리눅스 기반의 효율적인 보안 정책 적용을 위한 원격통합관리시스템 (Remote Integrated Management System for Applying Efficient Security Policy based on Linux)

김 동환* 김 현성**
(Dong-Hwan Kim, Hyun-Sung Kim)

요 약 보안관리 분야는 최근 전문적이고 세분화되는 보안 제품과 어플리케이션의 유지 관리 문제로 어려움을 겪고 있다. 현재 보안 관리 분야의 연구들은 정보보호관리체계 지침이나 표준 문서 그리고 보안 툴의 일괄적인 관리에 대해서는 수행되고 어플리케이션과 보안 정책을 함께 관리하는 연구는 없었다. 본 논문에서는 리눅스 서버에서 작동하는 어플리케이션을 통괄하면서 각 어플리케이션에 필요한 지속적인 업데이트와 어플리케이션과 서버에 맞는 적절한 보안 정책을 신속히 관리할 수 있는 원격통합관리시스템을 제안한다. 본 논문에서 제안한 시스템을 통하여 시스템관리자가 보안전문가가 아니더라도 효율적인 시스템 보안을 제공 할 수 있을 것으로 기대된다.

Abstract Management for security product and application is becoming more difficult because they became more specialized. Most of research is focused on combining policies for information security management policy, security standard, and security tools. However, there are no researches for total solution for both application and security policy. Thereby, the purpose of this research is to propose a remote integrated management system based on linux. The system could efficiently manage data update for application and policy update for a server supporting the distinct configuration of each server. By using the remote integrated management system, system manager with poor secure knowledge also could easily manage their system securely.

1. 서 론

인터넷의 확산으로 인해 정보화 환경이 가속화되면서 정보보호 기술개발도 다변화되고 있다. 내부 네트워크 환경에서 점차 외부 네트워크 환경으로 변화함에 따라 정보보호 제품 및 서비스도 개방화된 환경 속에서 보안을 담당할 수 있도록 발전하고 있다. 이러한 보안 패러다임의 변화로 많은 조직들이 자사가 보유한 유형, 무형의 정보보호를 위하여 앞다투어 보안 시스템을 구축하고 있으나, 제품 별로 보안 시스템의 인터페이스와 관리 방법이 상이하고 이를 통합할 수 있는 표준화된 상호연동 방안이 존재하지 않아 상대적인 관리비용이 증가하는 문제가 발생하여 복합적인 보안 서비스를 제공할 수 있는 통합형태 제품의 등장과 지속적인 모니터링 및 신속한 대응, 통합 관리 등의 기능을 제공하는 효율적이고 능동적인 보안 관리 제품인 통합보안관리시스템의 도입되고 있다[5].

* 경일대학교 컴퓨터공학과
** 경일대학교 컴퓨터공학과 교수

그렇지만 또 다른 문제로 부각되는 것이 그러한 통합보안관리시스템을 관리하는 관리자의 능력에서 또 다른 문제가 나타나고 있다. 해킹을 당한 기업체를 살펴보면, 보안 장비를 구입하지 않아서라기보다는 구입한 보안 장비를 적절히 운용하지 못함으로 발생하는 경우가 대부분이다. 즉 장비를 구입하는 것뿐만 아니라 장비의 운용 측면이 더 중요하다는 점인데 에스큐브에서 발표한 해킹 사례를 보면, 실제로 20여 개에 달하는 침입차단 시스템을 구입한 업체가 해킹을 당한 경우가 있었는데, 직접적인 원인은 침입 차단 시스템의 포트 하나가 열려 있었기 때문인 것으로 판명되었다[8].

기업에서도 보안 장비를 효과적으로 운영할 수 있는 전문적인 보안 지식을 가진 보안 관리자를 양성하기보다는 이러한 보안 관련 업무를 아웃소싱 하는 것이 오히려 더 경제적이라고 판단하고 있으며, 보안에 대한 지식과 기술력이 없는 기업들에게 보안 제품이 아닌 보안 서비스를 판매하는 보안 관제 서비스의 등장은 매우 반가운 일이 아닐 수 없다.

해외에서도 '보안 관제 서비스'와 꼭 같은 개념의 서비스를 찾아보기란 그리 쉽지 않다. 이와 비슷한 개념으로 MSP(Managed Security Provider)라는 개념이 등장해 있기는 하지만 아직 그 성숙도나 구체화 정도에서는 국내에 비해 단 한 걸음과 앞서 있지 못한 상태인 것으로 파악되고 있다[4].

현재 최적의 보안 관리는 보안 관제라 볼 수 있으나 이러한 보안 관제 또한 보안 툴과 보안 정책에 중점적으로 다루고 있으나 응용 어플리케이션에 관련한 문제는 배제하고 있다. 보안 관련 툴뿐만 아니라 Web Server, FTP Server, SSH Server, Mail Server등 자체 어플리케이션에 대한 보안 강화 또한 중요하다.

리눅스와 오픈 소스 진영에서는 이러한 많은 보안 툴들과 어플리케이션을 지원하고 있지만 그 많은 어플리케이션의 선택과 거기 맞는 보안 툴 결정 그리고 정책 적용과 어플리케이션의 업데이트는 너무나 많은 시간과 비용이 들어가게 된다.

하나의 서버에는 많은 어플리케이션들이 동작하고 있고 이러한 어플리케이션이 빠르게 업데이트되고 있는 시점에서 관리자가 이러한 것을 각 서버마다 업데이트하고 작동 어플리케이션에 맞는 적절한 보안 정책을 세우고 관리하는 데는 많은 비용이 든다.

본 논문에서는 이러한 어플리케이션과 보안 정책들을 원격으로 통합관리 할 수 있는 원격통합관리시스템을 제안한다. 제안한 시스템은 각 어플리케이션에 필요한 지속적인 업데이트와 어플리케이션과 서버에 맞는 적절한 보안 정책을 신속히 관리함으로써 시스템관리자가 보안전문가가 아니더라도 효율적인 시스템 보안을 제공 할 수 있을 것이다.

2. 관련 시스템의 현황 및 분석

본 장에서는 기존의 통합관리 시스템의 기능 및 특성을 이해하기 위해 보안 시스템의 종류를 살펴보고, 기존의 통합관리 시스템의 여러 유형에 대해서 알아보기로 한다.

2.1 보안 시스템

보안 시스템의 목적은 여러 형태의 보안서비스 구현을 통해 각종 위협으로부터 내부의 중요자산을 보호하는 것이다. 현재 적용 가능한 대표적인 보안시스템으로는 침입차단시스템, 침입탐지시스템, 안티바이러스 시스템, 취약점분석 시스템 등이 있다.

1) 침입차단시스템

외부망에서 해커 등 비인가자의 내부망 침입을 차단시키는 S/W 또는 H/W로서 주요 기능은 네트워크에 대한 접근통제나 외부 네트워크로부터의 보호이며 일반적으로 라우터나 응용게이트웨이에 구현된다[5].

2) 침입탐지시스템

침입탐지시스템은 사용자 및 외부 침입자가 컴퓨터 시스템 또는 네트워크의 자원을 정당한 권한없이 불법적으로 사용하기 위한

시도 또는 내부 사용자가 자신의 권한을 오용하여 권한이외의 자원을 사용하기 위한 시도를 사전에 탐지하여 그 피해를 예방하는 시스템이다[5].

3) 안티바이러스

최근에는 인터넷을 통한 바이러스의 유통이 급격히 증가함에 따라서 네트워크상에서 바이러스에 감염된 파일이 첨부문서의 형태로 들어오는 과정을 근원적으로 차단하는 안티바이러스 시스템이 부각되고 있다. 안티바이러스 시스템은 시스템 감시와 인터넷 감시 기능을 통해 모든 바이러스 유입경로를 24시간 백그라운드 동작으로 실시간 감시하고 파일의 복사, 이동이나 인터넷으로부터의 다운로드 등 다양한 상황에서 바이러스 유입을 차단시키는 종합적인 시스템이다[8].

4) 보안 리포팅 시스템

보안 리포팅 시스템은 보안 부서가 정책과 절차에 충실한지 추적하고, 그리고 조직내에 모든 취약성의 상태를 추적하는 메카니즘이다. 수동과 자동화 시스템 모두 이것을 위해 사용될 수 있다. 대부분의 경우에 두 가지 형태의 시스템으로 구성된다.

◎ 사용-모니터링

모니터링 메커니즘은 컴퓨터 사용 정책을 따른다는 것을 보증한다. 이것은 인터넷 사용을 추적하는 소프트웨어를 포함한다. 그 메커니즘의 목적은 조직의 정책을 일관되게 위반하는 고용인을 식별하는 것이다. 어떤 메커니즘은 또한 그러러한 시도의 로그를 유지하면서 그러한 접근을 막을 수도 있다.

모니터링 메커니즘을 사용하는 것은 또한 데스크탑 설치로부터 게임을 제거하는 간단한 설정 요구를 포함할 수 있다. 더욱 복잡한 메커니즘은 새로운 소프트웨어가 데스크탑 시스템에 로드되는 때를 확인하는데 사용될 수 있다. 그러한 메커니즘은 관리자와 보안 부서 사이에 협조를 필요로 한다.

◎ 시스템 취약성 정밀 검사

시스템 취약성은 보안에서 매우 중요한 토픽이다. 기본 운영 시스템의 설치는 보통 상당수의 불필요한 프로세스와 보안 취약성을 가져온다. 그러한 취약성의 확인이 현대의 도구를 사용하는 보안 부서에서는 간단한 문제인데 반해, 이러한 취약성의 정정은 관리자에게는 시간-소비적(time-consuming)인 프로세스이다.

2.2 통합보안관리시스템

1) Hybrid Integration 모델

Hybrid Integration 모델은 개별 보안시스템을 하나의 서버 또는 하드웨어에 탑재한 통합보안시스템 모델로서 침입 차단시스템 내부에 일부의 침입탐지시스템 기능을 탑재하는 것이 일반적이다. 그러나 이러한 시스템은 전문 침입탐지시스템에 비해 침입탐지 패턴 등의 수준은 떨어진다[1].

이 모델의 대표적인 사례로는 Cisco IOS 침입차단시스템이 있는데 이것은 기존 침입차단시스템에 가상사설망(VPN)과 최소한의 침입탐지시스템의 기능을 탑재한 것이고, NetScreen-10/100은 침입차단시스템에 가상사설망과 일부공격탐지 기능을 갖추고 있다[5].

2)Interoperational 모델

Interoperational 모델은 미리 정해진 프로토콜을 통하여 개별 보안시스템간의 상호 작용 및 통합이 이루어지는 모델이다[1]. 보안시스템간의 연결구조는 각 보안제품이 독립적으로 운영되면서 상호연동을 통하여 보안기능을 수행하는 구조이다. 이 구조에서는 각기 다른 보안제품이 직접 연결되어 있으며 특별한 이벤트 발생시 상호 정의된 규칙에 따라 동작할 수 있다. 예를 들어 침입탐지시스템에서 침입이 탐지되었을 경우 해당되는 출발지 주소에 대한 접속 거부 등의 규칙을 침입차단시스템에서 설정하도록 하는 것이다. 이 구조는 두 객체(Object)간에 관리자가 개입되지 않고 동작하는 구조로 별도의 관리자 인증 없이 객체 인증만을 통해서 동작하게 된다.

3)Broker 모델

Broker 모델은 개별 보안시스템 간의 상호 연동 및 통합이 브로커를 통해서 이루어지는 모델이며 개별 시스템은 자신에 탑재되는 에이전트와 연동에만 집중할 수 있다[1]. 이 모델은 다수의 보안 제품이 하나의 서버에 연결되어 있으며, 하나의 서버가 모든 통제를 하도록 구성되어 있다. 이와 같이 구성함으로써 각각의 보안시스템은 통합관리로 인해 발생되는 오버헤드를 서버로 분산 할 수 있게 된다. 또한 통합보안관리를 위해 서버는 데이터수집, 저장 및 분석작업을 항상 수행하게 되며 매니저는 필요한 경우 서버로 접속하여 그 상황과 결과를 모니터링 할 수 있게 된다[10].

그림1은 대규모 조직에 적합한 계층적 구조의 통합보안관리시스템의 논문에서 제안한 브로커(Broker) 모델을 기반한 통합보안관리 시스템이다.

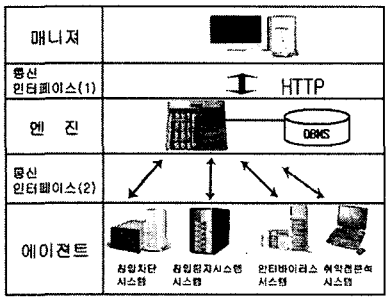

그림 1 통합보안관리시스템의 구성도

2.3 보안 관제 서비스

초창기의 보안 관제 서비스는 정보보호 시스템을 원격지에서 관리하여 외부의 침입으로부터 보호하는 것을 의미했으나, 현재의 보안 관제 서비스는 서버와 네트워크 장비, 보안 시스템 등을 원격지에서 안전하게 관리하며 인가 받지 않은 외부 침입자를 실시간으로 탐지하여 원격 관제실에 경보를 보냄으로써 외부 침입에 즉각적인 대응 및 역추적을 할 수 있을 뿐만 아니라 이 기종간에 보안 관리를 할 수 있는 보안 통합 개념의 관제 시스템으로 발전하고 있다.

이러한 시스템은 고객의 시스템에 에이전트를 설치하고 관제실에서 24시간 실시간으로 해커의 침입을 감시하고 대응하는 모니터링 서비스 형태로 자리를 잡아가고 있으며, 국내에서는 일부 보안 업체를 중심으로 보안 관제 시스템을 순수 국내기술로 개발하고 있으며, 자체적으로 운영중이거나 계획 중에 있다.

그림2는 사이버패트롤 관제 서비스의 개념을 나타낸 그림이다.

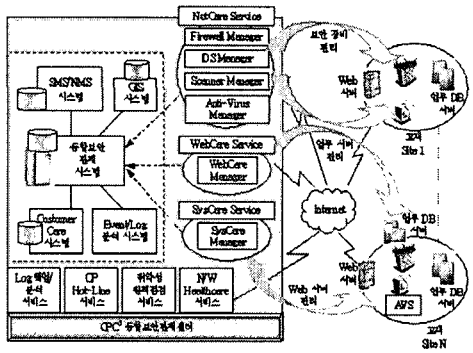

그림 2 사이버패트롤 전체서비스 개념도

사이버패트롤 관제서비스를 통해 불법 해킹을 비롯한 각종 침해 사고에 대한 실시간 대응, 보안 전문의 인력의 24시간, 365일 정보 보호 대응 체계 구현, 렌탈 방식 비용 지불로 인한 고객 부담 최소화, 다양한 비즈니스별 SLA(Service Level Agreement)을 통한 엄밀하게 정의된 최적의 서비스를 제공할 수 있다고 한다.

이제 인터넷과 전자상거래의 급속한 확산과 함께 불법 해킹이나 바이러스 등으로 인한 시스템과 네트워크 자원의 손상을 막을 수 있는 해결책 중의 하나로서 종래 보안 시장의 단순 제품 위주의 솔루션 중심에서 현재는 종합적인 보안 서비스 중심으로 변해가고 있는 추세이다. 따라서 정보 보호 서비스 업체는 정보 보호 컨설팅 및 교육에서부터

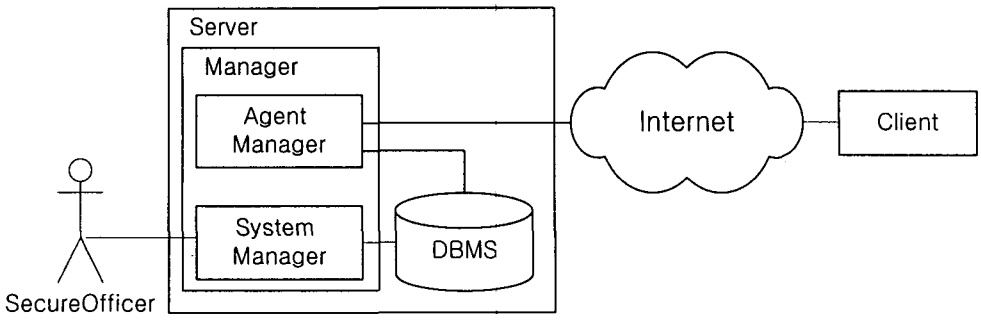

그림 3 원격통합관리시스템

시스템 구축 및 관리까지 보안과 관련된 제반업무를 대행해 주고 있으며, 정보 보호 컨설팅을 비롯한 취약점 점검 및 리포팅, 시스템 점검, 데이터 복구, 통합 보안 관리 시스템 운영 등의 종합 정보 보호 서비스로 발전해가고 있다.

보안 관제 서비스는 현재 분리 운영되어온 침입 차단 시스템과 침입 탐지 시스템, 바이러스 백신 서버 등을 연동하여 인가 받지 않은 불법적 외부 침입자가 탐지되면 이를 실시간에 자동 차단 및 역추적하는 기능까지 내장하고 있어 개별 회사나 기관이 보안 관리를 아웃소싱 할 경우 현장 보안 담당자를 따로 두는 것보다 훨씬 빠른 대응 능력을 갖게 될 것으로 보인다.

국내 보안 관제 서비스 시장은 2001년을 기점으로 코코넛, 해커스랩, 이글루 시큐리티, 한시큐어, 사이버패트롤 등의 5개 업체로 구도가 재편되었으며, 2002년 2월 26일 한시큐어가 코코넛으로 합병됨으로써 코코넛이 국내 최대 보안 관제 서비스 업체로 떠오르고 있다.

그러나 국내 보안 관제 서비스 업체의 주된 고객 IDC정도이며, IDC 내에서 보안 서비스를 받고 있는 업체의 비율도 10%를 크게 밑돌고 있는 것으로 나타나 앞으로 꾸준한 증세를 보인다 하더라도 국내 시장의 한계성과 원격 보안 관리 및 보안 아웃소싱이 성숙되지 않은 국내 상황을 고려해 볼 때 새로운 시장을 개척하기 위해 해외 진출은 또 다른 돌파구가 될 수 있을 것으로 보인다.

비교적 보안 부문에서 선진 기술을 가진 미국도 자사 보안 제품에 국한해 원격 관리를 시행해왔으며 이제 이 기종 보안기기에 대한 통합 기술이 개발되고 있는 추세이므로 많은 국내 업체들이 보안 컨설팅 서비스, 아웃소싱과 함께 패키지 형태의 보안 관제 서비스를 제공할 것으로 기대된다.

차후에 보안 관제 서비스 분야가 국내 뿐 아니라 세계 시장에서도 경쟁력을 가지고 성장하기 위해서는 국가 차원에서 이 기종 보안 장비들을 동시에 관리할 수 있는 보안 관제 엔진 개발을 추진하는 것이 필요하다[4].

3. 원격통합관리시스템

본 장에서는 논문에서 제안한 원격통합관리 시스템의 구성에 대해서 알아보고, 각 모듈의 특성에 대해서 살펴본다. 그림3은 본 논문에서 제안한 원격통합관리 시스템을 보여준다.

시스템의 주된 목적은 그림에서 보여준 바와 같이 보안 전문가의 정책을 각 리눅스기반 서버에 원격으로 적용할 수 있도록 제공하는 것이다. 즉, 특별한 보안관련 프로그램이 업데이트되었다면 이러한 사실을 서버에 등록하고, 등록된 서버로부터 서비스 받고 있는 클라이언트들은 자신의 Agent를 이용하여 새로운 변경사항을 설치하고 변경한다.

이러한 처리과정은 다음과 같이 이루어진다. 먼저 보안전문가가 새로운 정책이나 프로그램 업데이트 요청을 SystemManager

(SM)를 통하여 처리한다. SM은 보안 전문가의 요구 사항을 처리한 후 그 결과를 DBMS에 저장하고 클라이언트가 AgentManager에게 새로운 데이터 요구를 위임하는 형태를 취한다.

3.1 주요 모듈

본 절에서는 시스템을 구성하는 각 모듈의 역할에 대해서 살펴본다.

3.1.1 SecureOfficer

서버에 데이터 업데이트를 지시하는 보안 지식이 풍부한 시스템 관리자. 즉, 이 보안 관리자들의 보안 지식이 곧 그 시스템의 보안 레벨이라 볼 수 있다.

3.1.2 Manager

Manager 모듈이 원격통합시스템의 가장 주된 모듈로서 데이터를 관리하고 처리하는 모듈이다. 이 모듈은 두 개의 모듈, 즉, AgentManager와 SystemManger로 구성된다.

3.1.2.1 System Manager

SystemManager(SM)은 그림 4에서 보여주는 바와 같이 다시 DM, PM, AM, CM 으로 구성된다.

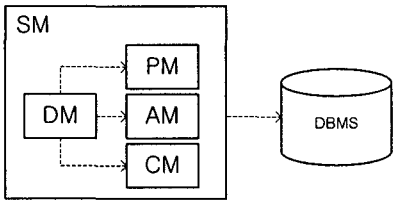

그림 4 System Manager

- **DM(DataModule)** : 새로운 어플리케이션, 보안 정책, 어플리케이션 설정을 등록할 AM(ApplicationModule), PM(PolicyModule), CM(ConfigureModule)과 연결해주는 역할을 한다. 전체적인 기능은 Function 1과 같다.

```
[Function 1] DATAMODULE
DataModule() {
 switch(InputDataClass) {
  case Application :
   ApplicationModule();
  case Policy :
   PolicyModule();
  case Configure :
   ConfigureModule();
  }
}
```

- **AM(ApplicationModule)** : 새로운 어플리케이션을 서버에 저장하거나 업데이트하는 역할을 한다. 기능은 Function 2와 같다.

```
[Function 2] APPLICATIONMODULE
ApplicationModule()
{
  if(inputdataname==DbApplictionName) {
    updateDB(inputdata);
  }else {
    insertDB(inputdata);
  insertRelationApplicationName();
  insertRelationValue();
 }
```

- **CM(ConfigureModule)** : 어플리케이션에 관련된 설정을 변경하거나 추가하는 일을 한다. 처리 과정은 Function 2와 비슷하다.

- **PM(PolicyModule)** : 어플리케이션이나 새로운 침입에 대한 보안 정책을 변경하거나 추가하는 역할을 한다. AM과 마찬가지로 Function 2와 비슷한 수행을 한다.

3.1.3 Agent Manager

그림5에서 보여주는 바와 같이, Agent Manager(AM)는 클라이언트의 요구에 따라 DMBS에서 필요한 정보를 수집하는 역할을 한다.

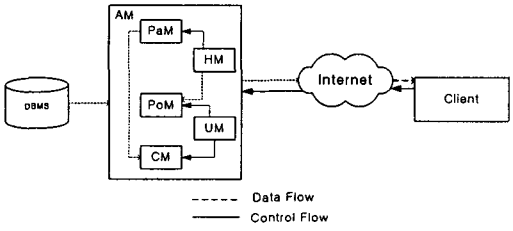

그림 5 Agent Manager

- **HM(Host Module)** : 서버에 등록되어 있는 클라이언트들의 정보를 관리하는 모듈로 HostDB에 정보를 저장하고 검색하는 역할을 한다. Function 3과 같이 동작한다.

```
[Function 3] HOSTMODULE
HostModule() {
 switch(quest) {
  case : HostRegistration
   insertDbHost();
   HApplicationModule();
  case : HostHWInfo
   selectDbHostHWInfo();
  case HostID
   selectHostID();
  }
}
```

- **UM(Update Module)** : UM은 클라이언트의 요청에 대한 데이터를 PoM과 CM에게 받아 전달해주는 역할을 한다.
- **HaM(Host Application Module)** : HM에 의해서 각 클라이언트들이 가지고 있는 어플리케이션들을 관리하는 하는 모듈로 HApplicationDB에 데이터를 관리하는 역할은 한다. 절차는 Function 4와 같다.
- **PoM(Policy Module)** : UM에 의해서 Agent에서 요구하는 Application에 관련된 보안 Policy를 검색하여 전달해주는 역할을 한다. 절차는 Function 5와 같다.
- **CM(Configure Module)** : Agent의 어플리케이션의 설정사항에 대한 요구를 처리한다. CM은 UM에 의해서 제어된다.

```
[Function 4] HAPPLICATIONMODULE
HApplicationModule() {
 switch(quest) {
  case Update :
   UpdateData();
  case Insert :
   InsertData();
  case Select :
   SelectData();
  }
}
```

```
[Function 5] POLICYMODULE
PolicyModule() {
 if(newAnyPolicy())
   pushAgent();
 if(AppID == HApplicationID)
   pushAgent();
}
```

3.1.4 Client

클라이언트의 역할은 그림 6과 같다. 클라이언트에 있는 Agent가 주기적으로 서버로부터 정보를 업데이트하여 각 어플리케이션에 적용한다.

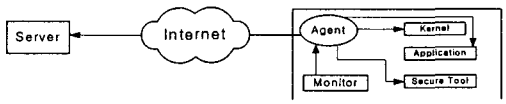

그림 6 Client의 구성도

- **Agent** : 각 클라이언트에서 동작하는 프로그램으로써 서버와 주기적인 통신을 통하여 서버의 정책과 응용프로그램의 상태를 클라이언트에 업데이트 하는 역할을 한다. 클라이언트 상에서 프로그램의 설치 및 관리기능을 가지고 있는 만큼 강력한 보안기능이 요구된다. 절차는 Function 6과 같다.

```
[Function 6] Agent
Agent() {
 switch(quest) {
   case : HostRegsitation
        SeverHostReg();
   case : RecvData
        RecvDataInstalled(); }
 if(nowTime == UpdateTime) {
   QuestUpdateData();
 }
}
```

3.2 데이터베이스

본 절에서는 제안한 시스템의 데이터 베이스 구조와 각 테이블의 역할에 대해서 살펴본다. 그림7은 본 시스템의 데이터 베이스 구성을 보여준다. ApplicaitonTbl과 HostTbl을 중심으로 PolicyTbl과 ConfigureTbl은 어플리케이션의 종류에 따라 결정되며 HostPolicyTbl과 HApplicatiobTbl은 HostTbl에 의존관계를 가진다.

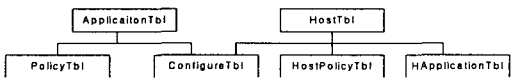

그림 7 데이터베이스 구성도

- **ApplicationTable** : 어플리케이션들을 총괄하는 데이터베이스로 현재 서비스 중인 패키지는 여기에 모두 등록된다.
- **PolicyTable** : 각 보안관련 툴이나 어플리케이션의 보안 설정에 관한 정보를 저장하는 테이블로서, 각 각의 어플리케이션별로 정책을 나누어 관리한다.
- **ConfigureTable** : 각 어플리케이션의 환경설정을 가지고 있는 테이블이다. 이 구조는 여러 테이블 중에 가장 복잡한 구조와 가장 많은 조합을 가진다.
- **HostTable** : 서비스하고 있는 클라이언트와 Agent에 관한 정보를 저장하는 테이블이다.
- **HApplicationTable** : 각 클라이언트에서 작동하고 있는 HApplicationList와 정책을 저장하는 테이블이다.
- **HostPolicyTable** : 각 클라이언트에서 적용되고 있는 PolicyList를 저장하는 테이블이다.

3.3 원격통합관리시스템의 주요정책

본 절에서는 제안하는 시스템의 주요정책에 대하여 기술한다. 시스템의 정책은 데이터 업데이트 및 클라이언트 관리 정책으로 구성한다.

3.3.1 데이터 업데이트

데이터 업데이트 정책은 방송(Broadcast) 형태를 취하지 않고 Agent의 요청에 의한 업데이트 방법을 사용한다. 그 이유는 침입자에 의한 서버 위장을 방지하고 서버의 과부화를 방지하기 위함이다. 서버는 Application, Policy, Configure 데이터를 가지고 다음과 같은 처리를 한다.

Application 등록

SecureOfficer가 DM에게 새로운 어플리케이션 등록을 알린다. 그러면 DM은 AM에게 Officer가 등록하고자 파일을 전달하여 DB에 등록하고 필요한 설정과 정책을 세운다.

Policy 등록

Application 등록 절차와 비슷하며 DM이 PM에게 새로운 정책의 추가나 변경에 대하여 알린다.

Configure 등록

Application 등록 절차와 비슷하며 DM이 CM에게 어플리케이션 설치에 관련된 정보를 전송한다.

3.3.1.1 데이터 업데이트 시나리오

서버의 정보에 대한 업데이트에서 보안정책은 하나의 정책을 어플리케이션별로 분할하

여 저장하고 클라이언트의 요구에 의해서 정책 파일을 조합하여 생성한다. 데이터 업데이트를 위한 시나리오는 다음과 같다.

단계 1 : Agent가 AM에 새로운 Data변경 정보를 요구

단계 2 : AM은 HM에게 Data변경 관련 처리를 요구

단계 3 : HM은 PaM에서 관련 Host의 H-ApplicationList를 UM에게 넘겨 줄 것을 요청

단계 4 : HM은 Host정보를 UM에게 전송

단계 5 : UM이 Update되어야 할 리스트를 작성하여 Agent에게 전송

단계 6 : UM은 Agent가 관리자의 결정을 가지고 올 때까지 대기

단계 7 : 어플리케이션의 변화에 대한 정보를 작성하고, 각 어플리케이션과 관련된 설치 관련 Configure파일들을 조합하여 변경정보 생성

단계 8 : 각 어플리케이션의 설치관련 Configure파일을 및 제어관련 설정을 조합하여 클라이언트에 전송

단계 9 : Agent는 설치관련 Configure에 따라 새로운 설정 및 변경을 클라이언트에 적용

단계 10 : 보안 어플리케이션별로 보안 정책 변경사항을 검색하여 처리

단계 11 : Agent는 모든 일이 끝나면 모니터 프로그램에게 변경 사항에 대하여 보고

3.3.2 Client 등록

클라이언트등록은 각 클라이언트의 Agent에 의해서 수행된다. 클라이언트를 서버에 등록할 때, 그 시스템의 하드웨어정보도 같이 등록한다. 등록 시나리오는 다음과 같다.

Agent가 서버에 등록

Agent는 서버의 AM에게서 AgentID를 받고 AM이 HM를 통해서 HostDB에 등록하고 패키지 정보를 PaM에게 주어서 HApplicationDB에 패키지리스트를 저장하게 함

Agent에 의한 클라이언트 변경

클라이언트의 정보를 최신의 정보로 유지하기 위해서 다음과 같은 작업단계를 수행한다.

단계 1 : Agent가 등록됨과 동시에 AM은 PaM으로부터 HApplicationList를 UM에 전송

단계 2 : UM은 HApplicationList에 있는 버전 정보와 AppDB에 있는 어플리케이션의 버전 정보를 비교하여 새로운 어플리케이션을 선택해서 Agent에게 전송

단계 3 : UM은 HM에게 Host H/W정보와 HApplicationList를 CM에게 넘겨주면 CM은 각 어플리케이션 별로 H/W정보를 참조하여 설정파일을 만들어서 Agent에게 전송

단계 4 : Agent는 새롭게 받은 어플리케이션을 설정파일에 따라 설치

단계 5 : AM은 HApplicationList를 PoM에 주고 처음 설치하는 것을 알림 PoM은 각 어플리케이션에 관련된 보안 정책과 기본 정책 파일을 조합 생성한 후 파일을 Agent에게 보냄

단계 6 : 모든 어플리케이션 업데이트와 보안 정책이 적용되면 SecureOfficer가 그 Host에 맞게 보안 정책에 사용되는 변수를 설정

4. 결론

본 논문에서는 리눅스 기반의 서버에 효율적인 보안정책적용을 위한 원격 통합 관리 시스템을 제안하였다. 제안한 시스템은 하나의 시스템에서 여러 어플리케이션의 업데이트와 보안정책적용을 여러 시스템에 적용할

수 있는 시스템이다. 보안전문가의 어플리케이션의 등록, 그리고 거기에 맞는 보안 정책을 여러 보안 툴에 맞게 나누어 데이터베이스에 저장한다. 그러면 AgentManager가 클라이언트 요청에 의해서 새롭게 업데이트된 어플리케이션을 제공하고 보안 정책 또한 새로운 어플리케이션에 관한 것이나 새로운 정책을 그 클라이언트에 맞게 조합하여 제공하고 있다. 제안한 시스템을 통하여 시스템관리자가 보안전문가가 아니더라도 효율적인 시스템 보안을 제공 할 수 있을 것으로 기대되었다.

향후 연구에서는 원격통합관리시스템의 성능 검증과 다양한 운영체제 적용에 대한 연구가 필요하다. 이러한 연구는 시스템의 보다 이식성이 높고 보다 효율적인 원격통합관리시스템을 구축하기 위함이다.

참고 문헌

[1] 안혜원, "통합보안관리시스템", 제7회 정보통신망 정보보호 워크숍, 2001.

[2] 서울대 중앙교육연구전산원, 전산망 보안 기록 정보 수집 및 관리를 위한 기본 시스템 개발에 대한 연구, 한국정보보호센터, 1997.

[3] 김현배, 이동 에이전트를 이용한 보안 관리 시스템 모델, 정보교육학회논문지, 1998.

[4] 김명은 외, 국내 보안 관제 서비스 기술 동향, http://new.itfind.or.kr/KIC/etlars/industry/jugidong/1045/104501.htm.

[5] 박준홍 외, 내규모 조직에 적합한 계층적 구조의 통합보안관리시스템에 관한 연구, CISC2001, 2001.

[6] Gerhard Mourani, Securing and Optimizing Linux: The Ultimate Solution, OpenNA, 2001.

[7] 이만영 외, 전자상거래 보안 기술, 생능출판사, 1999.

[8] 9개 전문업체 출사표, 정보보호21c, 2002. 1.

[9] Eric Maiward, Network Security: A Beginner's Guide, McGrow-Hill, 2001.

[10] 이동영 외 4명, "SNMP를 이용한 웹 기반의 통합보안관리시스템", 한국정보처리학회 추계학술 논문집, 1998