通信情報保護學會誌
第9券 第3號, 1999. 09

에이전트 기반의 침입탐지 시스템 구현

김 회 준*, 최 종 욱**

요 약

본 논문에서는 독립 에이전트 기반의 침입 탐지 시스템을 구현하고자 한다. 침입 탐지 시스템의 메시지 유실 및 메시지 내 정보에 대한 처리를 보다 안정적으로 제공하기 위해, 단위 침입 행동별로 학습된 모니터링 프로세스에서 전송되는 사용자 위협 행동 메시지에 대한 처리를 담당하는 조정자 에이전트 시스템을 제안한다. 본 논문에서 제안된 조정자는 안정화된 메시지 처리 문제 뿐 아니라, 기존 모델의 에이전트간 협력 작업에 의해 처리되었던 침입 판단 기능 및 모니터링 프로세스들의 관리 기능 또한 수행하도록 하게 된다. 따라서, 시스템의 유연성 및 확장성 향상을 도모하도록 하였다.

I. 서론

네트워크에서의 침입은 크게 오용침입(misuse intrusion)과 비정상적인 침입(anomaly intrusion)의 두 가지로 분류한다[8]. 오용 침입(misuse intrusion)은 시스템이나 응용프로그램 소프트웨어의 알려진 약점이나 버그를 이용한 침입으로 예를 들면, 인터넷에서 사용되는 send mail이나 fingerd의 버그를 예로 들수 있다. 비정상적 침입(anomaly intrusion)은 정상적인 시스템의 사용 패턴들과의 차이(deviation) 관찰에 의해 침입이 결정되는 것으로, 모니터 되어야 할 시스템의 프로파일(profile)을 구축하여 이로부터 중요한 차이를 찾아냄으로써 침입이 탐지된다. 오용 침입은 잘 알려진 패턴을 따르므로 감사증적(audit trail) 정보와의 패턴 매치를 통해 탐지할 수 있다. 그러나 이에 비해 비정상적 침입은 탐지하기가 쉽지 않다.

그리고 정상적인 사용자의 행동에 대해 이를 침입으로 분류해 버리는 경우, 이를 무시할 수 있는 능력 또한 갖추어야 한다. 따라서 이에 대한 판단은 보안 담당자에 의해 사전에 규정해 놓는다던가, 오랜 시간 시스템을 사용해본 경험에 근거한다던가 하는 Heuristic적인 면에 의존하게 되는 것이다.

침입 탐지 시스템에 있어 행위 판별(behavior classification)과 자료 축소(data reduction) 문제를 중심으로 수행한 연구들이 있다. 행위 판별은 주어진 일련의 행위들에 대해 이것이 침입인지 침입이 아닌지를 판단할 수 있는지를 결정하는 문제이고, 자료 축소는 수 메가바이트에 이르는 방대한

* (주)다우기술 (hoijun@netscape.net), ** 상명대학교 정보통신학부 (juchoi@pine.sangmyung.ac.kr)

본 연구는 과기부 국제공동연구(98-I-01-03-A-104)의 지원에 의해 수행되었슴.

양의 분석되어져야 할 데이터에 대해 이의 양을 줄여 나가는 것을 말한다. 대체적으로 이러한 문제의 해결에 있어 도입된 인공지능의 기법들은 규칙 기반 시스템 (Rule-based system)과 신경망[5] 또는 통계적 분류 시스템의 방법을 사용하고 있다[7][10]. 이러한 접근법의 한계점들은 이들이 많은 양의 초기 학습을 필요로 하며, 시스템의 수명기간 동안 지속적으로 시스템의 유지보수에 많은 시간이 소비된다는 것이다.

규칙 기반 침입 탐지 시스템의 대표적인 예로는 IDES 시스템이 있다. 이 시스템은 대상 시스템의 취약성 및 보안 정책, 그리고 과거의 침입들에 대한 지식을 담고 있는 규칙 DB를 가지고 있다. 현재 시스템의 상태로부터 침입이 발생할 경우, 탐지 시스템은 침입에 대한 해당 규칙에 의해 현 시스템이 침입 당했는지 여부를 분류하게 된다. 그러나 IDES시스템은 통계적인 기존의 규칙 기반 접근법에 비해 과거의 침입에 대해 이를 기억하고 있다는 중요한 특징이 있다. 이것은 침입 탐지 시스템의 성능 향상에 있어 매우 중요한 특징으로 대부분의 새로운 침입 형태들은 기존 형태의 부분적인 변형이기 때문이다.

Handy, Luger등에 의해 제안된 새로운 솔루션으로는 분류 시스템(classifier sys- tem)을 이용하여 네트워크 시스템의 현재 상태를 분류해 내는 방법이 있다[8]. 이것은 네트워크 패킷 정보에 관한 매트릭스들을 구한 후 이들로부터 네트워크에 관한 분류를 어떻게 할 수 있는지를 추론하게 된다. 그러나 이러한 접근법은 크게 두 가지의 한계점을 지니고 있다. ATM이나 FDDI 백본과 같이 많은 컴퓨터들이 고속의 네트워크에 연결되어 있는 경우 성능 저하가 현격하다는 것이다. 또한, 네트워크의 상태를 판단하는데 사용되는 정보가 패킷의 헤더데이터에 국한되어 있다는 점이다. 헤더로 부터 추출한 정보를 이용할 경우, 행위의 특징을 처리할 수 없기 때문에 헤더의 정보만으로분류에 필요한 유용한 정보를 추출하기에는 역부족이다. 이에 대한 예로는 합법한 메일 포트를 통하여 침입이 이루어질 경우 이 접근법에 의한 시스템은 침입을 구분해 낼 수 없게 된다.

Kephart에 의해 제안된 접근법은 인간의 면역 시스템과 비슷하게 컴퓨터 바이러스에 대한 탐지 및 제거 시스템을 제안하였다 [9]. 이 접근법은 PC 상에서 바이러스에 초점을 맞추고 있을 뿐 전체 시스템 상에서의 비정상적인 행위 또는 시스템 침입에 대한 한계성을 가지고 있다. 그러나 Kephart의 접근법은 몇 가지 중요한 이슈에 대해 언급하고 있다. 먼저 사용자에 의한 행동이 침입일 수 있음을 판단하는 침입 인식의 문제와 IDES시스템과 비슷한 침입에 대한 학습, 그리고 침입이 탐지되었을 경우 이를 어떻게 다룰 지에 대한 침입 대응에 관한 내용들이다.

Kumar와 Spaffod에 의해 제안된 패턴 매칭 기법에 기반한 접근법은 시스템상에 요구되는 유연성의 향상에 대한 문제로 초점이 맞추어 졌지만, 학습능력을 갖추지 못하였다는 단점을 가지고 있다[10]. 이들은 시스템상에 나타나는 현상들에 근거하여 침입을 어떻게 분류하는 지를 보여주었다. 여기서의 각 패턴들은 시스템 상태들간의 의존도를 인코딩하고 있는 것이다. 이러한 접근법은 침입을 탐지하는 강력한 방법이나 사전에 만들어진 패턴들에 의존적이라는 단점 또한 가지고 있다. 즉, 패턴 자체가 완전하지 못할 경우 시스템의 방어에 커다란 헛점이 나타나게 되는 것이다. 그리고 보안 정책이나 시스템 운영상에 변화가 있을 경우 패턴들을 다시 생성해야 한다.

II. 침입탐지 시스템의 설계

기존에 제안된 대부분의 침입 탐지 시스템들은 하나의 통합된 단일 시스템구조를 가지고 있다. 커널과 같은 대상 시스템의 운영체제 위에 놓여져 커널로 들어오는 모든 처리 요구에 대해 모니터링을 수행한다. 그러나 이러한 시스템 모델들은 전체 시스템에 걸리는 부하문제 및 탐지 모듈의 파괴에 따른 안정성의 문제, 시스템의 확장에 따른 성능 보장의 문제 등과 같은 많은 문제점을 지니게 되었다[2]. 이러한 성능상의 문제들의 해결을 위한 방법으로 대상 시스템을 지역적, 또는 기능적으로 분할하는 방식이 있다. 이는 가벼운 형태를 가지는 다수의 프로세스들로 하여금 각각 독립적인 동작으로 분할된 시스템 자원을 모니터링 하는 것이다. 전체 시스템에

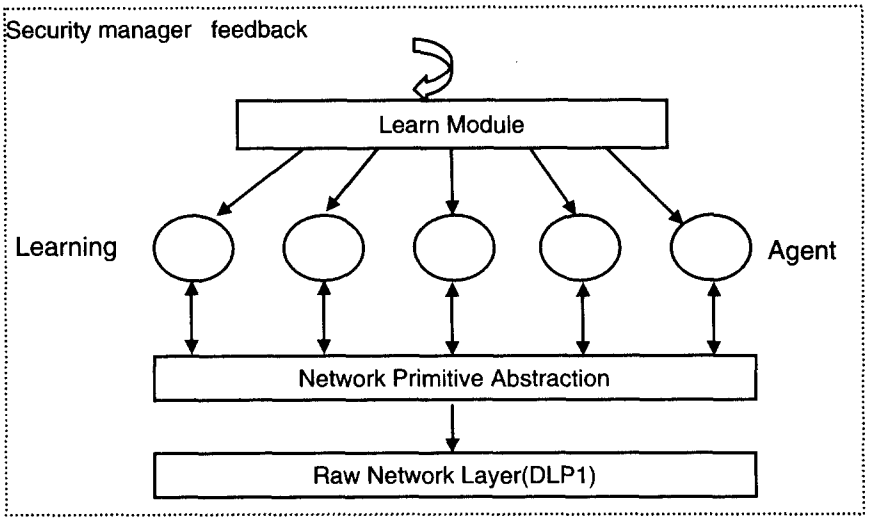

그림1. 독립 에이전트를 이용한 침입 탐지 시스템의 구성

대한 침입 발생시 이들 프로세스간의 협력을 통해 이를 탐지하도록 침입 탐지 시스템을 구성할 필요가 있다. 이러한 구조는 기존 시스템에 대한 부하문제 및 탐지 모듈의 파괴에 따른 전체 기능의 마비, 시스템의 확장에 따른 탐지 시스템의 확장성 보장 등의 문제점들을 해결할 수 있다.

1. 독립 에이전트 기반 침입탐지 시스템

본 연구에서 제안하고 있는 시스템에서는 각 모듈 프로세스들이 시스템 자원들에 대한 행동패턴을 관찰하여, 비정상적인 행동이라 여겨질 경우 이를 알릴 수 있도록 학습되어진 에이전트의 형태를 가진다. 학습을 위해서는 유전자 알고리즘을 사용하여 의심스러운 행위 패턴에 대해 이를 분별할 수 있도록 학습시킨 후 각 모니터링 대상이 되는 자원들 위에 올려놓는다.

시스템 사용자들에 의한 행동은 일련의 시스템 자원에 대한 접근 및 서비스 요구들로 이루어짐으로 각각 대상 시스템 자원에서 사용자의 행태가 모니터링된다. 상호 독립적으로 동작하는 에이전트들은 모니터링 중 비정상 행위가 발생하였을 경우, 주변의

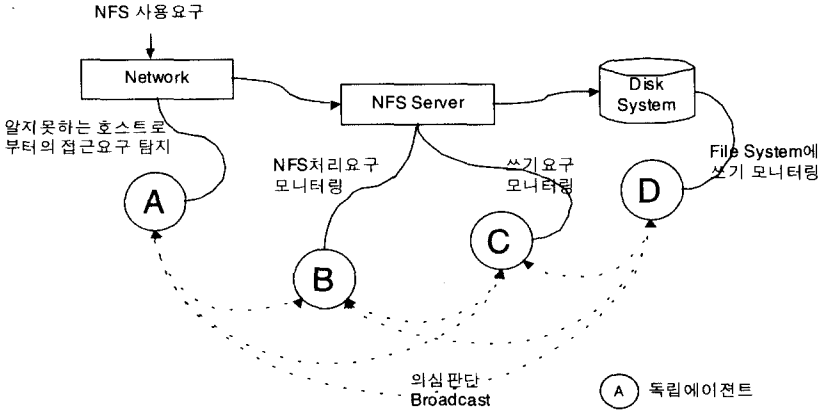

그림 2. 독립에이전트의 활동 예

에이전트들에게 알리고(broadcasting) 의심스러운 특정 사용자의 행위에 대해서는 계속적으로 모니터링을 하게 된다. 만약 그 행동이 시스템 전체에서 허용하는 특정 수위의 한계치(threshold)를 넘을 경우 이를 비정상적인 행동으로 여겨 이에 대한 대응 및 보고를 수행하게 된다. 이러한 침입에 대한 탐지 메카니즘은 탐지 시스템 상에서 발생할 수 있는 오류 중 False-positive(긍정적 오류) 오류의 발생가능성을 최소화시키기 위한 것이다. 그리고 각각의 에이전트들을 해당 리소스에 대한 모니터링을 위해 최적화해 구성할 수 있음으로 인해, 궁극적으로 침입 탐지 시스템의 성능 향상을 도모할 수 있다. 다음 그림은 독립에이전트 시스템의 작업 예를 보인 것이다.

2. 에이전트의 학습을 위한 Genetic Programming의 사용

본 논문에서 제안되는 탐지시스템에서 에이전트는 audit 정보를 모니터하기 위하여 침입프로파일에

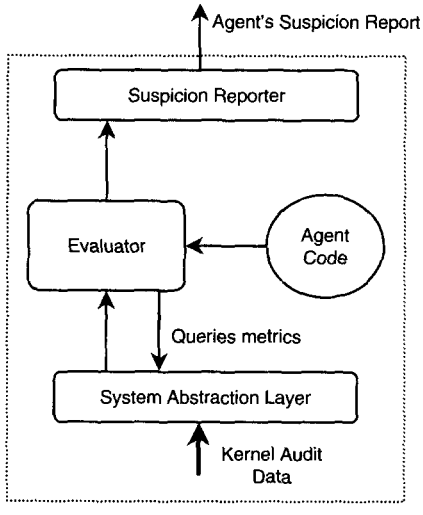

그림3. 독립 에이전트의 내부구조

대하여 학습을 수행한다. 이런 필요에서 도입된 Genetic Programming (GP)(Koza 1992)은 에이전트 프로그램이 새로운 침입유형과 알려지지 않은 침입유형에 따라 계속적으로 진화할 수 있도록 하며, 이는 실제 환경에서 탄력적으로 사용될 수 있다. Genetic Programming의 최종적인 결과는 프로그램의 집합이며 이 프로그램은 실제 시스템에 탑재되어 연속적으로 동작한다. Genetic Programming은 audit data field에 접근할 수 있으며, audit data를 잘 조정할 수 있는 성격을 가지는 단순한 언어로 코드화 된다. stand-alone solution에서 이 Genetic Programming은 audit 정보와 함께 이것을 제공하는 evaluator에 의해 해석된다. 아래그림은 에이전트의 내부구조를 보여주고 있다.

에이전트가 가지고 있는 코드는 진화과정으로부터 얻게 되고 evaluator에 의해 에이전트 안에 놓여진다. evaluator는 SAL (system abstraction layer)로부터 audit 정보를 획득하게 된다. SAL은 audit record에 대하여 다양한 통계를 계산하고 agent에게 그 정보를 제공하며 audit record를 해독하고 필요한 영역으로 추출한다. SAL은 모든 시스템의 기본적인 요소(평균 CPU 사용률, 평균 login 횟수 등)를 연속적으로 제공하는 목적을 가지고 있다.

2.1 Learning module

에이전트를 학습시키는 모듈로서 한번 학습된 에이전트들은 학습모듈과 독립적으로 동작할 수 있으며, 또한 에이전트들이 동작하는 동안의 모든 기록은 다음 학습의 입력으로 사용된다.

2.2 Agents

에이전트는 제안 침입탐지모델에서 중요한 역할을 하는 부분으로서 하위 네트워크layer에서 올라오는 패킷을 탐지모듈로 검사하여 시스템의 침입여부를 판단한다. 각 에이전트들은 자신이 맡은 모듈에 입력되는 메시지 검사를 담당한다. 이러한 검사 단계에서 침입으로 간주되면 침입 추정 값을 증가시킨다. 이 침입 추정 값이 특정 임계치에 이르면 이를 확실한 침입으로 간주하게 된다. 다음은 UDP bomb공격을 탐지하는 에이전트의 코드의 예이다.

```
for-each-packet do
if (get-subnet-part(ip-dest-addr-of-packet)
  is-not-equal-to my-subnet-address)
then
  generate-a-suspicion-broadcast
```

```
  if ( (packet-protocol equals UDP) and
      (udp-dest-port equlas 520)) then
       generate-a-suspicion-broadcast
  endif
endif
endfor
```

2.3 DLPI

Sun DLPI 인터페이스로 응용프로그램이 실제 데이터 링크 계층의 패킷을 전송하고 받을 수 있게 해주는 인터페이스이다.

2.4 Network Primitive Abstraction

DLPI은 인터페이스로부터 실제 네트워크 패킷을 받아 에이전트들이 그 패킷을 다룰 수 있도록 패킷의 구조를 바꾸어 주는 레벨이다. 에이전트들은 비정상적이라고 생각되는 일련의 행동에 의심에 대한 표시를 하고 시스템 행위를 관찰하도록 훈련 받는다. 이 원형에서 에이전트는 시스템상의 네트워크 트래픽 감시를 담당하게 된다. 에이전트를 이용한 시스템 감시는 먼저 시스템 관리자와 같이 시스템 보안에 관련된 사람의 수작업에 의한 지식데이터의 수립이 되어야 하는데 이것이 에이전트가 효과적으로 임무를 수행할 수 있는 선결 과제이다. 또한 이 지식데이터를 가지고 에이전트를 학습시키고 시스템에 적응시키는데 필요한 시간이 경우에 따라 많이 소모될 수 있다. 너무 낮은 레벨의 네트워크에서는 시스템 감시를 위한 정보의 수집이 어려워지게 되어 에이전트 본래의 목적과 상응하지 않게 된다.

아직까지 에이전트가 얼마나 빠르게 침입의 유형을 학습하고 정확하게 침입을 시스템 운영자에게 알릴 수 있는지에 대한 객관적인 근거는 없다. 만약 내부 사용자들 중 비정상적인 사용자 행위나 오용에 대한 패턴이나 경로가 빠르게 바뀔 경우 실제적으로 에이전트의 데이터추출, 학습 및 협동을 통한 감시는 상당한 한계를 가질 수 밖에 없다. 그러나 이러한 단점에도 불구하고 에이전트는 어느 정도 시스템에 대한 적응과 훈련을 위한 단계를 거치면 실제로 에이전트가 시스템의 감시자로서 역할을 하게 된다. 이는 과거의 시스템 침입 탐지를 위한 방법들보다 상당한 도움을 줄 수 있다.

3. 조정자 에이전트 중심의 협력에이전트 시스템의 설계

본 논문에서 제시하는 새로운 침입탐지 모델은 audit data를 통한 해석기를 이용하여 침입판단 및 탐지모듈과 시스템의 전반적인 조정할 수 있는 조정자 에이전트를 가진다. 또한, 그 하부에 활동적인(부분적인 제어를 단위행동 에이전트) 에이전트를 기반으로 두 단계의 에이전트의 구조를 가지고 있다. 상부의 에이전트는 genetic을 이용한 내부사용자의 비정상적인 사용과 오용 행위 패턴을 지식데이터베이스로 구축하며, coordinator agent와 autonomous agent간에는 black board 시스템을 이용한다. 이는 시스템상의 다양한 여러 에이전

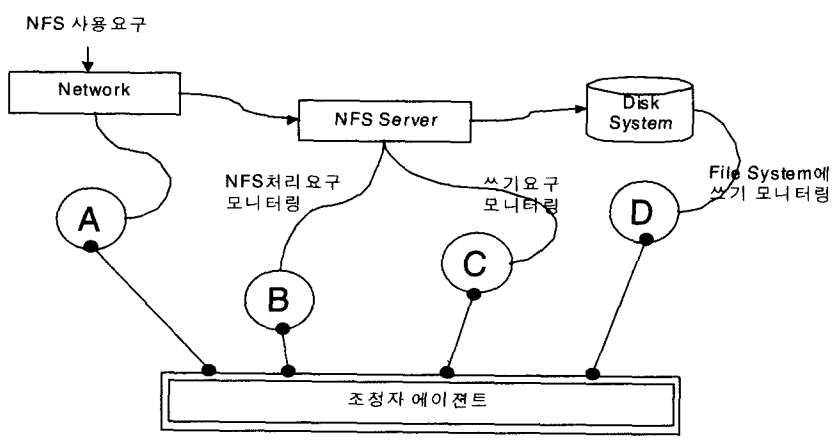

그림 4. 조정자 에이전트와 독립 에이전트간의 메시지교환

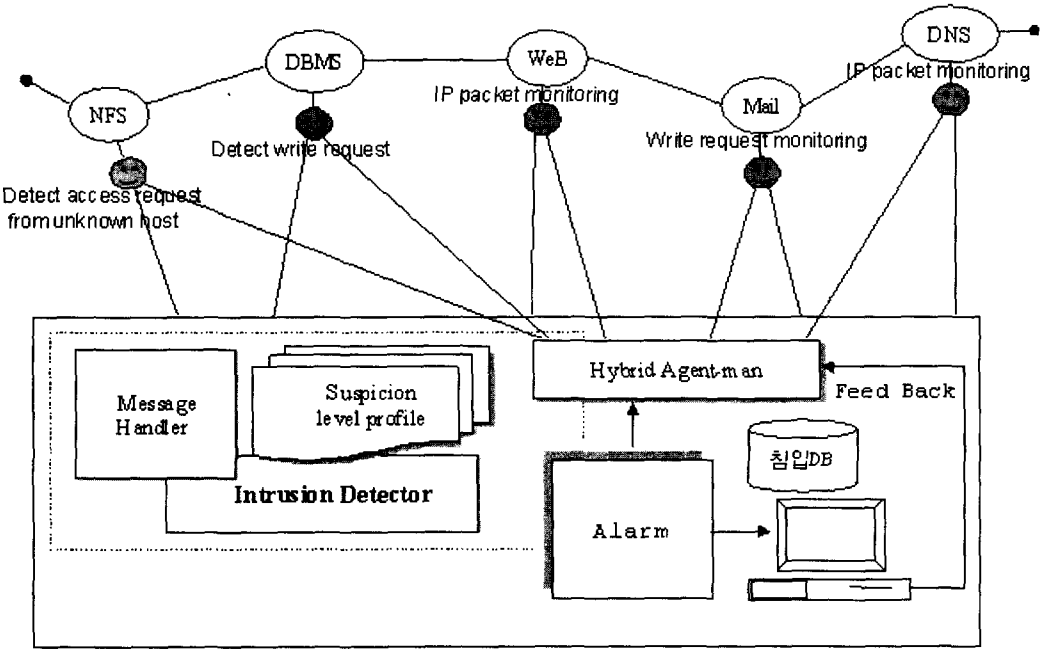

그림 5. 침입탐지시스템의 전체구조

트들이 협조하여 침입을 감시하고 새로운 형태의 침입에 대한 반응 속도를 높여 실시간적인 침입 탐지 시스템을 구축하기 위함이다. 제안 모델 상에서 핵심 부분인 조정자(coordinator)는 개별 작업을 수행하는 독립 모니터링 프로세스들과 일대 다로 연결되어 안정적 메시지 처리 지원, 침입 탐지 및 보고, 그리고 이들 모니터링 프로세스들로 구성되어진 전체 시스템 구조의 유지 및 운용을 담당한다. 본 논문을 통해 제안하는 조정자는 독립 에이전트 기반의 침입 탐지 시스템상에서 에이전트들에 의해 수행되어야 했던 많은 역할을 이양 받게 되며 이를 보완시키게 된다. 그러므로 제안 모델에서의 모니터링 프로세스와 FGA에서의 에이전트간에는 그 성격 및 역할에 있어 몇 가지 중요한 차이를 갖는다.

우선 제안 모델에서의 모니터링 프로세스는 FGA에서의 에이전트와 같이 위협행동을 수행한 수많은 사용자들에 대해 주위의 에이전트로부터 받은 메시지에 대한 처리 부하가 발생하지 않는다. 또한 회송 메시지를 전달하는 메시지 정보에 대해서도 제안 모델의 경우 이를 중앙의 조정자에서 관리하게 됨으로 모니터링 프로세스 상에 이 정보에 대한 별도의 저장 및 처리 모듈이 필요하지 않게 된다. 결국 FGA에서의 에이전트들에 비해 제안 모델에서의 프로세스들은 훨씬 가벼운 형태로 존재하게 된다. 에이전트의 역할은 학습에 의해 판단할 수 있는 개별 시스템 자원에 대한 사용자의 위협 행동을 찾아내 그 사실을 중앙의 조정자에 전달하기만 하면 되는 것이다. 이는 모니터링 프로세스 기반의 침입 탐지 시스템의 물리적, 또는 논리적 확장성을 제공할 뿐 아니라, 시스템의 유연성 등을 가지게 된다. 또한, FGA 모델이 갖는 기존의 중앙집중식 침입 탐지 시스템들에 대한 장점들을 강화할 뿐 아니라, 침입 탐지의 기반이 되는 메시지 처리를 중앙 조정자 내 메시지 처리자가 존재하여 안정적으로 제공하게 된다. 조정자 에이전트의 입장에서 시스템을 바라보면 조정자 에이전트는 시스템의 자체 침입 판별 알고리즘을 지원하기위해 모니터링 프로세스로부터의 메시지 가공, 사용자별 행위 데이터의 작성 및 이를 기반한 침입 판단, 그리고 이에 대한 보고 기능 등과 함께 전체 침입 탐지 시스템의 유지 및 운용을 위해 프로

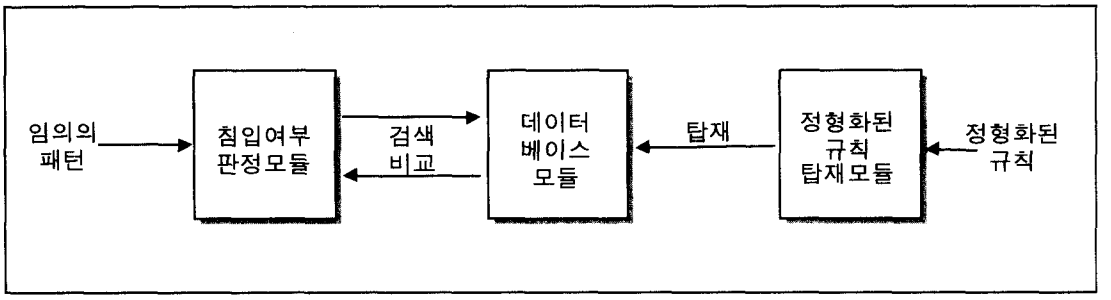

그림 6. 침입여부 판정엔진

세스들의 관리 기능까지 수행하게 된다. 이렇듯 모니터링 프로세스들과 상호작용하며, 침입에 대한 탐지 및 이에 대한 보고, 그리고 모니터링 프로세스들에 대한 관리를 조정자가 수행하게 된다. 이는 기존의 중앙집중식 시스템 구조를 가진 지식 기반의 침입 탐지 시스템과 유사성과 우수성을 가지게 된다.

III. 침입탐지 검사모듈의 구성

침입 탐지 검사모듈은 임의의 패턴을 source address와 protocol port&signature, 그리고 traffic분석을 통하여 침입행위인지를 판별한다. 다음 그림은 침입탐지 모듈의 구성도이다.

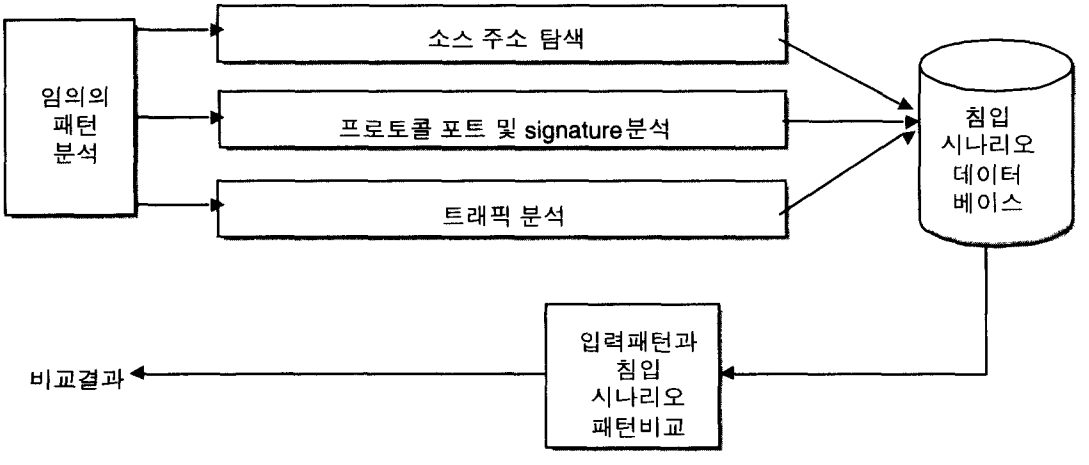

그림 7. 침입여부 판정모듈

임의의 사용자 행위는 침입여부를 판정하기 위해 위의 3가지 기준으로 '침입여부판정모듈' 에서 분석된다. 분석된 결과는 침입패턴들이 모여있는 '데이터베이스모듈' 과 비교하여 침입인지 아닌지를 가려낸다. 또한 새로운 침입패턴들이 발생한 경우에는 '정형화된 규칙 탑재모듈' 에서 데이터베이스 모듈로 새로운 패턴이 탑재된다.

2. 침입관련 데이터베이스 모듈

침입관련 데이터베이스 모듈은 여러가지 침입패턴들을 모아놓은 DB라고 할수 있는데, 이 DB에 얼마나 많은 양의 침입패턴들이 있느냐가 전체적으로 중요한 문제가 된다.

1. 침입여부 판정모듈

침입여부판정모듈은 임의의 패턴을 source address와 protocol port&signature, 그리고 traffic분석후에 침입관련DB와 검사하여 침입인지

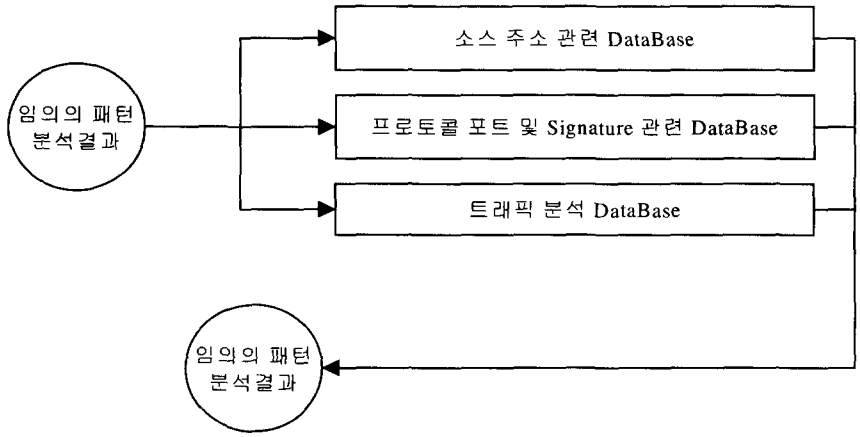

그림 8. 침입관련 데이터베이스 모듈의 설계

를 판정하는 모듈이다. 다음 그림은 침입여부 판정 모듈의 구성도이다.

침입여부 판정모듈에서 들어온 임의의 패턴을 분석하여 보내진 query들은 세 가지 기준을 모두 거쳐 침입인지 여부를 판정하게 된다. 다음 그림은 데이터베이스 모듈의 구성도이다.

위의 그림 구성도에서 세 가지 DataBase -source address관련 DataBase, protocol port& signature관련 DataBase,Traffic 분석 관련 DataBase - 대해서 각각 field구성과 실제 패턴 예를 살펴보도록 하자. 표 1은 source address관련 DataBase의 Field구성으로 Field

표 2. source address관련 DataBase Field

Field Name	Field Type	Field Length	Description
Rule Name	String	10	규칙의 이름
Source Address	String	15	근원지 주소
Source Port	Integer	2	근원지 프로토콜
Dest. Address	String	15	목적지 주소
Dest. Address	Integer	2	목적지 포트번호
Action	String	1	D(Deny)또는 P(Permit)허용 여부

표 3. source address 관련 DataBase의 패턴

Rule	Src.Addr.	Src.Port	Dest.Addr	Dest Port	Action
A	203.237.174.182	8	172.16.16.0	24	Permit
B	203.253.64.1	15	202.34.89.1	16	Deny
C	203.237.168.1	4	198.34.6.1	119	Deny

표 4. Protocol port & Signature 관련 DataBase의 Field

Field Name	Field Type	Field Length	Description
Rule Name	String	10	규칙의 이름
Command	String	10	사용한 명령어
Argument#1	String	10	인자 1
Argument#2	String	10	인자 2

는 rule name과 source address, source port, destination address, destination port, 그리고 허용여부를 결정하는 action으로 구성된다. 이에 대한 패턴 예를 다음 표에서 보여주고 있다.

표3은 protocol port & Signature 관련 DataBase의 Field구성표이다. 구성은 rule name, command, argument#1, argument#2로 구성된다. 실제 패턴 예를 표 4에서 보여주고 있다.

표 5. Protocol port & Signature 관련 DabaBase 의 패턴

Rule Name	Command	Argument1	Argument2	Argument3	.....
8	Mv	passwd	Pw.old		
9	rlogin	Host.com	-l	Toor	
10	ftp	-n			
11	Quote	User	ftp		
12	Quote	Cwd	~root		
13	Quote	Pass	ftp		
14	Tftp				

표 6. Traffic 관련 Database의 패턴

Field Name	Field Type	Field Length	Description
Rule Name	String	10	규칙의 이름
Command	String	10	사용한 명령어
Argument#1	String	10	인자 1
Argument#2	String	10	인자 2

표 7. Traffic 관련 Database의 패턴

Rule Name	Command	Argument1	Argument2	.....
1	Export	passwd	Pw.old	
2	./program_name	1124		
3	Directory_name	-fg	$EGG	
4	./program_name	768		
5	./vulnerable	$RET		
6	Trail	-5	Program_name	

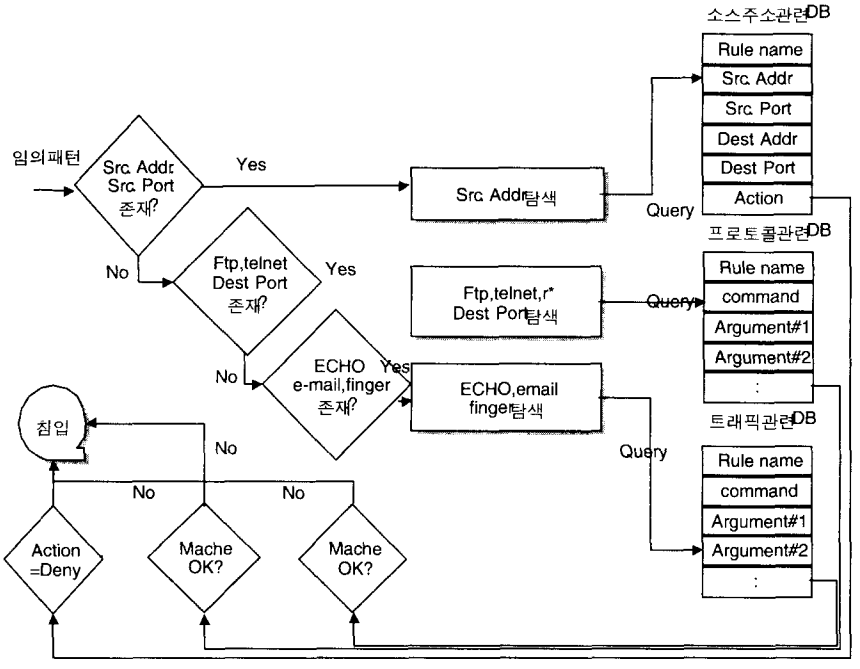

그림 10. 침입여부 판정모듈의 상세설계

3. 침입탐지모듈의 상세설계

지금까지 침입탐지 검사모듈의 세 가지 구성성분의 각각에 대한 구성도와 패턴 예를 살펴보았다. 그림 3-10는 침입여부 판정모듈의 상세설계이다.

임의의 패턴은 source address, source port가 존재하는지를 조사한다. 존재할 경우 탐색해서 source address관련 DataBase에 Query를 보낸다. 존재하지 않을 경우에는 다음 단계로 내려간다. 다음 단계에서는 ftp, telnet과 같은 protocol port등이 존재하는지를 검사한다. 이것도 마찬가지

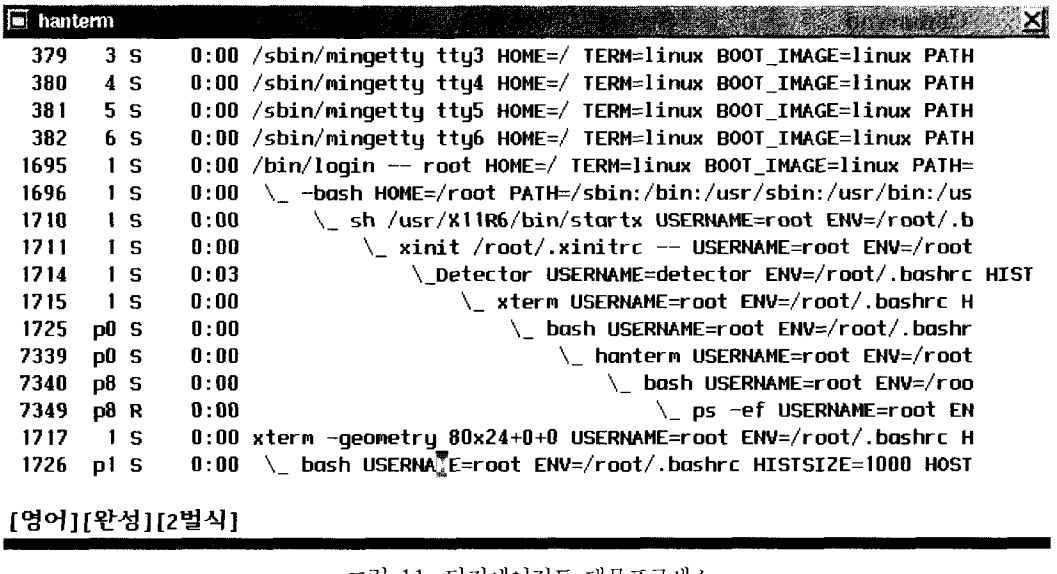

그림 11. 탐지에이전트 데몬프로세스

로 존재할 경우는 protocol port관련DB로 query

를 보내고 없으면 다음단계로 내려간다. 3번째 단계에서는 traffic과 관련된 e-mail, finger등이 존재하게 될 것이다. 여기서의 결과는 traffic관련 DataBase로 query를 보내게 된다. 이렇게 보내진 query들은 각각 matching이 되는지 검사되며, 한가지라도 matching이 되면 침입으로 결정된다.

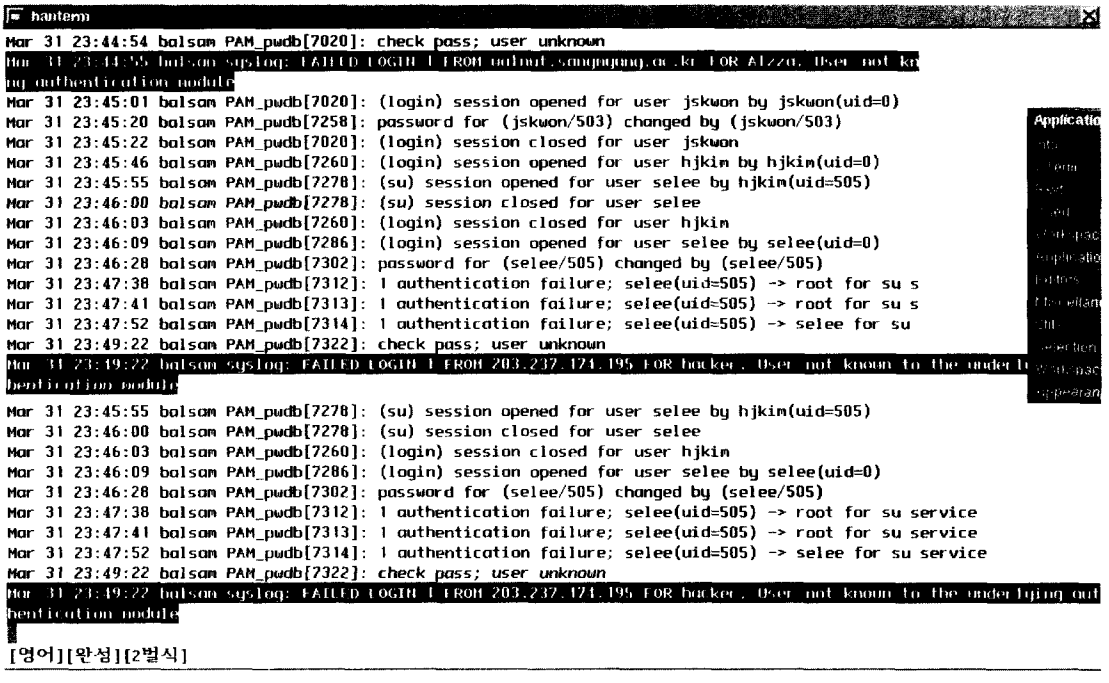

그림 12. 독립에이전트의 침입탐지과정

IV. 연구결과 및 Prototype

침입탐지 에이전트 프로그램은 Unix시스템상에서 데몬 프로세스(daemon process)로 동작하며 시스템에 대한 이상 발생시 즉각 화면상에 이상유무를 경고메시지와 함께 출력하며, 발생 대상 사용자를 지속적으로 감시하게 된다. 위의 그림.11 은 침입탐지 데몬 인 Detector가 데몬 프로세스로 동작중인

다음 perl script 코드는 위와 같은 시스템의 기본적인 침해 행위시 보안관리자의 모니터상에 메시지와 경고음을 출력하는 perl script program 의 configuration file이다.

```
# Detector 프로그램의 configuration file
/INVALID|REPEATED|INCOMPLETE/
  echo=inverse, bell=3
# login 이나 패스워드변경과 같은 행위 시 실행
/LOGIN/          echo=inverse, bell=3
/passwd/         echo=bold, bell=3
/ruserok/        echo=bold, bell=3
```

것을 프로세스 ID로 확인해본 그림이다. 아래 그림에서와 같이 여러 경로에서 login을 시도한 외부 사용자들이 단 한번의 login실패에도 콘솔상의 모니터상에서 login실패가 어느 경로로부터 일어났는가를 알 수 있다. 그리고 내부 사용자에 의한 루트권한으로의 시도역시 경고메시지와 함께 관리자의 모니터상에 출력되어 내부사용자에 의한 권한 남용을 사전에 인지할 수 있다.

```
# Ignore this stuff
```

```
/sendmail/,/nntp/,/xntp|ntpd/,/faxspooler/
ignore

# Report unusual tftp info
/tftpd.*(ncd|kfps|normal exit)/ignore
/tftpd/          echo,bell=3

# Kernel 과 관련된 행위 시 메시지 출력 및 경고
음
/(panic|halt|SunOS Release)/ echo=bold,
bell
/file system full/     echo=bold, bell=3
/vmunix.*(at|on)/ignore
/vmunix/          echo, bell

/fingerd.*(root|[Tt]ip|guest)/
echo,bell=3
/atkins/          echo=inverse,bell=3

/su:/          echo=bold,bell=4
/.*/          echo
```

V. 결론 및 향후 연구 과제

현재 국내에서는 침입탐지기술에 대한 연구가 활발하게 이루어지고 있다. 본 논문에서는 단일모듈의 침입탐지 시스템의 단점을 보안하고 기존의 중앙 집중식 구조의 침입 탐지 시스템의 장점을 수용하기 위해 모니터링 프로세스와 조정자를 가진 시스템 구조의 hybrid 침입 탐지 시스템을 제안하였다. 그리고 모델로만 제안되었던 단위행동 에이전트를 실제로 구현하여 보았다. 침입탐지모듈의 설계부분에서는 대체로 발생하는 침입 패턴들이 원치 않는 주소에서 오거나 특정서비스를 이용 혹은 시스템에 과부하를 걸고 다른 일을 하는 패턴이 많은데 주안점을 두었다. 따라서 이를DB와 DB에 새로운 침입패턴들을 탑재시키는 부분과 들어오는 패턴들을 DB와 비교하여 침입인지를 판정하는 부분으로 구성하였다. 그러므로 얼마나 많은 침입시나리오들이 데이터베이스에 구축되느냐가 중요한 문제이다.

참 고 문 헌

[1] Anderson J.P. "Computer security threat monitoring and surveillance", Technical Report, James P. Anderson & Co. April, 1980.

[2] Crosbie M.and Spafford G., "Defending a Computer System using Autonomous Agents", In Proceedings of the 18th NISSC Conference, October 1995.

[3] Denning D.E.,IDES-"an Intrusion Detection Model.", IEEE Trans. Software Engineering, February 1987

[4] Forrest S., Hofmeyr S. A., Somayaji A., and Longstaff T. A., "A Sense of Self for Unix Proccesses.", In Proceedings of the 1996 IEEE Symposium of Security and Privacy, pp120-128, 1996.

[5] Fox K.L., Henning R.R, Reed J.H. and Simonian R.P.,"A Neural Network Approach Towards Intrusion Detection.", Technical Report, Government Info. Systems Division, Harris Corp., July 1990

[6] Frank J., "Artificial Intelligence and Intrusion Detection: Current and Future Directions", NSA URP MDA904-93-C-4085, June, 1994.

[7] *Goldberg D. Genetic Algorithm in Search, Optimization and Machine Learning*. Addision-Wesley, pp24-31, 1989

[8] Heady R., Luger G., Maccabe A., Servilla M. "The architecture of a network level intrusion detection system". Technical Report, Department of Computer Science,University of New Mexico, August 1990.

[9] Kephart J. O., *A Biologically Inspired Immune System for Computers*, High

Integrity Computing Laboratory, IBM Thomas J. Watson Research Center, MIT Press, 1994

[10] Kumar S. and Spafford G. "A Pattern Matching model for Misuse Intrusion Detection". In Proceedings of the 17th National Computer Secutiry Conference, October, pp11-21, 1994

[11] Laird, P., "Machine-Learning in intrusion and misuse detection". In Proceeding of Workshop on Future Detections in Computer Misuse and Anomaly Detection, Univ. of California, Davis, March 1992

[12] Lawrence Livermore National Lab., Sandia National Lab.,"National Info-Sec Technical Baseline - Intrusion Detection and Response.". Oct. 1996.

[13]Continuous Assessment of a Unix Configuration: Integrating Intrusion Detection and Configuration Analysis. In Proceedings of the ISOC' 97 Symposium on Network and Distributed System Security.San Diego, California, 1997.

[14]Distributed Audit Trail Analysis.In Proceedings of the ISOC '95 Symposium on Network and Distributed Systems Security. San Diego, California, February 1995.

[15]Anderson, Valdes. SAFEGUARD Final Report:Detecting Unusual Program Behavior Using the NIDES Statistical Component

著者紹介

김 희 준 **(Hoi-Jun Kim)**

1996년 2월 세종대학교 경제학과 졸업 (학사)
1998년 2월 상명대학교 정보통신대학원 졸업 (이학석사)
1998년 ~ 현재 (주)다우기술 기술팀

※ 주관심 분야 : 네트워크 보안, 전자상거래

최 종 욱 **(Jong-Uk. Choi)**

1982년 2월 아주대학교 산업공학과 졸업 (공학사)
1982년 9월 서울대학교 경영학 석사과정 (이학석사)
1988년 2월 University of South Carolina 정보과학 졸업 (이학박사)
1986년 ~ 1987년 Johnson C. Smith University (Charlotte, NC)
1988년 ~ 1991년 한국과학기술원 시스템공학센터 인공지능 연구부 지식 처리 연구실장
1996년 ~ 1997년 전문가시스템학회 부회장
1998년 ~ 1999년 한국경영정보학회 부회장
1991년 ~ 현재 상명대학교 정보과학과 부교수

※ 주관심 분야 : 네트워크 보안, 암호분석, 디지털 워터마크