 m $i \in$ T 에 대한 판정율은 그림 7에서 보는 바와 같이 다음과 같은 수식으로 계산 된다.



[그림 7] 클러스터의 정규분포

$$\alpha\left(m_i f(\mu_i) - \int_{\mu_i}^{m_i} f_i(x)dx\right) + \beta*\text{support}(C_i)$$

where m $i \in$ T and $C_i \in$ CS and $\alpha + \beta = 1$

따라서 전체 판정 요소에 대한 판정율은 다음과 같이 계산된다.

$$\text{Abnormality} = \frac{1}{t}*\sum_{i=1}^{t}\ \alpha*\left(m_i*f(\mu_i) - \int_{\mu_i}^{m_i} f_i(x)dx\right) + \beta*\text{support}(C_i),$$

where $m_i \in T$ and $C_i \in CS$ and $\alpha + \beta = 1$

3.2.2 연관 규칙

연관 규칙을 이용한 정상행위 프로파일은 다음과 같이 maximal association-set(MAS)과 element support-set(ESS)으로 구성된다.

MAS = { (R_1, support(R_1)), ..., (R_n, support(R_n)) }

R_i= { (a_1^i, support(a_1^i)),(a_2^i, support(a_2^i)), ..., (a_l^i, support(a_l^i)) }

ESS = { (e_1, support(e_1)), (e_2, support(e_2)), ..., (e_m, support(e_m)) }

MAS는 연관 규칙 탐사 알고리즘에서 생성된 규칙의 집합을 의미한다. MAS는 여러 개의 규칙과 이에 대한 지지율을 포함한다. MAS의 구성 요소인 규칙은 여러 개의 패턴 내 연관 규칙과 이에 대한 지지율을 포함하고 있다. ESS는 연관 규칙 생성단계 중 첫 번째 단계에서 크기가 1인 item 집합의 원소와 지지율을 포함한다. 여기에서 ESS에 포함되는 원소는 빈발 item뿐만 아니라 저빈도의 item 집합도 함께 포함된다. MAS와 ESS를 이용하여 온라인 트랜잭션 T = p1, p2, ..., pt에 대한 비정상 행위를 구하는 과정은 다음과 같다. 먼저 정상행위 프로파일을 이용한 최대 정상행위도(MNA)는 다음과 같이 계산된다.

$$MNA = \alpha*\frac{1}{n}\sum_{i=1}^{n}\sup(R_i) + \beta*\frac{1}{n*l}\sum\sum\sup(a_j^i)*|a_j^i|,$$

where $R_i \in MAS$, $a_j^i \in R_i$ and $\alpha + \beta = 1$,

이 식에서는 규칙에 대한 지지율 평균과 규칙의 구성요소인 item의 평균을 구하여 이 두 수치에 대한 가중치 평균을 구하였다. 이와 같이 계산된 MNA는 온라인 트랜잭션에 대한 판정률을 정규화하기 위한 값이다. 한편, 온

라인 트랜잭션 T가 주어졌을 때, 정상행위 패턴과의 정합된 정도(RNA)는 다음과 같이 계산된다.

$$RNA = \frac{1}{N_c} * \sum_{i=1, R_i \subseteq T}^{n} \sup(R_i) \quad \text{where } R_i \subseteq T \text{ and } R_i \subseteq MAS$$

$$N_c = \sum_{i=1}^{n} I(R_i, T)$$

$$I(R_i, T) \quad : \text{if } Ri \subseteq T \text{ then } 1, \text{ otherwise } 0$$

이 수식은 트랜잭션 T가 포함하는 정상행위 패턴들의 평균을 나타낸다. 또한 트랜잭션 T의 원소가 연관 규칙의 각 원소와 정합된 정도(ANA)를 다음과 같이 계산될 수 있다.

$$ANA = \frac{1}{n*l*t} \sum_{i=1}^{n} \sum_{j=1}^{l} \sum_{k=1}^{t} | p_k \cap a_j^i | * a\sup$$

$$asup \; : \; support(a_j^i) \; if \; a_j^i \subseteq p_k$$

$$support(e) \; else \; if \; \exists e \in ESS \; and \; e \in p_k \; \; 0 \; otherwise$$

이 수식에서 트랜잭션 T에 포함되는 원소 p_k와 하나의 규칙에 포함되는 원소 a_j^i 사이의 정합율은 $|p_k \cap a_j^i|*a\sup$와 계산된다. 여기에서, 만일 a_j^i가 p_k에 포함되면 asup는 support(a_j^i) 값을 취한다. 만일 a_j^i 가 p_k에 포함되지 않고 ESS의 어떤 원소 e가 p_k에 포함되면 asup은 support(e)의 값을 취한다. 위의 두 가지 경우가 아닌 경우의 asup 값은 0이 된다. 위의 식에서 온라인에서의 네트워크 행위 트랜잭션(T)이 정상행위 패턴(R)과 유사하다면 값이 커지게 됨으로써 비정상행위도가 낮게 나타나게 된다. 반면, 사용자가 이상 행위를 하였을 경우에는 정상행위 패턴과 유사하지 않기 때문에 비정상행위도가 높게 나타나게 된다. 최종적으로 온라인 트랜잭션 T에 대한 비정상행위도는 다음과 같이 계산될 수 있다. 이 수식에서도 MNA에서처럼 가중치를 RNA과 ANA사이에서 가중치 합을 하였고, 이 수치를 정규화하기 위해서 MNA으로 나누었다.

$$\text{Abnormality} = 1 - \frac{\alpha * RNA + \beta * ANA}{MNA}$$

where $\alpha + \beta = 1$

4. 실 험

4.1 데이터 생성 방법

- 학습데이터 : 테스트베드 상의 순수한 네트워크 정상행위 로그데이터(본 연구에서는 10일 동안의 로그데이터 사용)
- 테스트데이터 : 오용침입데이터(표 1)와 네트워크비정상행위 데이터(표 2)

[표 1] Misuse list

Attack	Protocol	Description
Synsol	TCP	Syn - Flooding
Pscan	ICMP	Icmp-Flooding
Solaris_land	TCP	Syn - Flooding
Newpep	UDP	Udp - Flooding

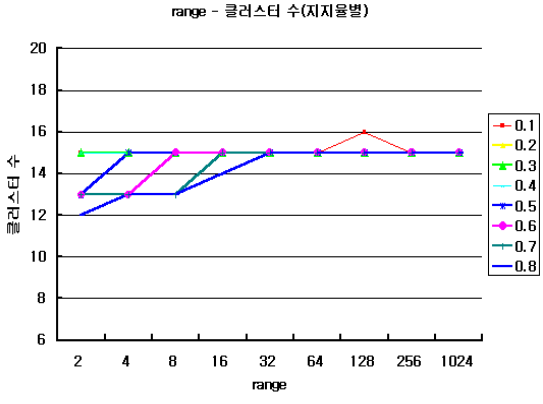
[표 2] Anomaly list

Attack	Method	Description
Anomaly1	TELNET, FTP	평소보다 많은 다량의 서비스를 발생
Anomaly2	SMTP,POP3, PING	평소 사용하지 않았던 서비스를 발생
Anomaly3	MainController	낯선 호스트로부터의 로그인을 발생
Anomaly4	Port 9999	잘 사용하지 않는 포트에 접속
Anomaly5	TELNET, FTP등	근무시간 외, 트래픽 발생

4.2 학습데이터 수집 방법

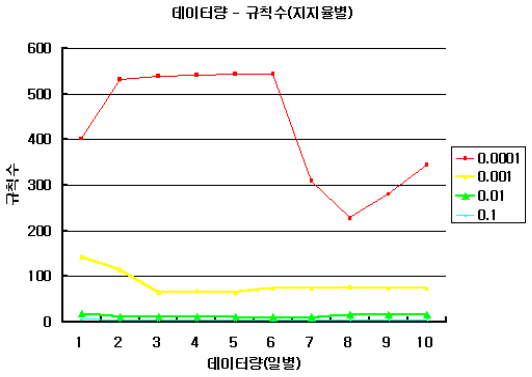
학습데이터는 비정상행위탐지시스템의 정상행위 프로파일을 위한 데이터로써 테스트베드 상에서 순수한 정상행위만을 발생하여 로그를 수집하였다. 수집된 로그는 수차례의 최적화 과정을 거쳐 정상행위를 가장 잘 묘사할 수 있는 대표적인 프로파일을 가지도록 하였다.

4.3 학습과정 및 최적화



[그림 8] 클러스터의 최적화

정상행위에 대한 최적의 학습을 하기위해서는 매개변수(range, min_support)에 대한 실험치가 있어야 한다. 그림 8은 최적화를 위한 매개변수에 따른 클러스터링 개수이다. min_support가 0.1이고 클러스터링 범위가 128일 때 가장 많은 클러스터링 개수를 얻을 수 있었고 이러한 최적화 과정을 통해 비정상행위에 대한 민감도를 높일 수 있었다.

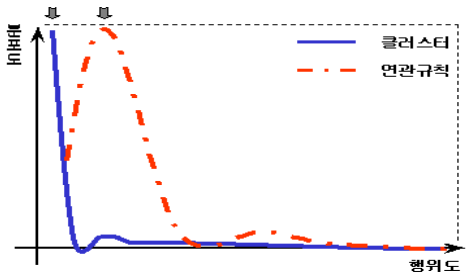


[그림 9] 연관규칙의 최적화

그림 9는 데이터량에 따른 연관규칙 수에 대한 최적화 과정을 보여주고 있다. min_support가 0.0001이고 6일정도의 데이터량이 있을 때 가장 많은 연관규칙을 생성함을 알 수 있었고 이때 정상행위에 대한 묘사성이 가장 높다고 할 수 있다. 이러한 최적화 과정을 거친 비정상행위 탐지시스템에 대해 실질적인 오용행위와 비정상행위를 발생시킴으로써 탐지영역에 대한 분석을 할 수 있었다.

4.4 프로파일링

최적화 과정을 거친 후 정상행위에 대한 프로파일링의 적절성 여부를 테스트하기 위해 학습 대상이 되었던 정상행위로그데이터에 대한 오프라인테스트를 하였다. 이러한 탐지결과는 정상행위에 대한 판정결과로써 침입이 포함되어 있지 않아야 한다. 탐지결과를 구간별로 나누어 분석해 본 결과, 일반적으로 그림 10과 같은 그래프 형태를 나타내고 있음을 알 수 있다. 정상행위에 대한 비정상행위탐지시스템의 판정 분포도는 정상행위가 수행되었을 때 비정상행위탐지시스템에서 어떻게 반응할 것인지를 미리 추정해 볼 수 있다. 이러한 추정값은 정상행위와 비정상행위 사이의 구분을 정확히 할 수 있으며 오탐지를 줄이는 효과를 발휘할 수 있다.



[그림 10] 정상행위에 대한 임계치 결정구간

그림 10에서 X축은 행위도를 나타내고 Y축은 행위도에 대한 분포도를 나타내고 있다. 행위도에서 오른쪽으로 이동할수록 비정상행위도가 높은 경우이며 분포도에서 위쪽으로 이동할수록 빈도수가 증가 한다고 볼 수 있다. 그림에서 보면 클러스터링 및 연관규칙에서 각각 2번의 군집현상이 보이는데 화살표가 지시하고 있는1차 군집에 해당하는 부분은 정확히 판단된 정상행위이고, 2차 군집에 해당하는 부분은 학습상의 오차나 알고리즘 상의 오차에 의하여 나타나는 현상이다. 2차 군집 부분은 정상행위일 수도 있고 비정상행위일 수도 있는 애매한 부분이다. 본 연구에서는 1차 군집에 있어서 적정값을 선택하여 임계치를 결정하였다.

4.5 결과

본 실험에서는 오프라인테스트를 통해 나온 임계치에 따라 탐지여부를 결정하였다. 클러스터링인 경우 평균적으로 정상행위도가 5에 수렴하는 것을 알 수 있었고 연관규칙인 경우 40에 수렴함을 알 수 있었다. 이러한 수렴여부에 따라 각 모델에 있어서 임계치를 5와 40으로 결정하였다.

	synsol	pscan	solaris_land	nestep	anomal1	anomal2	anomal3	anomal4	anomal5	
misuse	탐	탐	탐	탐						threshold
clustering	14	38	3	14	80	3	3	44	60	5
association	80	70	69	72	30	80	70	80	30	40

[그림 11] 오용행위 및 비정상행위도

그림 11은 오용행위탐지시스템과 클러스터링 및 연관규칙을 가지는 비정상행위탐지시스템에 대한 침입 테스트 결과이다. 그림 11에서 보는 바와 같이 오용행위에 대해서는 알려진 공격에 한해서만 탐지를 하였고 비정상행위에 대해서는 탐지를 못하였다. 여기에서 오용행위에 대한 한계점을 알 수 있었다. 클러스터링인 경우 임계치가5 이하인 경우는 오용행위 중 solaris_land인 경우이며 비정

상행위에 대한 임계치 이하 값을 가지는 경우는 평소 사용하지 않은 서비스에 대한 anomal2와 낯선 호스트로부터의 로그인을 발생하는 anomal3에 대해 낮은 비정상행위도를 나타내고 있다. 이러한 결과는 분포와 빈도에 따른 유사성을 가지는 클러스터링의 특성을 잘 나타내줌을 알 수 있다. 연관규칙인 경우 오용행위에 대해서는 비정상행위도가 높은 값을 가지며 비정상행위 테스트인 경우 평소보다 많은 다량의 서비스를 발생하는 anomal1에 대해서는 클러스터링에 비해 낮은 비정상행위도가 나왔다. 클러스터링에서 임계치 값 이하를 가지는 anomal2와 anomal3에 대해서는 높은 비정상행위도가 나왔으며 근무시간외 트래픽을 발생하는 비정상행위테스트에서는 클러스터링보다 낮은 비정상행위도가 나왔음을 알 수 있다. 이러한 결과는 네트워크상의 행위가 자주 발생되더라도 연관성이 없으면 의미가 없다는 것을 나타내주고 있음을 알 수 있다. 이 실험을 통해서 알 수 있는 것은 클러스터링의 경우 빈도와 분포의 성격을 가지는 행위에 대해서 묘사성이 높고 연관규칙인 경우는 빈도와 분포의 성격보다는 데이터의 의미 있는 연결성을 가지는 행위에 대해 묘사성이 뛰어남을 알 수 있었다.

5. 결론 및 향후 연구방향

본 연구에서는 방대한 데이터 분석을 좀 더 지능적이고 자동적으로 수행하기 위해서 인공지능 분야에서 활용되고 있는 데이터 마이닝 기법인 클러스터링과 연관규칙을 활용하였다. 오용행위와 비정상행위에 대한 탐지영역을 분석하기 위해 대표적인 DoS(Denial of Service)와 Probing, 비정상행위를 발생시켜 실험을 하였다.

결론적으로 오용행위탐지시스템만을 가지고 네트워크상에서 일어나는 침입행위에 있어서 모든 탐지를 한다는 것은 어느 정도 한계가 있음을 알 수 있었고 이러한 한계를 극복하기 위한 많은 연구가 진행 중이나 각각의 연구에서 이용되는 모델은 알고리즘 특성에 따라 다양한 탐지영역을 가지고 있음을 알 수 있었다. 하나의 모델이 네트워크상의 모든 침입행위를 탐지할 수 있는 것은 불가능하며 다양한 알고리즘을 가지는 모델이 상호 보완적인 구조로 침입탐지시스템이 발전되어야 한다는 것을 알 수 있었다. 침입탐지시스템 개발에 있어서 본 연구에서는 탐지영역 분석을 위해 클러스터링과 연관규칙 두 모델만을 사용하였으나 좀 더 다양한 모델에 따른 탐지영역을 분석해야 하는 앞으로의 연구방향을 제시하였다. 향후 연구로는 판정요소에 대한 좀 더 구체적인 연구가 필요하며, 네트워크상의 다양한 공격을 탐지하기 위해서는 모델의 특성에 따른 탐지영역을 구체화시키고 이를 바탕으로 여러 가지 모델을 적용할 수 있는 통합 메커니즘 연구가 필요하다고 하겠다.

참고문헌

[1] M. Sobirey, B. Richter, and H. Konig. The intrusion detection system AID. Architecture, and experiences in automated audit analysis. In Proceedings of the IFIPTC6/TC11 International Conference on Communications and Multimedia Security, pages 278-290, September 1996.

[2] G. B. White, U. W. Pooch. Cooperating Security Managers: distributed intrusion detection systems. Computers & Security 15(1996)5, pages 441-450.

[3] Gregory B. White, Eric A. Fisch, and Udo W. Pooch. Cooperating security managers: A peer-based intrusion detection system. IEEE Network, 10(1):20-23, January/February 1996.

[4] Staniford-Chen, S., Cheung, S., Crawford, R., Dilger, M., Frank, J., Hoagland, J., Levitt, K., Wee, C., Yip, R., Zerkle, D. GrIDS - A Graph Based Intrusion Detection System for Large Networks. In Proceedings of the 19th National Information Systems Security Conference, pages 361-370, Oct. 1996.

[5] Winkler, J. R., Landry, L. C. Intrusion and anomaly detection, ISOA update. In Proceedings of the 15th National Computer Security Conference, pages 272-281, Oct. 1992.

[6] Winkler, J. R. A UNIX Prototype for Intrusion and Anomaly Detection in Secure Networks. In Proceedings of the 13th National Computer Security Conference, pages 115-124, Oct. 1990.

[7] Winkler, J. R., Page, W. J. Intrusion and Abnormal Detection in Trusted Systems. In Proceedings of the 5th Annual Computer Security Applications Conference, pages 39-45.

[8] Hochberg, J., Jackson, K., Stallings, C., McClary, J., DuBois, D., Ford, J. NADIR: An automated system for detecting network intrusions and misuse. Computers and Security 12(1993)3, May, pages 253-248.

[9] Jackson, K. A. NADIR: A Prototype System for Detecting Network and File System Abuse. In Proceedings of the 7th European Conference on Information Systems, Nov. 1992.

[10] Jackson, K., DuBois, D. H., Stallings, C. A. An

expert system application for network intrusion detection. In Proceedings of the 14th National Computer Security Conference, pages 215-225, Oct. 1991.

[11]http://www.wheelgroup.com/netrangr/1netrang.html.

[12] P.G. Neumann and P.A. Porras, "Experience with emerald to date", 1st USENIX Workshop on IDS, Santa Clara, Cal, 11-12 April 1999.

[13] Porras, A. and Neumann, P. G. EMERALD: Event Monitoring Enabling Responses to Anomalous Live Disturbances. In Proceedings of the National Information Systems Security Conference, October 1997.

[14] J. Frank, "Machine learning and intrusion detection : Current and future directions, " Proc. 17th National Computer Security Conference, October 1994

[15] 한국정보보호센터 "정보통신기반구조보호기술개발" 1999.12, 2000.12

[16] D. Anderson, T.Frivold and A. Valdes, " Next-generation intrusion detection expert system(NIDES)," Technical Report SRI-CLS-95-07, May, 1995

[17] Harold S. Javitz and Alfonso Valdes, "The NIDES Statistical Component Description and Justification," Annual Report, SRI International, 333 Ravenwood Avenue, Menlo Park, CA 94025, March 1994.

[18] Eleazar Eskin, Matthew Miller, Zhi-Da Zhong, George Yi, Stolfo ,"Adaptive Model Generation for Intrusion Detection Systems", Columbia University

[19] L.Todd Heberlein, Gihan V. Dias, Karl N. " A NETWORK SECURITY MONITOR", University of California, Davis 1990 IEEE

[20] Wenke Lee, Salvatore J.Stolfo " Data Mining Approaches for Intrusion Detection" Computer Science Department Columbia University 500 West 120th Street, New York, NY10027

[21] Martin Ester, Hans-Peter Kriegel, Sander, Michael Wimmer, Xiaowei Xu, "Incremental Clustering for Mining in a Data Warehousing Environment", Proceedings of the 24th VLDB Conference, New York, USA, 1998.

[22] Sudipto Guha, Rajeev Rastogi and Kyuseok Shim, "ROCK: A Clustering Algorithm for Categorical Attributes," the 15th International Conference on IEEE Data Engineering, Sydney, Australia, 1999.

[23] Sudipto Guha, Rajeev Rastogi and Kyuseok Shim, "CURE: An Efficient Clustering Algorithm for Large Databases," ACM SIGMOD International Conference on Management of Data, Seattle, Washington, 1998.

[24] Tian Zhang, Raghu Ramakrishnan, and Miron Livny, "Birch: An Efficient data clustering method for very large databases," Proceedings for the ACM SIGMOD Conference on Management of Data, Montreal, Canada, June 1996.

[25] Wenke Lee, Salvatore J.Stolfo, Kui W.Mok " A Data Mining for Building Intrusion Detection Models", Computer Science Department, Columbia University

[26] Bing Liu, Wynne Hsu, Yiming Ma, " Integrating Classification and Association Rule Mining", National University of Singapore 119260, KDD-98, New York, Aug 27-31, 1998

[27] Wenke Lee, Salvatore J.Stolfo, Wei Fan, Shlomo Hershkop " Real Time Data Mining-based Intrusion Detection" Computer Science Department, Columbia University

[28] Rakesh Agrawal, Ramakrshnan Srikant, "Fast Algorithms for Mining Association Rules", In Proc. Of the 20th VLDB conference, 1994

[29] 신대철, 이보경, 유동영, 김홍근 "네트워크 비정상행위탐지를 위한 클러스터링 모델"(한국정보보호진흥원) 2001. 10. WISC 발표

신 대 철(Dea-Cheol Shin) **[정회원]**

- 2000년 8월 : 홍익대학교 일반대학원 전산학과(이학석사)
- 2005년 8월 : 한서대학교 일반대학원 디지털포렌직학 (박사재중)
- 2000년 4월 ~ 2009년 2월 : 한국인터넷진흥원 연구원
- 2009년 3월 ~ 12월 : 한서대학교 포렌직학과 겸임교수

<관심분야>
정보보호, 시스템/네트워크 보안, 경영정보시스템, 바이오인식, 정보통신, 디지털 포렌식

김 홍 윤(Hong-Yoon Kim) **[정회원]**

- 1982년 2월 : 인하대학교 전자계산학과 (이학사)
- 1984년 2월 : 인하대학교 전자계산학과 (이학석사)
- 1996년 2월 : 인하대학교 전자계산학과 (이학박사)
- 1995년 3월 ~ 현재 : 한서대학교 컴퓨터공학전공 교수

<관심분야>
센서 네트워크, 디지털 포렌직