통합보안관리시스템의 방화벽정책 분배를 위한 알고리즘 : FALCON

김 광 혁[†] · 권 윤 주[††] · 김 동 수[†] · 정 태 명[†††]

요 약

근래의 네트워크는 인터넷의 여러 보안 위협과 시스템의 취약성들로 인해 보안시스템의 도입, 보안 컨설팅 등을 통하여 시스템 보안에 많은 노력을 기울이고 있다. 통합보안관리시스템은 다수의 보안시스템 및 방화벽으로 구성된 보안 영역을 설정하고, 각각의 보안 제품들에 대해 일관성 있는 보안정책을 적용하는 시스템이다. 통합보안관리를 위해서는 적절한 정책과 더불어 정책을 적용할 보안시스템의 선정이 기본이 된다. 특히 통합보안관리시스템에서의 방화벽은 하나의 패킷흐름에 대해 패킷 경로에 놓인 여러 개의 방화벽 시스템이 관여를 하게 된다. 본 논문에서는 다수의 방화벽으로 구성된 통합보안관리시스템에서 접근제어 정책을 수행할 방화벽을 선정함에 있어서 문제점을 살펴보고 정책을 설정할 방화벽 선정 알고리즘인 FALCON 알고리즘을 제시한다. FALCON 알고리즘의 사용으로 방화벽 선정에 있어서 안정성, 확장성, 간결함 등의 이점을 기대할 수 있다.

The Policy Distribution Algorithm of Firewall in Integrated Security Management

Kwang H. Kim[†] · Yun J. Kwon[††] · Dong S. Kim[†] · Tai M. Chung[†††]

ABSTRACT

Recently, Networks are required to adopt the security system and security consulting because of security threats and vulnerabilities of systems. Enterprise Security Management (ESM) is a system which establishes the security zone composed of security systems and Firewalls and applies the security policy to each security system. A relevant ESM is based on the effective policy and the proper security system. Particularly, multiple firewalls in ESM are concerned with the security policy about each traffic. In this paper, we describe the problems that can be occurred when we select the firewalls to apply security policy of access control in ESM composed of multiple firewalls and propose the FALCON algorithm, which is able to select the firewalls to apply the policy. We expect that FALCON algorithm offers stability, scalability and compactness for selecting firewall set.

키워드 : 통합보안관리(ESM), ISMS, 방화벽(Firewall), FALCON

1. 서 론

근래의 정보화 추세는 말단의 호스트까지 인터넷에 연결되어져 네트워크를 구성하고, 개인 및 기업의 정보들이 개인과 기업의 존립에 커다란 영향을 끼칠 정도로 중요시 되고 있다. 반면에 인터넷을 통한 불법적인 해킹시도의 급격한 증가와 불특정 다수를 대상으로 하는 바이러스, 악성코드들의 증가로 개인과 기업의 존립에 큰 위협이 되고 있으며, 실제로 그 피해 사례도 해마다 증가하고 있는 실정이다. 최근 CERT-CC의 해킹동향을 살펴보면 해킹 도구의 자동화, 지능화로 해킹 시도, 바이러스, 웜 등이 급격히 증가하고 있으며 이에 따른 침해사례가 상당수에 이르고 있다[12]. 해킹은 정보 유출, 금전적 손실 유발등으로 기업의 존폐에 막대한 영향을 미칠 수도 있으며, 바이러스, 웜등은 SANS의 Nimda worm의 분석보고서에서도 알 수 있듯이 단일 호스트가 목표시스템이 아닌 네트워크를 목표시스템으로 하고 있어 네트워크 관리와 보안에 많은 어려움을 더해주고 있다[13]. 이에 효과적으로 대응하기 위해 엔터프라이즈 네트워크에서는 일찍이 방화벽을 비롯한 침입탐지시스템, 가상사설망, 취약점분석시스템등을 도입하여 이러한 공격으로부터 정보들을 보호하고자 노력해 왔다. 그러나 다수의 보안시스템의 사용은 제각기 다른 사용자 인터페이스와 기능을 가져 많은 인력과 사용자에게 고도의 관리지식을 필요로 하고 있으며, 가장 큰 문제로는 보안 정책의 일관성 유지에 어려움을 더해주고 있다는 것이다. 최근의 보안 시장은 이런 어려움을 겪고 있는 엔터프라이즈 네트워크 환경에서 통합보안관리를 수행하는 ESM이 주목을 받고 있다[4]. ESM은 제각기 다른 사용자 인터페이스와 정책설정 방식을 일관화시키고, 보안 제품들을 대상으로 중앙 집

† 준 회 원 : 성균관대학교 대학원 정보통신공학부
†† 정 회 원 : 한국과학기술정보연구원
††† 종신회원 : 성균관대학교 정보통신공학부 교수
논문접수 : 2002년 2월 8일, 심사완료 : 2002년 7월 12일

중식 정책 제어 방식을 통하여 일관성 있고, 전사적인 정책을 수립하여 집행할 수 있는 이점을 제공한다. 그러나 다수의 보안시스템의 통합에는 이기종간의 인터페이스 호환, 환경 및 정책설정 인자의 다양성, 다양한 실행 플랫폼 등의 해결해야 할 제반 문제들이 존재하고 있다. 그 중 보안정책을 중앙집중 관리하면서 발생하는 문제점들이 보안관리라는 시스템 측면에서는 크게 대두되고 있다.

본 논문에서는 현재 개발하고 있는 ESM 프로젝트인 Integrated Security Management System(이하 ISMS)을 기반으로 기술이 이루어질 것이며, ISMS중 보안정책의 설정시에 발생하는 모호성의 발생원인과 해결책을 제시하도록 한다. 이 문제는 ISMS가 방화벽의 통합관리에서 빚어지는 문제인 다수의 방화벽으로 구성된 네트워크에서 접근제어 정책의 할당시 정책의 적용대상을 추출하는 알고리즘을 제안함으로써 모호성 문제를 해결하고자 한다.

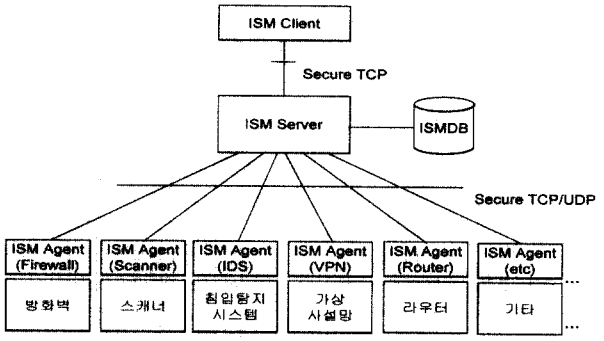
본 논문의 구성은 다음과 같다. 제2장에서는 ISMS의 필요성과 구성, 기능 등을 알아보고, 제3장에서는 ISMS에서 방화벽의 기능과 보안 정책의 설정과정에서 발생하는 보안 설정대상의 모호성 문제에 대해 기술하도록 한다. 제4장에서는 방화벽의 정책 설정과정에서 발생하는 모호성을 해결하기 위한 알고리즘을 제안하며, 제5장에서 결론과 향후과제에 대해 언급하도록 하겠다.

2. ISMS의 구성과 기능

앞서 기술한 ISMS의 필요성을 다음과 같이 요약 할 수 있겠다. 첫째, 각종 보안시스템의 도입으로 인한 전사적인 보안정책 확립 필요성 인식, 둘째로 이기종간의 보안 제품의 관리 효율 극대화, 셋째로는 관리 운영의 비용 절감이다. 비록 ISMS를 이용하여 전사적인 보안정책 운용, 이기종간의 통합관리가 필요하지 않을 만큼 대규모 네트워크가 아니다 할지라도 ISMS가 사용 될 수 있다[9]. 날로 복잡해지고 위험수위도 높아지는 보안 위협과 같이하여 보안 위험관리 역시 관리 기술의 고도화, 전문 인력 등의 채용으로 인해 고비용을 요구함에 따라 최근 기업들은 이러한 운영상의 기술 및 인력들을 아웃소싱을 통하여 해결하고자 하고 있다. 보안관제 아웃소싱 업체는 각 회사에서 사용하고 있는 보안제품들을 효율적으로 운영하고, 제어하기 위해 ESM을 반드시 필요로 하고 있다. 그렇지만 현재의 ESM 시장은 사용자와 정책관리 위주의 제품, 혹은 취약점과 위험요소 분석에 주안점을 두고 있는 제품군으로 나누어져 효과적인 위험 분석으로부터 보안 정책의 제어를 통해 적절히 보안 위험 요소를 해결한다고 볼 수 없다[4].

현재 개발중에 있는 ISMS는 취약점과 위험요소의 분석과 함께 정책제어를 할 수 있는 ESM을 구현하고 있다.

(그림 1)은 ISMS의 구성을 나타내었다. 취약성 분석을 위한 취약점분석도구, 보안정책을 통해 접근제어를 수행할 방화벽, 가상사설망, 실시간 침입탐지를 위한 침입탐지시스템, 네트워크 구성정보 획득을 위한 라우터, 기타 스위치, 개별 호스트 접근제어를 수행하는 TCP Wrapper등이 ISMS의 주된 보안장비들이다.



(그림 1) ISMS의 구성

ISMS의 구조는 ISM Client, ISM Server, ISM Agent, ISMDB의 4개의 부분으로 구성되어 있다[3]. ISM Agent들은 대규모 네트워크 상에 분산 설치되어 있는 이기종의 보안 시스템에 각각 설치되어 로그, 실시간 Event등의 보안 관리 정보들의 수집과 ISM Server로부터 전달 되어지는 보안 정책을 각 시스템에 할당하는 기능을 수행한다. ISM Server은 관리 대상 네트워크의 보안 시스템들을 통합 관리를 위해 사용자 관리, 정보가공과 제어를 위한 ISM 데이터베이스 운용, 각 ISM Agent 들과의 연동을 수행한다. 수집된 정보는 ISM Client에게 제공되어 보안 정책 설정 근거 자료로 사용되어진다. 보안 관리자는 이 정보를 바탕으로 보안 정책을 설정하며 ISM Server을 통하여 해당 ISM Agent에게 전달한다. ISMDB(ISM Database)는 사용자 정보, 보안 시스템의 구성 정보, 보안 정책, 성능 정보 등으로 구성되어지며 보고서등의 결과를 만들어내기 위해 실시간 탐지 이벤트, 시스템 로그 등의 정보를 유지하고 있다.

ISMS는 이종의 운영 시스템, 서로 다른 보안 제품 환경에서 각 시스템들에 대해 투명성 있게 동작하고 보안 정책의 일관성을 유지할 수 있다. ISMS를 구축함으로써 얻는 주된 이점은 다음과 같다.

- 효율적이고 정책 지향적인 보안 관리
- 각 시스템 보안 정책의 일관성 유지
- 보안 시스템들의 통합 모니터링
- 실시간으로 발생되는 보안 이벤트의 적기 대응

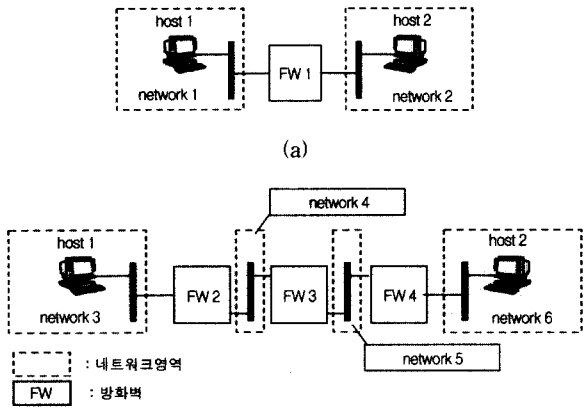
다음 장에서는 ISMS에서 접근제어 기능을 수행하는 방화벽과 관련해 각 방화벽으로 접근제어 정책의 배분시 생기는 문제점과 원인을 알아본다.

3. ISMS상에서의 방화벽

인터넷상의 모든 보안 위험 요소들을 적절히 해결할 수 있는 제어 시스템이 존재하지 않지만, 방화벽은 여러 가지

면에서 다른 제어 시스템보다 유용한 기능을 제공해 주고 있다. 방화벽은 두 개 이상의 네트워크 사이에서 정책에 따라 들어오거나 나가는 연결 혹은 서비스에 대해 거부하거나 허용해주는 기능을 수행한다[1]. 인터넷으로 연결되어 있는 각 호스트들은 인터넷으로부터의 위협에 노출되어있거나 자체 취약점을 가지고 있지만 보안 권고안대로 유지 보수한다는 것은 매우 많은 비용과 노력을 필요로 한다. 또한 드러나는 취약점과 위협의 출현 속도는 보안 관련 패치의 배포 속도를 넘어서고 있는 실정이다[8]. 그래서 모든 호스트에 대해 모든 보안 관련 패치를 실행한다는 것은 사실상 불가능하게 보이기까지 한다. 이러한 상황에 방화벽은 적은 비용으로 좋은 효과를 얻을 수 있는 시스템으로 꼽힌다[1, 2]. 방화벽은 많은 보안 시스템 중에서 정책을 기반으로 하는 제어 시스템으로 소규모의 네트워크부터 대규모의 네트워크에 이르기까지 널리 사용되고 있다. ISMS에서도 역시 ISMS관리 영역의 네트워크 접근 제어 수행을 위해 방화벽을 사용한다.

본 논문에서 언급하고 사용하는 방화벽은 패킷 필터링 방화벽을 사용하는 것을 전제로 한다[1, 6]. 방화벽은 보안 결정과 정책에 대해 두가지의 기본적인 방침을 사용하는데 거부가 기본인 방침(Default Deny Stance)과 허용이 기본인 방침(Default Permit Stance)이 그것이다[1, 2]. 보안시스템의 운영이라는 관점에서 볼 때 디폴트가 거부인 방침이 필요한 서비스에 대해 보안문제를 충분히 고려한 후에 정책을 적용시킬 수 있으므로 더 견고한 보안정책을 설정 할 수 있다[5]. 현재 많은 방화벽 제품들은 이 방식을 채택하고 있으며 ISMS에서도 관리대상 방화벽들에 대해 디폴트가 거부인 방침을 취하고 있다. 따라서 ISMS를 사용시에 초기 설정은 모든 패킷경로가 거부로 되어있는 상태가 되며 ISMS의 각 보안 시스템의 정책 설정을 위해서는 ISM Server로부터 각 보안시스템으로 흐르는 패킷경로에 대한 접근제어 정책이 먼저 방화벽에 정의되어 있어야 하겠다.



(그림 2) 방화벽과 네트워크 영역

패킷 필터링 방식의 방화벽은 패킷의 근원지 및 목적지 주소와 포트번호, 프로토콜 종류 등의 정보를 근거로 해서 전송을 제어 할 수 있게 해준다[7, 10]. 정책 설정의 피대상은 근원지 주소와 목적지 주소로 명기된 호스트 혹은 네트워크가 된다. (그림 2)의 (a)에서 호스트1로부터 호스트2에 대한 텔넷 서비스를 허용한다고 할 때, (a)의 경우 방화벽1에 다음과 같은 정책을 내릴 수 있다.

FW	출발지	목적지	프로토콜	출발지포트	목적지포트	서비스	action
1	host 1 / 32	host 2 / 32	tcp	any	23	telnet	permit

(b)의 경우는 호스트1은 방화벽2에 의해 네트워크 영역이 구분되는 네트워크3에, 호스트2는 방화벽4에 의해 네트워크 영역이 구분되어지며 네트워크6에 위치하고 있다. 근원지 주소와 목적지주소에 기인하여 방화벽2와 방화벽4에 다음과 같은 정책을 각각 내린다고 하자.

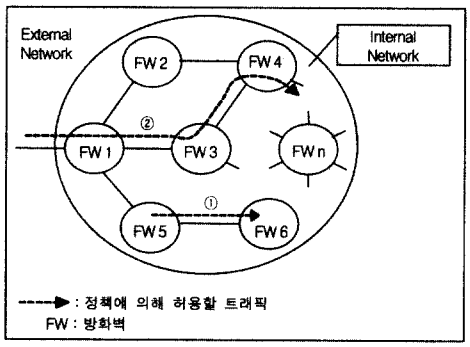
FW	출발지	목적지	프로토콜	출발지포트	목적지포트	서비스	action
2	host 1 / 32	host 2 / 32	tcp	any	23	telnet	permit
4	host 1 / 32	host 2 / 32	tcp	any	23	telnet	permit

허용하고자 하는 패킷경로는 방화벽2는 통과하겠지만 호스트1에서 호스트2로 가는 경로 상에 놓여진 방화벽3에 의해 거부가 되어지며 결국 패킷경로는 허용되지 못한다. 패킷경로를 허용하기 위해서는 방화벽3에 대해서도 같은 정책을 내려줘야 한다. 이렇게 다수의 방화벽이 존재하는 상태에서는 패킷경로상에 위치한 방화벽에 모두 정책을 설정해주어야 한다.

(그림 3)은 ISMS의 임의의 네트워크를 방화벽의 네크워크 인터페이스 정보를 이용하여 그래프로 표현한 것이다.

(그림 3)의 그래프위에 허용 정책을 필요로 하는 패킷경로를 점선으로 표시하였다. 방화벽의 접근제어 정책은 호스트의 근원지 및 목적지 주소를 기본으로 하고 있다. 패킷경로①은 방화벽5를 지나 방화벽6을 통과하는 패킷경로를 표시한 것이다. 패킷경로①과 관련한 방화벽 정책을 내리고자 할 경우에는 패킷경로①이 거치는 모든 방화벽, 즉 방화벽5와 6을 추출하여 정책 적용의 대상으로 삼는다. 패킷경로②의 경우에는 근원지 및 목적지 네트워크와 직접적으로 관련이 있는 방화벽1, 방화벽4 이외에 두 방화벽의 경로 상에 있는 방화벽3까지 정책설정 대상에 추가해야 한다. 또한, 방화벽2로 우회하는 경우도 고려하여 방화벽2에도 정책을 설정해야 하는데 이것은 방화벽3 방향의 네트워크 장애로 패킷경로가 방화벽2로 우회하여 라우팅 될 경우를 위해서인데 신뢰성 있는 패킷 전달을 위해서는 잠재적인 대체경로에 위치한 방화벽에도 정책을 내려야 한다. ISM Client는 사용자에게 네트워크 구성을 시각화하여 관리자는 허용하고자 하는 패킷경로에 위치한 방화벽을 직접 찾아 낼 수 있다. 그러나 각 방화벽 정책 설정에 대해 관리자의 방화벽 선정은 정책 설정의 오류를 유발할 수 있다. 방화벽 선정의 오류는 올바른 패킷 흐름을 막을 수도 있지만, 최악의 경우 네트워크 침해를 유

발 할 수 있는 취약점을 만들 수도 있다. 이로 인해 정책설정에 있어서 방화벽 선정은 관리자에 의존하기보다는 방화벽 선정 알고리즘을 이용하여 정확성을 기대할 수 있어야 한다. 따라서 앞서 살펴본 프로세스를 자동화함으로써 사용자의 실수를 방지하여 정확성을 기하고, 관리자에게 네트워크 구성요소에 대한 고려 없이 근원지와 목적지 호스트 혹은 네트워크 정보만을 이용하여 정책을 설정 할 수 있는 투명성 및 편의성의 제공은 ESM 시스템의 필수적 기능이다.



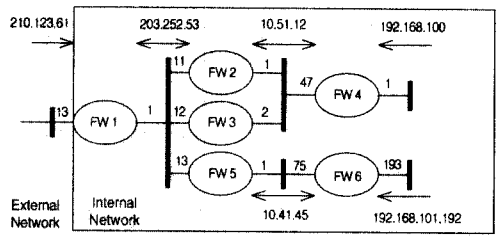
(그림 3) ISMS의 임의 네트워크의 방화벽 구성

제4장에서는 ISMS에서 패킷경로의 경로 상에 위치한 방화벽의 접근제어 정책 설정과 관련하여 네트워크 경로상의 방화벽 선택 방법을 살펴보도록 한다.

4. ISMS에서의 방화벽의 정책 설정

제3장에서 제기한 바와 같이 여러 대의 방화벽이 ISMS상에서 제어를 받고 있을 때는 하나의 접근제어 정책의 설정에도 여러 대의 방화벽이 관여하게 된다. 이때 추가/설정하고자 하는 정책의 적용 대상 방화벽은 정책의 근원지와 목적지 경로상에 위치한 방화벽의 집합이 된다. 그러므로 ISMS는 각 방화벽이 네트워크 상에서 설치된 위치를 파악하고 있어야 하며, 여기에는 다음과 같은 방법이 사용 될 수 있다.

먼저 IP로 구성된 네트워크의 Topology Discovery 알고리즘의 응용을 이용 할 수 있다[14]. 이 알고리즘은 tracerouter를 이용한 라우팅 정보와 ICMP 메시지, DNS정보, SNMP를 이용하여 네트워크 구성을 완성시킬 수 있다. 이렇게 얻어진 정보를 방화벽의 설정 정보와 결합을 하면 방화벽의 연결 정보 및 패킷경로상에 위치한 방화벽들을 선택하는 것이 가능하다. 추가 프로토콜의 구현 없이 네트워크 장애 및 구성정보 획득에 대해 유동적인 정책 설정이 가능한 것이 장점이라 할 수 있겠다[14]. 그러나 라우팅 프로토콜, ICMP, DNS 정보의 사용은 ISMS상의 침입탐지시스템에게 정상적인 탐지 장애 및 침입오판등의 문제를 야기할 수 있으며, ISMS상의 모든 방화벽에 ICMP 메시지를 허용하는 정책을 필요로 한다는 점에서 ISMS의 작동에 영향을 줄 우려가 존재한다. 매번 각 호스트에게 ICMP 메시지를 통한 구성정보를 계산해야 하는 부담이 존재하며 연산시간이 다소 소요돼 즉각적인 대응행동을 취해야 하는 ISMS에는 적합하지 못하다.



(a) 구성된 네트워크와 방화벽 배치의 예

TBL_FW

FID	NAME
1	FW1
2	FW2
3	FW3
4	FW4
5	FW5
6	FW6

TBL_CONNECT

FID	FW_INTERFACE_IP	NET_MASK	IN_OUT
1	210.123.61.13	24	out
1	203.252.53.1	24	in
2	203.252.53.11	24	out
2	10.51.12.1	24	in
3	203.252.53.12	24	out
3	10.51.12.2	24	in
4	10.51.12.47	24	out
4	192.168.100.1	24	in
5	203.252.53.13	24	out
5	10.41.45.1	24	in
6	10.41.45.75	24	out
6	192.168.101.193	26	in

(b) 방화벽 연결정보 테이블

(그림 4) 방화벽 연결정보

따라서 ISMS에서는 방화벽의 연결정보를 담은 인접 행렬을 이용하여 정책 설정 경로상에 위치한 방화벽을 찾아내고자 한다. 인접 행렬은 (그림 4)와 같이 구성한 네트워크 연결정보를 이용한다.

(그림 4)는 (그림 3)의 네트워크 형태로 임의로 구성한 네트워크 정보이며 FID는 방화벽 식별자, NAME은 방화벽 이름, FW_INTERFACE_IP는 방화벽 인터페이스의 IP주소, NET_MASK는 서브넷마스크, IN_OUT은 방화벽 인터페이스가 내부네트워크와 연결되어 있는지 외부네트워크에 연결되어 있는지를 각각 표시하며 외부네트워크쪽으로 향하는 인터페이스를 out으로 설정하였다. (그림 4)의 정보들은 각 방화벽의 ISM Agent로부터 받은 ISM Agent 구성정보, 방화벽 구성 정보를 이용하여 구성 한다. (그림 5)는 (그림 4)의 네트워크 정보를 행렬로 표현한 인접 행렬이다. 인접 행렬의 구성은 (그림 4)의 TBL_CONNECT 테이블의 각 방화벽이 서브넷마스크를 적용한 IP 주소가 같은 것이 존재하면 1, 존재하지 않으면 0으로 설정한다. 행과 열은 방화벽 ID를 지칭하며 표시된 값들은 다음과 같은 의미를 지닌다.

1 : 두 방화벽간 다른 방화벽을 거치지 않고 직접적으로 연결
0 : 다른 방화벽을 거치거나 연결이 이루어져 있지 않음

(그림 5)의 인접행렬의 경우 단순히 0과 1로만 방화벽들의 연결 관계를 표시하였다. 인접행렬의 값들은 라우팅 정보의 metric 값을 이용하여 라우팅 경로의 선택등의 연산을 수행할수 있다[11]. 여기에서는 정책설정대상으로 사용되는

근원지와 목적지에 위치한 방화벽의 선택이 주 연산이므로 직접 연결의 유무만 사용하도록 한다.

	FW1	FW2	FW3	FW4	FW5	FW6
FW1	1	1	1	0	1	0
FW2	1	1	1	0	0	0
FW3	1	1	1	1	0	0
FW4	0	0	1	1	0	0
FW5	1	0	0	0	1	1
FW6	0	0	0	0	1	1

(그림 5) 방화벽의 인접 행렬

새로운 방화벽에 대한 설치가 ISM Server에게 통지가 되면 ISM Server는 새로운 ISM Agent 구성 정보와 방화벽 구성 정보를 추가해 인접 행렬을 다시 구성한다.

(그림 6)은 정책을 적용시켜야 할 방화벽들의 집합을 구하는 Firewall selection ALgoritm for policy CONtrol(이하 FALCON) 의사코드이다. 각 프로시저의 인자는 다음과 같다.

src_obj : 근원지 호스트/네트워크의 주소
dst_obj : 목적지 호스트/네트워크의 주소
FWsrc : 근원지 호스트/네트워크를 제어하고 있는 방화벽
FWdst : 목적지 호스트/네트워크를 제어하고 있는 방화벽
FWset : FALCON 알고리즘에 의해 선택된 근원지 호스트/네트워크로부터 목적지 호스트/네트워크로 가는 경로상에 위치한 방화벽의 집합

```
PROCEDURE GetFWSet (src_obj, dst_obj)
  FWset_result ← 0 ;
  fw_init ← Determine FWsrc using src_obj,
          src_obj's NET_MASK,
          TBL_CONNECT.FW_INTERFACE_IP,
          TBL_CONNECT.NET_MASK and
          TBL_CONNECT.IN_OUT ;
  fw_end ← Determine FWdst using dst_obj,
          dst_obj's NET_MASK,
          TBL_CONNECT.FW_INTERFACE_IP,
          TBL_CONNECT.NET_MASK and
          TBL_CONNECT.IN_OUT ;
  CALL FALCON (fw_init, fw_end, Fwset_result) ;
END PROCEDURE GetFWSet

PROCEDURE FALCON (FWsrc, FWdst, FWset)
   Initializing inter_fw to be empty ;
   FOR Each FWsrc's adjacent firewall DO
     Inter_fw ←selects one of FWsrc's adjacent firewall ;
①  IF Inter_fw = FWdst THEN
     Add Inter_fw to FWset ;
     GIVE FWset to PROCEDURE GetFWSet ;
     RETURN 0 ;
   END IF
②  IF Inter_fw ∈ FWset THEN
     RETURN 0 ;
   ELSE IF
③    Add Inter_fw to FWset ;
     CALL FALCON(Inter_fw, FWdst, FWset) ;
   END IF
  END FOR
END PROCEDURE FALCON
```

(그림 6) FALCON 알고리즘

(그림 6)의 FALCON 알고리즘은 인접방화벽을 구하기 위해 (그림 5)의 인접 행렬을 이용하며 다음과 같이 수행된다.

먼저 프로시져 GetFWSet에서는 src_obj와 dst_obj를 제어하는 방화벽을 선택하게 된다.

근원지와 목적지의 두 호스트/네트워크를 제어하는 방화벽은 호스트/네트워크의 IP주소와 서브넷마스크, 방화벽 네트워크 인터페이스의 IP주소와 서브넷마스크를 비교하고 인터페이스가 내부 인터페이스인지의 여부로 얻을 수 있다.

프로시져 FALCON는 다음과 같이 수행된다.

① 선택한 방화벽이 목적지 방화벽과 일치하는 경우에는 FWset에 inter_fw를 추가하고 FWset을 GetFWset에 반환한다. FALCON가 recursion으로 작성되어 있으므로 그 이후의 연산을 계속하기 위해서 0(null set)을 반환한다.

② Inter_fw이 이미 FWset안에 존재한다면 이미 연산을 수행하여 거쳐간 방화벽이며 Loop를 돌고있는 것이므로 0(null set)을 반환하며 현재 연산을 종료한다.

③ 목적지 방화벽이 아니고, FWset에 포함되어있는 방화벽이 아닐 경우에는 목적지 방화벽으로 가는 경로에 놓인 방화벽일 가능성이 있으므로 FWset에 추가하고, 추가한 방화벽을 근원지 방화벽 인자로 하는 FALCON 프로시저를 재귀적으로 호출한다. 재귀적으로 호출된 FALCON 함수는 함수 Call 스택에 쌓여 해당 경로를 찾아 반환을 한다.

이렇게 얻어진 방화벽 집합은 패킷이 이동하는 라우팅 경로가 되며 근원지 호스트/네트워크와 목적지 호스트/네트워크를 대상으로 하는 정책의 설정 대상이 된다.

ISMS에서 FALCON 알고리즘을 사용할 때 얻을 수 있는 특징과 이점을 다음과 같이 나열할 수 있다.

- 침입탐지시스템, 방화벽 등이 존재하는 ISMS내에서 추가적인 패킷 생성을 최소화한다. 네트워크 구성 정보를 획득하기 위해 ICMP등의 프로토콜이나 메시지를 사용한다면 ISMS 시스템 내의 침입탐지시스템 같은 보안시스템에 직접적인 영향을 줄 수 있다. 이로 인해 정상적인 패킷 흐름을 방해 하거나 ISMS의 오동작을 유발 할 수 있다. 따라서 ISMS의 정상적인 작동을 위해서는 네트워크 패킷 흐름의 추가적인 발생을 줄이는 것이 바람직하다.
- (그림 3)의 방화벽 1, 2, 3, 4처럼 순환식으로 구성된, 즉 그래프 형태로 구성된 네트워크에서도 응용 가능하며, 각각의 경로에 대해 방화벽 집합을 얻을 수 있다.
- 방화벽 연결에 대한 정보를 인접행렬로 가지고 있으므

로 데이터베이스에 대한 접근 회수가 적다. 인접행렬의 갱신은 방화벽 연결 정보에 대한 변경이 일어나는 시기, 즉 방화벽의 구동, 중지, 추가, 삭제시에 시행한다. 또한 인접행렬은 주기억장치에 위치하고있어 방화벽의 검색시 빠른 응답을 보장받을 수 있다.

- 두 호스트 혹은 네트워크간 사이의 연결에 대해 이용가능한 모든 경로의 검색이 가능하다. 이것은 한 경로가 단절로 인한 장애시 대체경로를 가질 수 있음을 나타낸다.
- 인접행렬에 단순히 0, 1의 연결정보 대신에 라우터의 매트릭 정보를 이용하여 확장할 수 있다. 매트릭 값을 이용할 경우에는 우선경로, 대체경로등의 우선순위를 가질 수 있다.
- 네트워크 및 방화벽등의 보안시스템의 추가에도 유동적으로 대처할 수 있다. 데이터베이스내의 방화벽 및 방화벽의 연결정보만을 이용하여 인접행렬을 구성하므로, 데이터베이스내의 정보의 갱신을 통해 추가되는 보안시스템에도 많은 비용을 들이지 않고도 시스템의 정보를 갱신할 수 있다.

5. 결론 및 향후 과제

본 논문에서는 다양한 보안시스템을 중앙 집중 제어하는 통합보안관리시스템의 개요와, 통합보안관리시스템에서 방화벽의 접근제어 정책 설정시 고려해야 할 문제점, 그리고 이 문제점을 해결하기 위한 방화벽 검색 알고리즘인 FALCON 알고리즘을 논술했다. 이 알고리즘은 통합보안관리시스템에서 보안 정책 설정 대상이 되는 방화벽을 추출하는 알고리즘으로 신뢰성있고 관리자에게 투명성을 제공하는 시스템을 구축하기 위해서 필요한 알고리즘이다.

향후 연구 과제로는 라우팅 정보의 동적인 이용방안과 라우팅 경로의 매트릭값의 이용, 같은 패킷경로에 대해 같은 방화벽 정보가 나오도록 유지하는 무결성을 보장하기 위한 연구가 계속되어져야 한다.

참 고 문 헌

[1] D. Brent Chapman, E. D. Zwicky, "Building Internet Firewalls," O'Reilly&Associates, 1995.

[2] William R. Cheswick, Steven M. Bellovin, "Firewalls and Internet Security-repelling the willy hacker," Addison-Wesley, 1994.

[3] 이동영, 김동수, 홍승선, 정태명, "웹 기반의 방화벽 통합 보안관리 시스템 개발", 정보처리학회논문지, 제7권 제10호, pp. 3171-3181, 2000.

[4] 정연서, 류걸우, 장종수, "네트워크 보안을 위한 ESM 기술동향", 주간기술동향, 제1026호, pp.24-35, 2001.

[5] Simson Garfinkel, Gene Spafford, "Practical UNIX and Internet Security," O'Reilly&Associates, pp.637-668, 1996.

[6] "Internet Firewalls and Security," 3Com Corporation, 1996.

[7] Matt Curtin, Marcus J. Ranum, "Internet Firewall : Frequently Asked Questions," http://www.interhack.net/pubs/fwfaq/, 1999.

[8] Harold F. Tipton, Micki Krause, "Information Security Management Handbook," Auerbach publications, pp.73-131, 2000.

[9] "Technology Update Active Security," Ernst&Young, LLP, 1999.

[10] Charlie Kaufman, Radia Perlman, Mike Speciner, "Network Security-PRIVATE Communication in a PUBLIC World," Prentice Hall PTR, pp.6-35, 1995.

[11] W. Richard Stevens, "TCP/IP Illustrated, Volume1 The Protocols," Addison-Wesley, 1994.

[12] http://www.cert.org/stats/cert_stats.html.

[13] "NIMDA Worm/Virus Report Final," SANS Institute, 2001.

[14] H. Chun Lin, S. Chuan Lai, P. Wen Chen, "An Algorithm for Automatic Topology Discovery of IP Network," ICC 98, Vol.2, pp.1192-1196, 1998.

김 광 혁

e-mail : byraven@rtlab.skku.ac.kr
2000년 성균관대학교 정보공학(학사)
2000년 한국산업은행 정보시스템부
현재 성균관대학교 정보통신공학부 석사과정
관심분야 : 네트워크 보안, XML, 그리드 컴퓨팅, IPng, 홈네트워크

권 윤 주

e-mail : yulli@kisti.re.kr
2000년 성균관대학교 정보공학(학사)
2002년 성균관대학교 전기전자 및 컴퓨터공학부(석사)
현재 한국과학기술정보연구원 재직
관심분야 : 네트워크 보안, 그리드 컴퓨팅

김 동 수

e-mail : dskim@rtlab.skku.ac.kr
1998년 성균관대학교 정보공학(학사)
2000년 성균관대학교 정보공학(석사)
현재 성균관대학교 정보통신공학부 박사과정
관심분야 : 네트워크 관리, 네트워크 보안, 시스템 보안

정 태 명

e-mail : tmchung@ece.skku.ac.kr
1981년 연세대학교 전기공학(학사)
1984년 University of Illinois Chicago, 전자계산학과 학사
1987년 University of Illinois Chicago, 컴퓨터공학과 석사
1995년 Purdue University, 컴퓨터공학 박사
1985년~1987년 Waldner and Co., System Engineer
1987년~1990년 Bolt Bernek and Newman Labs., Staff Scientist
현재 성균관대학교 정보통신공학부 부교수
관심분야 : 실시간시스템, 네트워크 관리, 네트워크 보안, 시스템 보안, 전자상거래