2001년 한국정보처리학회 추계 학술발표논문집 제8권 제2호

안전한 E-Business 모델을 위한 다중 웹 클러스터 그룹의 분산 침입 탐지 시스템

이기준*, 정채영**

*조선대학교 대학원 전산통계학과
**조선대학교 자연과학대학 전산통계학과
e-mail: limitlee@orgio.net

Distributed Intrusion Detection System of Multi-Web Cluster Group for Safe E-Business Model

Kee-Jun Lee*, Chai-Yeoung Jung**

*Dept. of Computer Science and Statistics, Graduate School, Chosun Univ.
**Dept, of computer Science and Statistics, College of Natural Sciences, Chosun Univ

요약

고가용 E-Business 모델을 위해 구축된 다중 웹 클러스터 모델은 구조적 특성상 내부 시스템 노드들이 노출되어 있으며, 불법적인 3자에 의한 고의적인 방해와 공격으로 정상적인 작업수행이 불가능할 가능성을 지고 있다. 따라서 구성된 시스템 노드들을 보호하고 불법적인 사용자로부터의 정보유출과 부당한 서비스 요구를 효과적으로 대응할 수 있는 보안 시스템이 필요하다. 제안한 분산 침입 탐지 시스템은 불법적인 침입을 탐지하기 위하여 일차적으로 Detection Agent를 이용한 작업요구 패킷의 검사를 수행하며, 이후 작업이 진행되었을 때 Monitoring Agent를 통하여 작업과정을 관찰하며 허용되지 않는 자원의 접근 및 요구가 발생하였을 때, 다른 시스템 노드와의 긴밀한 협조작업을 통해 침입여부를 판단한다.

1. 서 론

최근 인터넷 기술의 눈부신 성장과 고성능 마이크로 프로세서의 비약적인 발전은 시간적, 공간적 개념을 뛰어넘은 새로운 생활 문화공간을 만들어 가고 있다 많은 이들이 인터넷을 이용하여 원하는 정보를 취득하며 자신의 자료 및 자원을 다른 이들과 공유, 활용할 수 있게 됨에 따라 서버에서 처리해야할 데이터의 양과 사용자가 요구하는 정보의 진송량은 급속히 증대되고 있으며, 과중한 데이터 양은 네트워크의 병목현상(bottle netck), 데이터 처리량에 따른 시스템 부하(System load)등의 문제점들을 야기 시켰다.[1] 현재 이러한 문제점들을 극복하고, 다양한 E-Business 환경에서 효율적인 시스템을 운영을 위한 여러 방안들이 모색되어지고 있으며 이중 저가의 중형서버(middle server)를 병렬 클러스터 시스템으로 구성한 대형 병렬 서버가 등장하였다[2][3].

본 논문에서 제안한 분산 침입 탐지 시스템은 E-Business를 위해 구축된 다중 웹 클러스터 모델을 기반으로 하고 있다. 기존의 클러스터 모델은 고속의 지역 네트워크를 기반으로 일정 수준 이상의 시스템으로 구성되는데 반하여 다중 웹 클러스터 모델은 개방화된 웹 상에 존재하는 저가(low price), 저속(low speed)의 다양한 시스템 노드를 대상으로 구축된다. 구성된 시스템 노드들은 개방화된 네트워크 환경을 기반으로 구성되었기 때문에, 구조적 특성상 불법적인 3자에 의해 내부의 시스템 노드들이 노출되어있으며[4], 각 시스템 노드간의 협조작업을 진행할 때 고의적인 방해와 공격으로 정상적인 작업 수행을 불가능하게 할 수 있는 가능성을 지니고 있다[5]. 따라서 이러한 불법적인 공격에 대하여 시스템 노드를 보호하고 불법 사용자로부터의 정보유출과 서비스 요구를 효과적으로 대응할 수 있는 보안 시스템이 필요하다[6].

분산 침입 탐지 시스템은 시스템 노드들에 대하여 불법적인 요구나 시스템 자원에 대한 접근을 탐지하는 기술로서, SC-Server의 공유메모리를 이용한 SC-Agent 간의 유기적인 제어방법을 제시하기 위하여 먼저 각각의 시스템 노드에서 독립적으로 수행하는 SC-Agent를 설계하고, 각 SC-Agent 간 긴밀한 질의와 협조를 통한 분산 침입탐지 시스템을 제안한다.

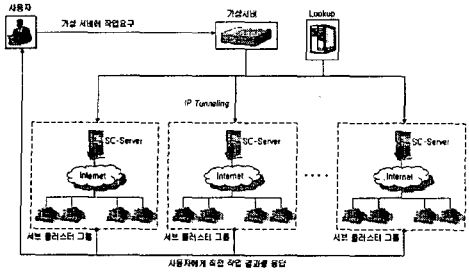
본 논문은 다음과 같이 구성되어 있다. 먼저 2장에서는 제안된 분산 침입 탐지 시스템의 배경이 되는 다중 웹 클러스터 모델에 대하여 기술하며 3장에서는 SC-Agent의 설계와 이를 이용한 분산 침입 탐지 방식에 대하여 기술한다. 그리고 4장의 실험 및 고찰에서 제안된 분산 침입 탐지 시스템의 효율성과 활용가능성을 모색하고 마지막 5장에서 결론을 맺는다.

2. 다중 웹 클러스터 모델

다중 웹 클러스터 모델은 서브 클러스터 그룹들을 기반으로 구성되며 사용자가 요구하는 대규모의 작업을 부하분배 및 병렬 컴퓨팅 방식을 이용하므로 처리 효율의 극대화시킬 수 있다.

2.1 다중 웹 클러스터 모델 구성

다중 웹 클러스터 모델은 단일한 가상 네트워크에 묶여져 있는 서브 클러스터 그룹과 서브 클러스터 그룹의 집합체인 다중 웹 클러스터 모델로 구분한다. 서브 클러스터 그룹은 일정 수량의 시스템 노드들을 동적(dynamic)으로 구성하고, 가상 서버로부터 진송되어온 사용자의 서비스 작업을 분산 처리할 수 있는 병렬 컴퓨팅 구조로 구성되어 있다.



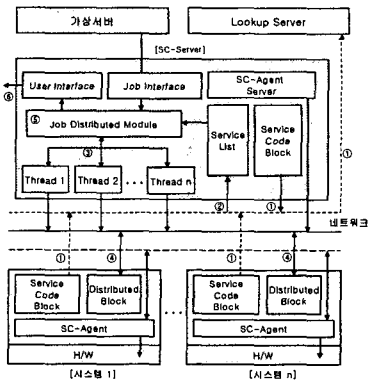
[그림 1] 다중 웹 클러스터 모델의 구성도

가상서버에 의해 묶여진 서브 클러스터 그룹은 네트워크 상에 분산되어 있는 여러 시스템 노드들을 대표하는 SC-Server에 의해 하나의 노드로 그룹화 되어있고, 이를 외부에서 바라볼 때 서브 클러스터 그룹은 한 개의 노드로 구성된 단일 시스템으로 보여지게 된다. 따라서 가상서버와 SC-Server 간의 연결은 단일 네트워크 상에 묶여진 추상화된 내부 네트워크로, SC-Server와 시스템 노드간은 개방 네트워크상에 연결된 외부 네트워크로 인식하게 된다.

2.2 서브 클러스터 그룹의 구성

가상서버의 서비스 수행요구에 의해 구성된 서브 클러스터 그룹은 내부의 노드중 한 개의 시스템 노드를 동적 선출하여 SC-Server의 임무를 부여한다. 따라서 다중 웹 클러스터 그룹을 구성하고 있는 전체 노드는 모두 SC-Server가 될 수 있는 잠재적인 가능성을 지니고 있다.

[그림 2]는 구성된 서브 클러스터 그룹과 가상서버, 전체 시스템 노드의 서비스 모듈을 통합 관리하는 Lookup Server의 작업과정을 나타내고 있다. Lookup Server는 시스템 노드들의 서비스 코드 블록(Service Code Block)이 등록되어 있으며, SC-Server나 다른 시스템 노드가 요구하는 서비스 코드 블록을 제공하는 역할을 수행한다. 이때의 서비스 코드 블록은 해당 시스템 노드에서 처리할 수 있는 서비스의 코드이다.

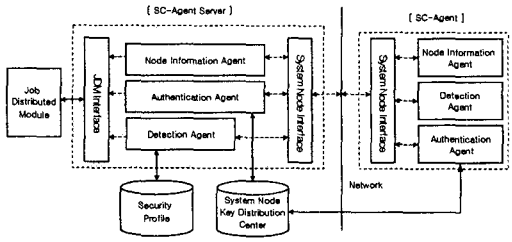


[그림 2] 서브 클러스터 그룹

3. 다중 웹 클러스터 모델의 침입탐지 시스템

3.1 시스템 보안을 위한 SC-Agent 구조

SC-Agent는 서브 클러스터 그룹을 구성하는 시스템 노드의 Agent 모듈로 노드간의 상호인증과 불법적인 침입에 대한 탐지기능을 지니는 Agent로 구성되어있다.



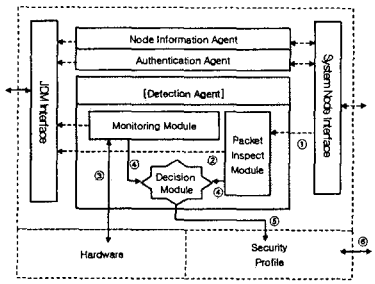
[그림 3] SC-Agent의 구조

[그림 3]에서 SC-Agent Server의 역할은 서브 클러스터 그룹내의 시스템 노드중에서 SC-Server로 선출된 시스템 노드의 SC-Agent가 수행하게 된다. SC-Agent Server는 구성된 시스템 노드내의 SC-Agent와 유기적으로 시스템의 정보와 인증, 침입탐지에 관한 정보를 교환한다. SC-Agent 모듈은 세 개의 Node Information Agent, Detection Agent, Authentication Agent로 구성되며 이외에 Job Distributed Module과의 자료 전달을 위한 JDM Interface와 System Node Interface로 구성되어진다.

3.2. SC-Agent내의 Detection Agent의 구조

Detection Agent는 시스템 노드에 전송되는 작업요구 패킷을 검사하는 패킷 검사모듈과 불법적인 침입을 탐지하는 모니터링 모듈로 구성되어있다. 이때 운영되는 시스템 노드에 침입으로 간주되는 징후가 탐지되었다면 SC-Server의 보안 프로파일을 통해 침입여부를 확인한다. 구성된 Detection Agent의 동작방식은 다음과 같다.

SC-Server 또는 일반 시스템 노드가 System Node Interface를 통하여 다른 시스템 노드로부터 작업요구 패킷을 전달받으면(①) Detection Agent의 Packet Inspect Module은 전달받은 작업패킷을 분석하여 일차적인 불법적 행위여부를 탐지하게 된다. 만일 작업패킷이 정상적인 패킷이라면 이를 JDM Interface를 통하여 Job Distributed Module (SC-Server인 경우) 또는 Distributed Block (일반 시스템 노드의 경우)으로 전달한다(②). 그러나 만일 요구된 작업패킷이 정상적인 패킷이 아닌 경우 Decision Module로 전달되어(④) 침입여부를 판정한다. 패킷 검사 모듈로부터 일차적인 검증을 받은 작업패킷은 Job Distributed Module을 통하여 시스템 노드에 요구된 작업 내용을 수행한다. 이때 Monitoring Module은 현재 작업과정이 시스템 내부에서 수행되는 상황을 주시하여(③) 만일 이상의 징후가 발견되었다면 Decision Module에 현재 불법침입여부를 문의하게된다(④). Packet Inspect Module과 Monitoring Module로부터 이상징후의 메시지를 전달받은 Decision Module은 구성된 전문가 영역인 보안 프로파일(security profile)을 이용하여 전달받은 이상징후에 대한 침입여부를 판정한다.



[그림 4] Detection Agent의 동작방식

3.3 분산 침입탐지 시스템

분산 침입 탐지 시스템은 각 시스템 노드에서 수행되는 SC-Agent 모듈의 Detection Agent를 이용하여 현 시스템에 대한 수행 현황을 파악하고, 만일 이상징후가 발견되었을 때에는 SC-Server의 보안 프로파일에 의해 침입여부를 판단하게 된다. Detection Agent는 불법적인 침입탐지를 감지하기 위하여 두 단계의 보안과정을 수행한다. 먼저 패킷 검사 모듈은 시스템 노드에 진달된 작업 패킷에 대한 정보를 분석하여 제약사항에 위배되는 모든 네트워크 패킷을 불법적인 침입으로 간주하고 이를 SC-Server의 Security Profile에 저장한다. 두 번째 보안방식은 Monitoring Module을 이용하여 현재 수행중인 작업의 상황을 관찰하고 있다가 만일 허용되지 않는 자원의 접근이나 불법적인 행위가 발견되었으면 이에 대한 적극적인 질의와 응답을 통해 침입여부를 판단하는 방식이다. 만일 불법 침입으로 간주된 경우 이에 대한 패턴을 Security Profile에 저장하여 이후 불법침입여부의 판단근거로 사용한다.

3.3.1 패킷 검사

Packet Inspect Module은 패킷이 주어지면 패킷검사를 통하여 불법침입을 결정한다. 작업패킷 검사의 내용은 다음과 같다.

(a) 서브 클러스터 경계 검사

서브 클러스터 경계검사는 작업을 의뢰한 시스템 노드가 현재 어떠한 서브 클러스터 그룹에 속해져 있는가를 검사한다. 현재의 시스템 노드가 소속된 서브 클러스터 그룹을 내부 클러스터 그룹이라 하고 그 외의 클러스터 그룹을 외부 클러스터 그룹, 다중 웹 클러스터 모델에 포함되지 않는 시스템을 외부 네트워크라 할 때 네트워크 연결은 근원지 주소와 목적지 주소에 따라 다음과 같이 구분될 수 있다.

① 내부 클러스터그룹의 시스템 노드로부터의 작업의뢰
② 외부 클러스터그룹의 시스템 노드로부터의 작업의뢰
③ 외부 네트워크로부터의 작업의뢰

이중 ③의 외부 네트워크로부터의 작업이 의뢰되는 경우는 해당 클러스터 그룹을 구성하고 있는 시스템 노드가 아닌 다른 외부의 시스템으로부터의 접근시도임으로 이를 침입으로 간주하여 접근을 거부한다.

(b) 서브 클러스터 그룹 검사

서브 클러스터 그룹을 구성하고 있는 시스템 노드들은 하나의 그룹으로 묶여지어 시스템 노드의 자원, 자료, 객체에 대하여 동일한 허가방식을 부여받는다. 만일 다른 서브 클러스터 그룹에 속한 시스템 노드가 작업을 의뢰하였을 경우 해당작업의 내용이 정의된 허가방식을 벗어난 행위를 수행하려할 경우에는 내부의 침입으로 간주한다. 예를 들어 '서브 클러스터 그룹 1'은 시스템 노드 ⓐ, ⓑ, ⓒ로 구성되어 있고, '서브 클러스터 그룹 2'는 시스템 노드 ⓓ, ⓔ, ⓕ로 구성되어있으며, 시스템 노드 ⓐ의 자원 A, B, C, D에 대한 접근허용 내용이 <표 1>과 같이 구성되어 있다면,

<표 1> 시스템 노드 ⓐ의 자원에 대한 접근 허용
Read : R Write : W Execute : X

	자원 A	자원 B	자원 C	자원 D
Owner	R, W, X	R, W, X	R, W, X	R, W, X
Group	R, X	R, W, X	R, W	R, W, X
Other	R, X	R, X	R, X	R, X

다음과 같은 상황들이 발생할 수 있다.

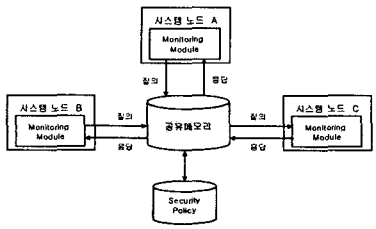
① 시스템노드 ⓐ로부터 자원A에 대한 쓰기(Write)요청
② 시스템노드 ⓑ로부터 자원B에 대한 실행(Execute)
③ 시스템노드 ⓒ로부터 자원C에 대한 읽기(Read)요청
④ 시스템노드 ⓓ로부터 자원D에 대한 쓰기(Write)요청
⑤ 시스템노드 ⓔ로부터 자원A에 대한 읽기(Read) 요청
⑥ 시스템노드 ⓕ로부터 자원B에 대한 쓰기(Write)요청

이때 상황 ①, ②, ③, ⑤는 지정된 접근허가 정책에 부합된 작업내용이므로 정당한 사용이 가능하다. 특히 상황 ⑤는 타 그룹(다른 서브 클러스터 모델)의 시스템 노드에 대해 읽고(Read), 실행(Execute)만 허용된 자원 A에 대해서 읽기(Read)요청을 요구했으므로 위반사항이 되지 않는다. 그러나 상황 ④와 ⑥의 경우 읽고(Read), 실행(Execute)만 허용된 자원 D와 B에 대해 쓰기(Write)요청을 하였으므로 접근허가에 위배되는 경우이다. 따라서 상황 ④, ⑥은 내부 침입으로 간주한다.

3.2.2 Monitoring Module

각 시스템 노드에서 수행되고 있는 Monitoring Module은 통제와 감시를 통해 시스템 상태에 의심스러운 징후가 발생하였을 경우 침입여부를 결정한다. 이는 다중 웹 클러스터 시스템의 Security Profile을 기반으로 SC-Server의 공유메모리를 이용한 타 Monitoring Module과의 질의와 응답을 통하여 보호하고자 하는 자원에 대한 침입여부를 판단한다. 만일 Packet Inspect Module의 패킷검사를 우회한 외부의 침입이 있을 경우 Monitoring Module은 이러한 침입여부를 탐지한다.

시스템 노드에서 수행중인 Monitoring Module은 시스템 내부를 감시하는 도중 불법적 침입의 징후가 나타날 경우 SC-Server내의 공유메모리를 이용한 다른 Monitoring Module과의 질의와 응답을 통해 서브 클러스터 모델의 불법 침입여부를 판단한다.
각 Monitoring Module은 상호간의 긴밀한 협조를 유지하기 위하여 [그림 5]에서와 같이 SC-Server의 공유메모리를 Monitoring Module간의 통신 중앙에 위치하여 Monitoring Module의 질의와 응답을 통해 침입에 대한 정확도를 높인다. Monitoring Module은 시스템 노드 내에서 주기적으로 발생하는 상태의 변화를 SC-Server의 공유메모리에 기록하여 타 시스템 노드들이 이를 활용할 수 있도록 제공하여주며, 만일 의심스러운 징후가 발생되었을 경우 시스템 자원에 대한 변경여부와 해당 상태를 Security Policy 내의 감사자료와의 비교를 통하여 침입여부를 판정한다.



[그림 5] 공유메모리를 통한 Monitoring Module간의 통신

4. 실험 및 고찰

4.1 작업 요구 패킷의 검사

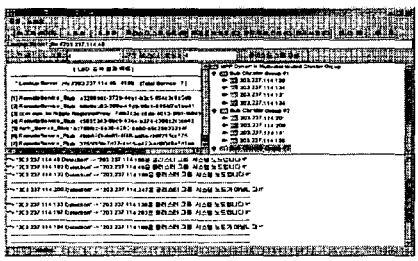
실험은 위하여 구성된 서브클러스터 그룹의 시스템노드에 임의의

불법적인 작업요구 패킷을 진달하였을 때 Detection Agent의 탐지 능력을 점검하였다. 진달된 작업요구패킷은 <표 2>를 기반으로 클러스터 그룹 경계검사를 수행하였다. 작성된 10개의 작업요구 패킷 중 3번, 6번, 9번 패킷은 웹 클러스터 그룹 외부의 네트워크에서 작성된 불법적인 작업 요구 패킷이다. 따라서 각 시스템 노드에서 수행중인 Detection Agent는 이들 패킷에 대한 네트워크 검사를 수행하여 외부 네트워크에서 진송되는 불법적인 패킷에 대한 일차 검사를 수행한다. [그림 6]은 진송된 작업요구 패킷에 대한 해당 시스템 노드의 Detection Agent의 클러스터 경계검사 결과를 SC-Server관리자에 보고하는 내용이다.

<표 2> 패킷실험에 사용한 작업요구 패킷

Nc	Source IP	Source Port	Destination IP	Destination Port	SC Group	Service
1	203.237.114.196	15487	203.237.114.48	4160	NO. 2	R
2	203.237.114.48	8457	203.237.114.193	4160	NO. 3	W
3	203.237.114.247	7548	203.237.114.200	4160	NO. 2	W
4	203.237.114.132	1457	203.237.114.199	4160	NO. 4	W
5	203.237.114.138	2345	203.237.114.133	4160	NO. 4	R
6	203.237.114.100	7845	203.237.114.194	4160	NO. 2	W
7	203.237.114.203	12954	203.237.114.197	4160	NO. 3	R
8	203.237.114.200	8127	203.237.114.206	4160	NO. 1	R
9	203.237.114.158	5486	203.237.114.136	4160	NO. 1	W
10	203.237.114.134	22457	203.237.114.199	4160	NO. 1	R

불법적인 작업요구 패킷을 진송 시스템은 Security Profile에 저장되어 다중 웹 클러스터 그룹을 구성하고 있는 모든 시스템 노드는 이후 해당 시스템에서 진송되는 모든 패킷에 대하여 수신 거부를 수행한다.

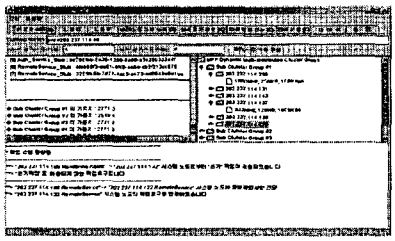


[그림 6] 작업패킷에 대한 클러스터경계검사

4.1.2 시스템 노드 자원 관리

시스템 노드 자원관리를 위하여 SC-Agent 모듈의 Monitoring Agent는 해당 시스템 노드에서 수행되는 작업의 내용을 관찰하다가 불법적인 행위나 허용되지 않는 자원에 대한 불법적인 접근이 시도되었을 때 이를 침입으로 간주하고, 시스템 관리자에게 보고한다. 시스템 노드의 자원관리 방안은 위의 패킷검사를 통하여 일차 검증된 패킷의 작업요구사항을 관찰하는 이차적인 보안방안이다. <표 2>의 작업요구패킷에 대하여 일차적인 패킷검사를 통하여 작업수행이 요청된 1, 2, 4, 5, 7, 8, 10번 패킷들은 해당 시스템 노드에서 Monitoring Agent의 감시하에 작업을 수행한다. 실험을 위하여 같은 서브 클러스터 그룹내의 시스템 노드들 간에는 읽기(R), 쓰기(W), 실행(X)의 권한을 지니고, 다른 서브 클러스터 그룹간에는 읽기(R) 권한만 부여되었다고 설정하였다. 구성된 서브 클러스터 그룹의 시스템 노드는 이전 실험과 동일하다.

일차 패킷검사를 통한 7개의 패킷들중 4번 패킷은 1, 5, 7, 8, 10번 패킷의 경우 읽기(R)작업만을 요청하였으므로 소속된 서브 클러스터 그룹에 관계없이 시스템 노드에서 수행 가능한 작업이다. 2번 패킷은 서브 클러스터 NO. 3에 소속되어있는 "203.237. 114.48" 시스템 노드가 "203.237.114.193" 시스템 노드에 쓰기(W) 작업을 요청하였다. 이때 두 시스템 노드는 모드 서브 클러스터 NO. 3에 소속되어 있으므로 쓰기(W)작업은 정당한 요청작업이 된다. 그러나 4번 패킷의 경우 서브 클러스터 NO. 4에 소속된 "203.237.114.132" 시스템 노드가 서브 클러스터 NO. 3에 소속된 "203.237.114.199" 시스템 노드에 쓰기(W)작업을 요청하였다. 이는 같은 그룹내의 시스템 노드가 아님으로 "203.237.114.199" 시스템 노드의 Monitoring Agent에 의해서 불법적인 자원접근으로 간주되어 SC-Server의 관리자에 보고된다. 보고된 내용은 SC-Agent의 공유메모리를 기반으로 해당 시스템 노드와의 협조를 통해 조율된다. [그림 7]은 4번 패킷에 대한 불법적 접근을 SC-Server에 진달하고 이를 "203.237.114.132.Remote" 시스템 노드와의 협조를 통하여 작업내용을 변경하고 있다.



[그림 7] Monitoring Agent에 의한 작업감시

5. 결 론

다중 클러스터 그룹의 특성을 고려하여, 내부 시스템 노드 자원과 정보를 보호하며, 효율적인 작업수행을 제공하기 위하여 다중 웹 클러스터 모델의 침입탐지 시스템은 SC-Agent를 기반으로 한 시스템 노드의 보안을 수행하였다. 따라서 분산 침입 탐지 시스템은 분산 구축되어, 중앙 집중식 통합관리가 불가능한 다중 웹 클러스터 시스템의 특성을 고려하여, 각 시스템 노드에 대하여 개별적인 보안시스템을 적용하였다.

실험을 통하여 제안된 분산 침입탐지 시스템은 각 시스템 노드에 구성된 SC-Agent의 작업요구 패킷 검사를 통하여 불법적인 요구 패킷을 차단하였고, Monitoring Agent를 이용하여 불법적인 행위나 허용되지 않는 자원의 접근이 발생하였을 때 다른 시스템 노드의 Monitoring Agent와의 협조를 통하여 문제를 해결하였다. 향후 연구과제로 분산환경에서의 새로운 유형의 침입에 대한 탐지방법과 함께 거짓 탐지율을 최소화하는 탐지 알고리즘에 대한 연구가 수행되어야 하리라 사료된다.

참 고 문 헌

[1] Rajkumar Buyya, "High Performance Cluster Computing: Architectures and Systems". Vol 1. 1999, Prentice Hall, New Jersey, USA.

[2] Committe on Physical, Mathematical, and Engineering Sciences. Grand Challenges : High Performance Computing and Communikcations, National Science Foundation 1991.

[3] Lawrence Livermore National Laboratory, Accelerated Strategic "Computing Initivtive(ASCI)" http://www.llnl.gov/asci/ 1998.

[4] T.F.Lunt, "A Survey of Intrusion Detection Techniques", Computer & Security, Vol 12, No 4, Jun, 1993.

[5] Halsall, F. "Data Communications. Computer Networks and Open Systems" 4th Edition, Addison Wesley, 1996.

[6] Crosbie, M and E. H . Spafford " Active Defense of a Computer System using Autonomous Agents" Department of Computer Sciences, Purdue University CSD-TR-95-022. 1994.