프로그래머블 네트워크 기술을 이용한 네트워크 보안 관리 구조 제안

정회원 김 명 은*, 오 승 희**, 김 광 식***, 남 택 용****, 손 승 원*****

Proposal of Network Security Management architecture using Programmable Network Technology

Myung-Eun Kim, Seung-Hee Oh, Kwang-Sik Kim, Taek-Yong Nam, Sung-Won Sohn ***Regular Members***

요 약

본 논문에서는 이기종의 보안 장비를 보다 효율적으로 관리할 수 있는 보안 관리 구조와 관리 트래픽의 부하를 줄일 수 있는 프로그래머블 미들웨어를 제안하였다. 제안된 보안 관리 구조는 정책 기반 네트워크 관리 (Policy Based Network Management: PBNM) 구조에 프로그래머블 네트워크 기법을 접목한 것으로, 다양한 이기종의 보안 장비를 보안 정책을 통해 관리할 수 있으며, 보안 장비 간 연동을 제공한다. 또한, 미들웨어에서 보안 정책을 실행 가능한 형태로 변환해 줌으로써 관리상의 편이성을 제공하고 정책 서버의 부하를 줄일 수 있다. 본 논문에서는 제안된 구조와 PBNM 구조에서의 정책적용 및 변환시간과 메시지 전달시간을 비교함으로써 프로그래머블 미들웨어가 관리 트래픽의 부하를 줄일 수 있다는 것을 검증하였다.

Key Words : network security, security management architecture, security policy, programmable network

ABSTRACT

In this paper, we propose security management architecture that manages efficiently security systems that are produced by different companies and programmable middleware that can reduce the load of management traffic. The proposed architecture applies programmable networks technology to policy based network management (PBNM). The proposed architecture manages and cooperates various security systems using security policy. Also, the programmable middleware provides convenience of management and reduces the overhead of a policy server by translating security policy into execution command. In addition, using programmable middleware, an administrator can manage various security systems that are produced by different companies. We showed that the programmable middleware could reduce the load of management traffic by comparing processing time for enforcing and transferring of policies/messages between the proposed architecture and PBNM architecture.

Ⅰ. 서 론

네트워크 기술이 발달함에 따라 인터넷을 통해 사용자들은 원하는 정보를 보다 신속하게 얻을 수 있을 뿐 아니라 다양한 생활의 혜택을 누리게 되었다.

* 한국전자통신연구원 네트워크보안구조연구팀(mekim@etri.re.kr), ** 한국전자통신연구원 네트워크보안구조연구팀(seunghee5@etri.re.kr), *** 한국전자통신연구원 네트워크보안구조연구팀(kks634535@etri.re.kr), **** 한국전자통신연구원 네트워크보안구조연구팀(tynam@etri.re.kr), ***** 한국전자통신연구원 네트워크보안구조연구부(swsohn@etri.re.kr)

논문번호 : 030100-0312, 접수일자 : 2003년 3월 12일

그러나 이와 더불어 역기능인 바이러스, 해킹 등으로 인한 개인 정보 유출 및 시스템 파괴, 네트워크 침해도 점차 늘어가고 있는 실정이다. 90년대에는 바이러스나 해킹으로 인한 피해가 개인용 컴퓨터에 국한되어 피해 규모가 그다지 크지 않았으나, 바이러스와 해킹 기술이 고도화됨에 따라 네트워크 자체를 위협하는 공격으로 인한 피해 규모는 1.25 인터넷 서비스 대란과 같이 재난 수준으로 확대되고 있다. 이제 바이러스나 해킹은 네트워크를 사용하는 사용자라면 누구나 민감하게 반응하는 중요한 사회적 문제가 되고 있다.

이러한 사이버 위협에 대응하기 위한 보안 기술도 끊임없이 발전 양상과 변화를 거듭하고 있다. 최근 들어 나타나는 보안 기술의 변화는 다음과 같다.[1] 첫째로 네트워크 기술이 발달함에 따라 보다 많은 양의 데이터를 보다 빠르게 주고받을 수 있게 되었고, 이에 부응하여 보안 장비 또한 고속화, 대용량화를 추구하고 있다. 둘째로 기존의 네트워크 장비와 별도로 보안 장비를 추가 관리하는 부담을 줄이기 위해 표준 인터페이스를 이용하여 네트워크 장비와 통합하고자 하는 시도가 있다. 셋째로 분산되어 설치된 네트워크 장비를 중앙 집중적으로 관리하는 것처럼 네트워크 상에 설치된 보안 장비를 관리하기 위해 보안 관리 시스템에 정책 기반 관리 기법을 도입하고 있다. 넷째로 네트워크를 이용한 다양한 해킹 기법이 발달함에 따라 피해 범위가 점차 확대되고 피해 정도 역시 심화되고 있으므로, 수동적인 모니터링 수준이 아닌 적극적인 침입 대응 방안이 활발하게 연구되고 있다. 마지막으로, 기존의 단일 보안 기술을 통합함으로써 다양한 보안 제품들을 관리하는데 드는 비용을 줄이고, 다양한 개별 보안 시스템간의 연동을 통해 통합 보안 서비스를 제공하는 방향으로 변화하고 있다.

이와 같은 보안 기술의 변화와 다양한 보안 요구사항을 반영하여 기능별로 여러 보안 장비들이 등장하게 되었고, 이에 따라 분산 설치된 보안 장비들을 중앙 집중적으로 관리해 주는 보안 관리 시스템이 필요하게 되었다. 그러나 관리자가 다양한 이기종의 보안 장비를 연동하여 중앙에서 제어한다는 것은 쉽지 않은 일로 현재 통합 보안 관리 기법의 새로운 이슈로 떠오르고 있다.[2]

본 논문에서는 공중망에서 네트워크 보안을 위해 분산 설치된 이기종 보안 장비를 중앙에서 효율적으로 관리하기 위한 목적으로 보안 정책을 통해 제어하는 보안 관리 구조를 제안한다. 제안하는 구조는 프로그래머블 미들웨어를 이용함으로써 관리용 트래픽의 부하를 줄일 수 있고, 새롭게 추가되는 보안 기능을 쉽게 수용 가능하다. 본 논문의 구성은 다음과 같다. 2장에서 기존의 네트워크 관리의 한계점을 통합 보안 관리 시스템 측면에서 살펴보고, 프로그래머블 네트워크 기술을 적용한 정책 기반 네트워크 관리에 대한 기존 연구를 알아본다. 3장에서는 미들웨어 측면에 중점을 둔 네트워크 보안 관리 구조를 제안하고, 시나리오를 통해서 설명한다. 4장에서는 제안하는 네트워크 보안 관리 구조의 성능을 평가하고, 마지막으로 5장에서 결론을 맺는다.

II. 관련 연구

본 장에서는 통합 보안 관리 기술과 프로그래머블 네트워크 기술을 적용한 정책 기반 네트워크 관리(Policy Based Network Management: PBNM) 기술에 대한 기존 연구 동향을 살펴본다.

1. 통합 보안 관리 기술

보안 요구사항이 다양화됨에 따라 요구사항에 부합하는 다양한 개별 보안 장비들이 나오게 되었고, 이러한 이기종의 보안 장비들을 효율적으로 관리하기 위해 통합보안 관리시스템(Integrated Security Management System)이 등장하게 되었다.

통합 보안 관리 기술을 도입하면 중앙 집중화된 관리자 인터페이스를 통한 전체적인 정보보호 정책의 수립 및 제어, 보안 관리 정보 공유를 통한 보안 영역(Secure Domain) 형성, 보안 관리의 자율성 및 안전성 확보, 유연성, 확장성 확보 등의 장점을 얻을 수 있다. 현재 가장 대표적인 통합 보안 관리 기술로는 체크포인트 (Check Point Software Technologies, Inc.)의 OPSEC (Open Platform for Security)[3]과 네트워크 어소시에이트 (Network Associates, Inc.)의 Active Security[4]가 있다.

1) OPSEC

OPSEC은 체크포인트사가 제안하고 있는 프레임워크로 Symantec, Axent, RSA security, VeriSign, IBM, Novell 등 320여 개의 보안 업체가 파트너로 참여하고 있다. OPSEC은 침입차단시스템을 중심으로 하여 보안 시스템들을 상호 연동한 자율적 보안 관리를 목적으로 한다. OPSEC은 Secure Virtual Network (SVN) 구조를 기반으로 Firewall-1,

VPN-1과의 호환성 확보를 통해 보안 환경을 제공한다. 1차적으로 침입탐지시스템, CA 서버 등을 통합하여 전사적인 보안 환경을 제공하고자 한다.

OPSEC은 이기종의 보안 시스템 간 상호 연동 및 정보 교환을 위한 프로토콜인 Content Vectoring Protocol (CVP), Suspicious Activity Monitoring Protocol (SAMP)와 관리자의 편의를 위한 Log Export API (LEA), User-to-Address Mapping (UAM) 등의 애플리케이션으로 구성되어 일관되고 효율적인 보안 관리를 제공한다. 그러나 OPSEC은 각각의 정보 보호 제품이 이러한 애플리케이션과 프로토콜을 수용해야 하는 단점과 인터페이스가 연결된 두 제품 사이에서만 통합 가능하다는 한계가 있다.

2) Active Security

"Active Security"는 네트워크 어소시에이트가 제안한 것으로, 자사의 보안 제품들을 연결하여 보안 제품을 통합하기 위한 자율적 중앙 관리 보안 방식으로써, 현재 Gauntlet Firewall, Gauntlet VPN Server, Event Orchestra, CyberCop Scanner, Net Tools PKI 등이 Active Security 환경을 지원한다.

Active Security는 보안 사건 및 문제점을 감시하고 진단하는 센서 (Sensor), 보안 정책 설정을 위한 아비터 (Arbiter), 각종 사건에 대응 행동 양식을 결정하는 주체인 이벤트 오케스트라 (Event orchestrator)와 이벤트 오케스트라로부터 명령을 받아 보안 사건에 직접 대응하는 액터 (Actor)로 구성되어 있다. 이벤트 오케스트라는 보안 시스템을 관리하는 관리 시스템으로 사건 수집, 로그 관리, 보안 제품들에 대한 통합적 보안 정책 실행 지시 등을 중앙 집중적으로 수행한다. Active Security도 OPSEC과 마찬가지로 자사 제품을 중심으로 통합을 제공하고 있으므로 서로 다른 보안 시스템을 연동하는데 구조적인 한계가 존재한다.

2. 프로그래머블 네트워크 기술

프로그래머블 네트워크[5]는 기존의 store-and-forward 방식의 라우터에 컴퓨팅 능력을 추가함으로써 store-compute-forward 방식으로 패킷을 처리할 수 있도록 한다. 기존 라우터에는 컴퓨팅 능력이 없었으므로 모든 처리가 종단 시스템 기반으로만 수행되었던 데 반해, 프로그래머블 네트워크 기술을 적용한 패킷을 이용하면 프로그래머블 노드에서 중간 처리를 수행함으로써 기존 종단 간 (end-to-end) 방식에 비해 성능의 향상을 얻을 수 있다. 또한, 프로그래머블 네트워크는 새로운 네트워크 서비스를 빠르고 효율적으로 배치하기 위해 제안된 구조로, 기존의 IP 네트워크 서비스의 중단 없이 새로운 서비스 적용이 가능하며 유연성, 성능 향상, 보안 강화, 관리의 편리성과 같은 장점들을 제공할 수 있다.

액티브 네트워크는 프로그래머블 네트워크를 구현하는 여러 방식 중 하나로 액티브 응용 (Active Application), 실행환경 (Execution Environment), 노드 OS (NodeOS)로 구성된다.

다음 절에서는 프로그래머블 네트워크 기술의 다양한 연구 분야 중에서 정책 기반 네트워크 관리 (PBNM)를 적용한 IST (Information Society Technologies)의 FAIN (Future Active IP Networks) 프로젝트[6]와 일본의 OKI Electric Industry Co.에서 수행했던 PBNM 관련 프로젝트[7]에 대해 간략히 살펴본다.

1) FAIN

IST의 FAIN 프로젝트는 기존의 액티브 노드 개념에 기반을 두고 있으며, 개방성을 지향하고, 유연성을 제공하는 프로그래머블 네트워크 구조를 개발하고 있다. FAIN에서 액티브 네트워크 기술을 기반으로 연구하고 있는 분야 중에서 Policy Based Active Network Element Management (PBANEM)은 정책 기반 네트워크 관리에 대한 것으로 Internet Engineering Task Force (IETF)와 Distributed Management Task Force (DMTF)의 표준을 수용하고 있다.

PBANEM는 Policy Decision Point (PDP)와 Policy Enforcement Point (PEP)로 구성되어 있다. PBANEM에서는 조건부 (condition)와 실행부 (action)로 이루어진 메타 정책 (meta-policy)을 액티브 패킷 형태로 전달한다. 기존의 PBNM은 각 PEP에 해당하는 정책을 개별적으로 생성하여 전달해야 했다. 그러나 FAIN의 PBANEM은 액티브 패킷을 이용하여 정책을 전달하므로, PEP 종류별로 적합한 형태의 보안 정책이 적용되기 때문에 PDP에서 생성한 하나의 정책이 서로 다른 PEP에서도 적용 가능한 장점을 가지고 있다.

PBANEM에서 액티브 네트워크 기술을 기존의 PBNM에 적용함으로써 전달되는 액티브 패킷의 프로그래머블 가능한 특징을 통해서 효율적으로 정책

을 전달 및 적용할 수 있다.

2) Active Program Execution System (APES)

OKI에서는 IP 네트워크에 서비스 품질 보장 (QoS)을 제공하기 위하여 액티브 패킷 방식을 적용한 연구를 수행한 바 있다. OKI에서는 초기에 IP 네트워크에 PBNM 기법을 적용하였으나, SNMP (Simple Network Management Protocol), COPS (Common Open Policy Service), CLI (Command Line Interface) 등의 프로토콜을 이용한 경우 과도한 관리용 트래픽이 발생하는 문제점이 발견되었다. 따라서 이를 해결하기 위하여 정책 기반 네트워크 관리에 액티브 네트워크 기술을 적용하였다.

OKI에서 개발한 정책 기반 네트워크 관리 시스템은 관리자와 네트워크 장비인 액티브 노드로 구성되며, 각 관리자와 액티브 노드에는 액티브 패킷 형태로 전달된 정책을 처리하고 근접한 다른 액티브 노드에게 전달하기 위한 APES가 설치되어 있다.

OKI 프로젝트는 액티브 패킷을 이용하여 정책을 전달 및 적용함으로써 중복된 관리용 트래픽을 제거하여 네트워크 효율을 높이고, 네트워크에 유연성, 확장성 및 사용자 요구에 부합하는 서비스를 제공하고 있다.

III. 프로그래머블 네트워크 기법을 이용한 보안관리구조

기존의 통합 보안 관리 시스템은 설치된 모든 보안 장비에 대한 정보를 관리해야 하는 부담과 제품간 호환성 문제로 인해 대부분 자사 제품의 보안 장비만을 관리하는 것에 그치고 있다.

보안 장비간의 연동 및 관리상의 편이를 위해 동일한 회사의 보안 장비로만 네트워크를 구성하는 것이 하나의 방안이 될 수 있으나, 네트워크의 단위가 점차 거대화되고 있는 현실을 볼 때 모든 네트워크에 동일한 회사의 보안 장비를 설치한다는 것은 거의 불가능한 일이다. 이기종 보안 장비간의 호환성을 위해 보안 업계에서는 다양한 관점으로 접근하고 있으나, 이기종의 보안 장비를 하나의 보안 관리 시스템에서 관리한다는 것은 여전히 쉽지 않은 일이며, 이기종 보안 장비간 연동은 해결해야 할 문제로 남아있다.

본 논문에서는 이러한 문제를 해결하기 위해 기존의 네트워크 위에 프로그래머블 미들웨어를 기반으로 하는 관리 계층을 제공하고, 미들웨어를 통해 보안 정책을 적용함으로써 이기종의 다양한 보안 장비간의 호환성을 제공할 뿐만 아니라 관리상의 부하를 줄일 수 있는 보안 관리 구조를 제안한다.

그림 1은 제안된 보안 관리 구조를 나타내고 있으며 그림에서 알 수 있듯이 제안된 구조는 크게 SPMP (Security Policy Management Part)와 SPEP (Security Policy Enforcement Part) 두 부분으로 구성된다. SPMP는 보안 정책 관리부로 생성된 정책을 보안 정책 저장소에 저장하고, 각 보안 장비에 적합한 보안 정책을 배포하는 일을 담당한다. SPEP는 보안 정책 실행부로 SPMP로부터 받은 보안 정책을 LPR에 저장되어 있는 보안 정책과 충돌이 일어나는지 검사한 후에 충돌이 없을 경우 보안 정책을 실행하게 된다. 각 구성요소에 대한 자세한 설명은 다음 절에서 하기로 한다.

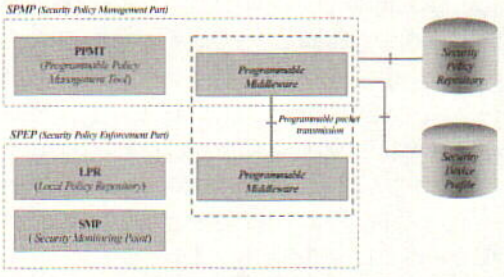

그림1. 프로그래머블 미들웨어를 이용한 보안관리구조

1. 프로그래머블 정책 관리 툴(PPMT)

PPMT는 보안 정책을 관리하는 역할을 담당하며, 각 보안 장비에 필요한 보안 정책을 생성하여 보안 정책 저장소에 저장한다. PPMT가 관리하는 보안 정책은 특정 보안 장비에서 실행 가능한 형태가 아닌 추상화된 형태로 표현된다. 즉, 제안된 구조에서 정의한 보안 정책이란 침입 탐지 규칙, 침입 차단 규칙, 바이러스 탐지 및 차단 규칙 등 각 보안 장비에서 수행되는 커맨드 및 보안 장비의 설정값 등을 추상화한 것이다. 따라서 관리자는 이기종의 보안 장비의 종류 및 실행커맨드를 자세하게 알 필요 없이 상위레벨에서 보안정책을 정의하고 관리하므로 관리상의 부하를 줄일 수 있다.

이러한 보안 정책은 역할에 따라 침입 탐지 시스템에 적용될 정책, 침입 차단 시스템에 적용될 정책, 항바이러스 시스템에 적용될 정책 등으로 분류하여 관리된다.

그림 2는 침입 차단 시스템에 적용할 수 있는 보안 정책으로 추상화된 형태와 프로그래머블 형태의 보안 정책을 도시화 한 것이다.

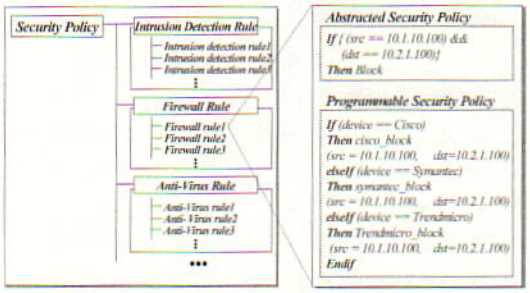

그림2. 보안 정책의 예

PPMT는 보안 정책을 배포할 때 보안 정책 저장소에서 해당되는 정책을 검색한 후 해당 정책이 존재할 경우 보안 장비 프로파일을 읽어 보안 정책을 다양한 보안 장비의 실행 커맨드로 구성된 프로그래머블 보안 정책으로 변환한다. 변환된 프로그래머블 보안 정책은 각 보안장비로 전송되어 실행된다. 존재하지 않을 경우, PPMT는 새로운 보안 정책을 생성하고 프로그래머블 보안 정책으로 변환하여 미들웨어에게 전달한다.

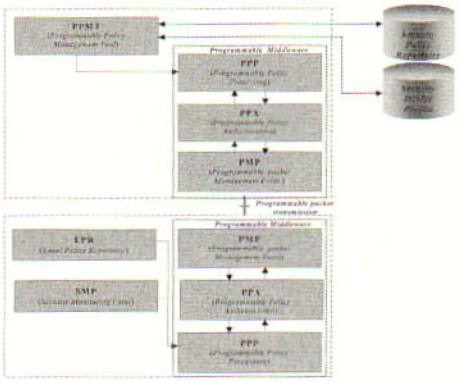

그림3. 보안 정책 처리흐름도

그림 3은 제안된 보안 관리 구조에서 보안 정책의 흐름을 나타낸 것이다. PPMT에서 생성된 보안 정책은 미들웨어를 통해 프로그래머블 패킷 형태로 변환되어 SPEP에게 전달된다. 전달된 프로그래머블 보안 정책은 미들웨어내의 PPP를 통해 실행되는 것을 볼 수 있다. 또한 보안 정책 실행시 SMP에서 정책 충돌여부를 확인하여 기존의 보안 정책과 충돌이 일어나는지 확인한다. SPMP에서 SPEP로 보안 정책을 전달할 때 오류가 발생할 경우 SPEP는 SPMP에게 전송실패를 알리고 재전송을 요구한다.

2. 프로그래머블 미들웨어

프로그래머블 미들웨어는 PPP (Programmable Policy Processing), PPA (Programmable Policy Authentication), PMP (Programmable packet Management Point)의 세 부분으로 구성되며 PPMT에서 생성한 프로그래머블 보안 정책을 프로그래머블 패킷의 형태로 가공한다.

PMP는 프로그래머블 패킷을 처리를 담당하고, PPA는 패킷의 보안을 위해 인증 및 암호화 처리를 담당하며, PPP는 프로그래머블 보안 정책 실행을 담당한다. 그림 4는 프로그래머블 미들웨어의 구성요소를 나타낸 그림이다. 각 부분에 대한 자세한 설명은 다음 절에서 하기로 한다.

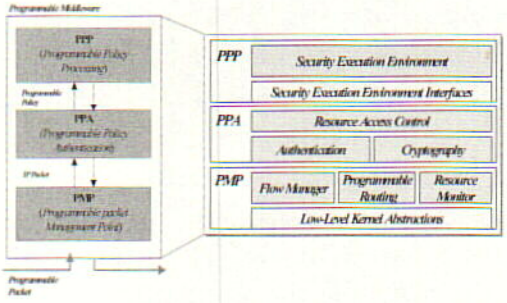

그림4. 프로그래머블 미들웨어 구조

1) 프로그래머블 정책 처리부(PPP)

PPP는 PPMT로부터 프로그래머블 정책을 받아 페이로드 부분에 채워 넣어 PPA로 보내고, PPA로부터 프로그래머블 보안 정책을 받아 해당 보안 장비에 맞는 보안 정책을 실행시킨다.

PPP에 전달된 프로그래머블 보안 정책은 제공된 보안 실행환경에서 수행되며, 제안된 구조는 이외의 다른 실행환경을 지원하지 않는다.

2) 프로그래머블 정책 인증부(PPA)

PPA는 PPP로부터 받은 정책을 보호하기 위해 프로그래머블 미들웨어의 성능에 따라 적합한 암호 알고리즘(예. 3DES, 등)을 선택하여 암호화한다. PMP로부터 패킷을 받은 PPA는 패킷의 인증서를 통해 근원지 네트워크 구성요소와 송신자를 확인한다. 또한 인증 및 신원 확인을 기반으로 적합한 자원에 대해 접근권한을 부여한다.

3) 프로그래머블 패킷 관리부(PMP)

PMP는 프로그래머블 패킷에 관련된 모든 관리를 담당하는 부분으로 PPA로부터 하나의 패킷이 들어오면 프로그래머블 패킷으로 변환하여 전송하고, SPMP로부터 받은 프로그래머블 패킷을 IP 패킷으로 변환하여 PPA로 전달하는 역할을 한다. 프로그래머블 패킷에 대한 스케줄링은 플로우 관리자가 담당하고, 보안 장비간의 통신은 프로그래머블 라우팅 모듈이 담당한다. 또한 자원 관리자는 사용되고 있는 모든 자원을 모니터링 함으로써 적절히 자원을 분배하도록 한다.

3. 지역 정책 저장소(LPR)

LPR은 SPMP로부터 전송 받은 보안 정책을 일시적으로 보관하는 역할을 한다. 이미 적용한 동일한 보안 정책을 다시 필요로 할 경우, 혹은 이웃한 도메인에 있는 보안 장비로의 정책 전달이 실패하여 재전송을 요구하는 경우 LPR에 있는 보안 정책을 사용하여 다시 재전송하기 위해 보관한다.

4. 보안 감시부(SMP)

SMP는 SPMP에서 생성한 보안 정책이 SPEP에서 올바르게 수행되고 있는지 모니터링 하는 역할을 한다. 또한 새로 적용되는 보안 정책이 기존에 수행되고 있는 보안 정책에 위배되는지 검사하여 충돌이 발생할 경우 정책 서버에게 메시지를 보낸다.

5. 인터페이스

본 절에서는 정책 서버와 보안 장비간의 통신을 위해 SPMP와 SPEP 간의 인터페이스와 보안 장비간의 통신을 위해서 SPEP 간의 신뢰성 있는 통신을 제공하기 위한 인터페이스에 대해 다룬다.

1) 프로그래머블 라우팅

기존의 IP 네트워크와 오버레이된 상태로 프로그래머블 네트워크가 존재하는 경우, 다른 프로그래머블 노드를 찾아가기 위해서는 새로운 라우팅 기법이 요구된다. 제안하는 프로그래머블 라우팅은 네트워크 상에 존재하는 프로그래머블 노드인 보안 장비들을 찾아갈 수 있도록 하는 방식이다. 본 논문에서는 IP 네트워크에서 프로그래머블 노드들이 네트워크 내의 다른 프로그래머블 노드의 위치와 특성을 알 수 있도록 하기 위하여 ALCATEL에서 제안한 OSPFv2 Opaque LSA (Link State Advertisement) Option[8]을 이용한다. 이 때 사용하는 Opaque LSA의 형식 중에서 일부 옵션은 [9]에 따라 아래와 같이 수정하였다.

- Opaque Type: IANA (Internet Assigned Numbers Authority)에서 실험적인 사용을 위해 제공하고 있는 범위의 번호인 128을 사용하도록 한다.
- Opaque ID: 0을 넣어 Opaque 정보가 실행환경 ID임을 나타낸다.
- Opaque Information: 프로그래머블 노드 자신의 실행환경 ID를 넣도록 한다.

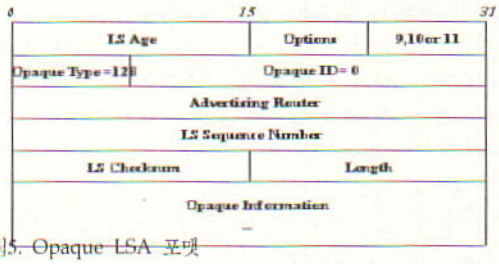

그림5. Opaque LSA 포맷

프로그래머블 노드가 위치하고 프로그래머블 패킷이 전송되어 실행되는 환경 역시 [9]에서 제안한 가정을 따른다.

- 프로그래머블 패킷의 송신자와 수신자는 프로그래머블 패킷 전송 엔진을 포함하는 프로그래머블 노드이다.
- 전송되는 프로그래머블 패킷은 'Source Route Option'에서 'Loose Source Route' 옵션[10][11]을 이용한다.
- 신뢰성 있는 보안 정책 전송을 위해서, 프로그래머블 보안 정책을 위한 프로그래머블 패킷 포맷으로 ANEP (Active Network Encapsulation Protocol) 패킷 구조[12]를 이용한다.

그림 6과 같이 정책 서버의 SPMP에서 생성된 프로그래머블 보안 정책은 특정 보안 장비에게 전달된다. 보안 정책 내에는 정책이 반영되어야 하는 장비의 리스트가 있으며, 보안 정책은 프

로그래머블 패킷 형태로 리스트에 있는 보안 장비에게 전달된다. 프로그래머블 패킷이 각 보안 장비를 방문하여 보안 정책을 적용하면 보안 장비는 적용 성공 여부를 프로그래머블 패킷에 기록한다. 최종 보안 장비를 방문한 후 프로그래머블 패킷은 정책서버에게로 돌아가 수행결과를 알린다. 수행 결과를 통해 정책서버는 적용이 실패한 보안장비에게 보안 정책을 재전송한다. 이 때 보안 장비간의 정책 전달 과정은 다음과 같다. 해당 프로그래머블 패킷을 전달 받은 보안 장비는 이를 복사한 후에 다음 보안 장비에게 전달하고, 동시에 복사본 패킷에서 헤더를 제거한 ANEP 부분을 해당 도메인의 네트워크 장비가 이해할 수 있는 형태로 정책을 재구성한다. 이와 같은 과정이 모든 보안 장비에게 해당 패킷이 전달될 때까지 반복된다. 여기서 다음 보안 장비는 Opaque LSA를 이용해서 찾아간다.

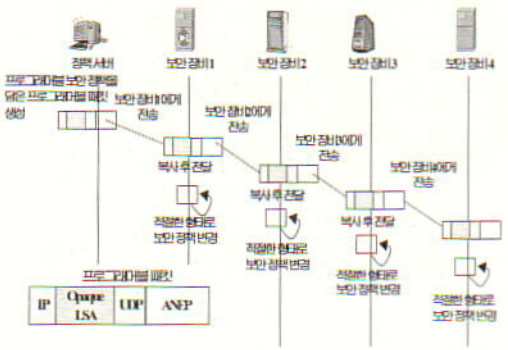

그림6. 보안 정책을 담은 프로그래머블 패킷의 전달 과정

프로그래머블 패킷이 중간에 손실되었을 경우에 재전송하는 과정은 그림 7에서 보여주고 있다. 신뢰성 있는 전송을 제공하기 위하여 프로그래머블 패킷의 시퀀스 번호를 검사하여 손실을 감지한다. 시퀀스 번호의 시작 번호나 전송될 패킷의 수 등에 대한 정보는 최초로 송신자가 보내는 메시지 첫 패킷에 실어 보낸다. 손실을 감지하면 수신자는 바로 직전의 프로그래머블 노드인 보안 장비에게 사라진 패킷의 시퀀스 번호를 담고 있는 "재전송 요구 패킷"을 보낸다. 가장 가까운 보안 장비는 이것을 받으면 해당 프로그래머블 패킷을 다시 재전송을 요구한 수신자에게 전송한다.

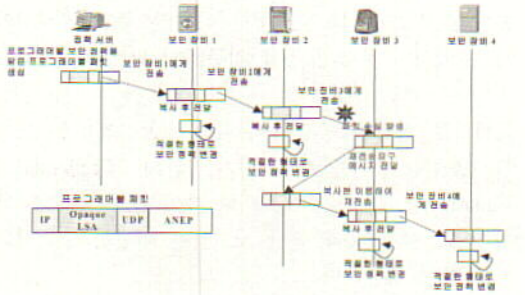

그림7. 보안 정책을 담은 프로그래머블 패킷의 전달 과정

2) 프로그래머블 패킷 신뢰성 제공 방식

본 논문에서는 프로토콜 스택으로 IP 프로토콜과 UDP 프로토콜을 이용함에 따라 발생할 수 있는 신뢰성에 대한 취약 부분을 ANEP 패킷 포맷을 일부 수정함으로써 해결한다.

그림 8은 프로그래머블 네트워크에서 안전한 통신을 위한 요구 사항을 바탕으로 설계한 프로그래머블 보안 정책을 위한 패킷 포맷이다.[13]

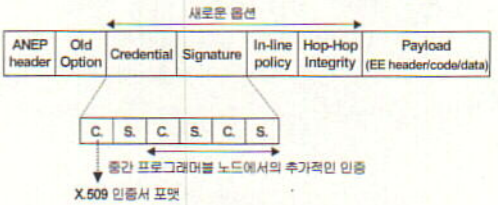

그림8. 수정한 ANEP 패킷 포맷

본 논문에서는 IP 헤더와 UDP 헤더 다음에 ANEP 헤더를 붙이고 ANEP 헤더의 기존 옵션(old option)으로 근원지 식별자 (Source Identifier), 목적지 식별자 (Destination Identifier), 무결성 검사 (Integrity Checksum), 비협상 인증 (Non-Negotiation Authentication)의 네 가지 항목을 두고 새로운 옵션으로 아래의 4 가지 항목을 추가한다.

- Credential: X.509 포맷의 Credential을 이용하여 프로그래머블 패킷을 생성한 사용자의 신분을 검증하고 송신측 프로그래머블 노드를 검증하고 인증한다.
- Signature: 수신한 프로그래머블 패킷의 변경 여부를 검증한다.
- In-line policy: 수신한 프로그래머블 패킷의 데이터나 실행할 프로그램을 위한 권한 부여에 사용한다.
- Hop-hop Integrity: 프로그래머블 패킷을 생성한

송신 프로그래머블 노드뿐만 아니라 중간 홉에서의 인증, 메시지 무결성 등을 검증하기 위해 이용한다.

목적지로 향하는 중간에 위치한 프로그래머블 노드인 보안 장비를 신뢰하기 위해 Credential과 Signature를 각 중간 노드마다 추가할 수 있도록 되어 있고, 이를 통해 중간 노드들을 인증하므로 신뢰성 보장이 가능하다.

6. 적용 시나리오

본 절에서는 제안된 관리구조를 사용하여 적용 가능한 수행 시나리오에 대하여 설명한다. 먼저 프로그래머블 미들웨어를 이용하여 이기종 보안 장비에 정책을 배포하는 시나리오에 대하여 설명하고, 이기종 보안 장비의 연동을 통해 유해 트래픽을 차단하는 시나리오에 대하여 설명한다.

1) 정책 배포 시나리오

그림 9는 제안된 관리 구조를 사용하여 이기종 보안 장비를 관리할 경우 정책 서버에서 프로그래머블 미들웨어를 통해 보안 정책을 배포하는 과정을 나타낸 그림이다.

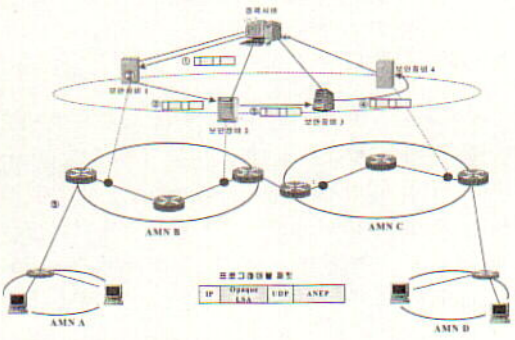

그림9. 정책 배포 시나리오

정책 서버는 새로운 보안 정책을 생성하여 관리하고 있는 모든 보안 장비에게 배포하고자 할 때 프로그래머블 미들웨어를 통해 보다 효율적으로 배포할 수 있다. 먼저 정책 서버내의 PPMT는 보안 정책을 생성하여 보안 정책 저장소에 저장한다. PPMT는 보안 장비의 프로파일을 읽어 생성된 보안 정책을 각 보안 장비에서 수행 가능한 프로그래머블 보안 정책으로 변환한다. 프로그래머블 미들웨어에 위치한 PPP는 변환된 프로그래머블 보안 정책을 받아 패킷의 페이로드 부분에 넣고, PPA는 패킷의 보안을 위해 암호화하고 인증을 위한 헤더를 붙인다. PMP는 이것을 프로그래머블 패킷으로 변환하여 보안 장비에게 송신한다.

보안 장비1은 프로그래머블 패킷을 받아 PMP에서 IP 패킷으로 변환하고, PPA는 해당 패킷이 유효한 패킷인지 인증작업을 통해 검사하고, 유효한 패킷일 경우 페이로드 부분을 복호화한다. PPP는 복호화된 프로그래머블 보안 정책을 받아 자신에게 해당되는 프로그래머블 보안 정책을 실행하면서 보안 정책을 다음 목적지인 보안 장비2에게 전송한다. 이와 같은 형태로 보안 정책은 최종 목적지인 보안 장비4까지 전달되어 실행된다.

2) 침입 차단 시나리오

그림 10은 제안된 보안 관리 구조를 적용하여 침입을 탐지하고 유해 트래픽을 차단하는 시나리오를 나타낸 그림이다. 시나리오를 단계별로 살펴보면 다음과 같다.

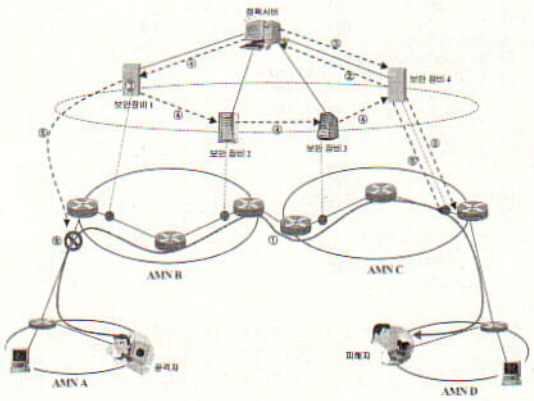

그림10. 침입 차단 시나리오

(단계1) AMN A에 있는 공격자가 AMN D에 있는 시스템을 공격한다.

(단계2) 보안 장비4가 유해 트래픽을 감지하고 정책 서버에게 경보 메시지를 송신한다.

(단계3) 정책 서버는 경보 메시지를 분석하여 보안 장비4에게 차단 명령을 내린다.

(단계4) 정책 서버는 유해 트래픽을 차단하라는 보안 정책을 프로그래머블 패킷형태로 모든 보안장비에게 전송한다.

(단계4-1) 정책 서버는 보안 장비의 프로파일을 읽어 보안 정책을 프로그래머블 보안 정책으로 변환하여 프로그래머블 패킷 형태로 보안 장비1에게 전

송한다.
(단계4-2) 보안 장비1은 보안 정책을 복사한 후 보안 장비2에게 전송한다.
(단계4-3) 보안 장비2는 보안 정책을 복사한 후 보안 장비3에게 전송한다.
(단계4-4) 보안 장비3은 보안 정책을 복사한 후 보안 장비4에게 전송한다.
(단계5) 보안 정책을 다음 보안 장비로 전송하면서 보안 장비는 유해 트래픽의 근원지가 자신의 관리 도메인인지 확인하여 보안 정책을 실행한다.
(단계5-1) 보안 장비 2, 3, 4는 자신의 관리 도메인이 아니므로 실행하지 않는다.
(단계5-2) 보안 장비 1은 유해 트래픽이 자신의 관리 도메인에서 발생했음을 확인한다.
(단계5-3) 보안 장비 1은 프로그래머블 보안 정책을 실행한다.
(단계6) 보안 장비 1에 의해 유해 트래픽은 차단되고, 보안 장비 4에게 내린 차단 정책을 제거한다.

Ⅳ. 성능 검증

본 장에서는 이기종의 보안 장비를 관리할 때 기존의 PBNM 기반 관리 방식과 제안된 관리 방식에서 정책 전달 및 메시지 전달시 필요한 시간을 비교함으로써 제안된 구조의 성능을 검증한다.

1. PBNM 구조에서의 성능 검증

기존의 PBNM 기반 관리 방식은 TCP 기반에서 서버와 클라이언트 간에 COPS를 통해 PIB (Policy Information Base)형태로 정책을 전송한다. 이 때 서버는 클라이언트와의 연결을 유지해야 하며, 연결 상태의 이상 유무를 항상 확인하고 있어야 한다. 서버는 연결 상태를 확인하여 이상이 없을 경우 클라이언트에게 정책을 전달하고 정책을 받은 클라이언트는 서버에게 리포트 메시지를 전송한다.
그림 11은 PBNM 기반 관리방식에서 이기종 보안 장비에게 정책을 배포하는 과정을 나타내고 있다. 여기서 정책 서버는 관리자가 이기종 보안 장비의 커맨드를 직접 입력하여 정책을 생성하는 것으로 가정한다.

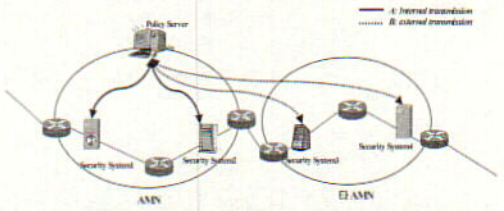

그림11. PBNM 구조에서의 정책 배포 및 적용 과정

정책 서버에서 모든 보안 장비에게 정책을 배포하고 적용하는 데 걸리는 시간을 T1이라고 하자. T1은 동일한 AMN (Autonomous Management Network)내에 있는 보안 장비에게 정책을 배포하는 데 걸리는 시간과 다른 AMN내에 있는 보안 장비에게 정책을 배포하는데 걸리는 시간의 합으로 표현할 수 있다. 이 때 전송실패확률 Fh는 5%라고 가정하고, 관리하고 있는 AMN 개수를 Nm, 평균 재전송 횟수를 Nrt 라 할 때, 모든 보안 장비에 안전하게 보안 정책을 배포하기까지의 시간 T1은 식 (1)으로 구할 수 있다. 여기서 사용되는 모든 파라미터는 표 1에 정리하였다.

표 1 보안 정책 배포 및 적용에 사용된 파라미터

파라미터	설 명	종류	값
Tpt	정책 서버내의 정책 변환 시간	정수	20msec
Tpp	보안 장비내의 정책 처리 시간	정수	30msec
Tpc	정책 서버내의 정책 생성 시간	정수	10msec
Tout	타 AMN간 메시지 정책 지연 시간	정수	100msec
Tin	동일 AMN내에서의 정책 전송 지연 시간	정수	10msec
Sh	전송성공확률	정수	98%
Ns	AMN당 평균 보안 장비 갯수	변수	1->10
Nrt	평균 재전송 횟수	변수	-
Nr	반복 전송 횟수	변수	-
Nm	관리하고 있는 AMN 갯수	변수	1->10
Gs	보안 장비당 패킷 처리 시간	정수	1msec
Ft	전송실패 목표확률	정수	5%
Fh	전송실패확률	정수	2%

$$T1 = \{ A + B \times (Nm - 1) \} \times Nrt \quad (1)$$

여기서 A는 정책 서버에서 정책을 생성하여 동일한 AMN내에 존재하는 모든 보안 장비에 전송하고, 각 보안 장비에서 정책이 실행되기까지의 시간을 의미

하며 표 1에 있는 파라미터를 이용하여 식(2)를 도출할 수 있다.

$$A = (Tpc + Tpt + Tin \times 2 + Gs + Tpp) \times Ns \quad (2)$$

B는 정책 서버에서 정책을 생성하여 다른 AMN내에 존재하는 모든 보안 장비에 전송하고, 각 보안 장비에서 정책이 실행되기까지의 시간을 의미하며 표 1에 있는 파라미터를 이용하여 식(3)를 도출할 수 있다.

$$B = (Tpc + Tpt + Tout \times 2 + Gs + Tpp) \times Ns \quad (3)$$

식(2)와 식(3)에 의해 식(4)를 도출할 수 있다.

$$T1 = [\{ (Tpc + Tpt + Tin \times 2 + Gs + Tpp) \times Ns \} + \{ (Tpc + Tpt + Tout \times 2 + Gs + Tpp) \times Ns \} \times (Nm-1)] \times Nrt \quad (4)$$

여기서 정책전송이 실패할 확률을 Fh라고 할 때 평균 재전송 횟수 Nrt는 1/(1-Fh)로 구할 수 있으며 T1은 식(5)로 구할 수 있다.

$$T1 = [\{ (Tpc + Tpt + Tin \times 2 + Gs + Tpp) \times Ns \} + \{ (Tpc + Tpt + Tout \times 2 + Gs + Tpp) \times Ns \} \times (Nm - 1)] \times 1 / (1 - Fh) \quad (5)$$

2. 제안된 구조에서의 성능 검증

그림 12는 제안된 구조에서 관리하고 있는 보안 장비에 정책을 배포하고 적용하는 그림을 나타낸 것이다.

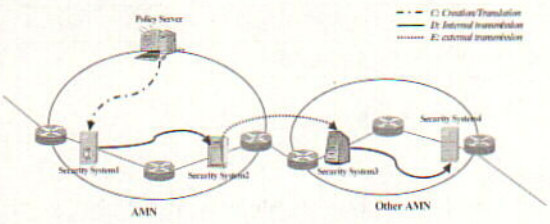

그림12. 제안된 구조에서의 정책 배포 및 적용 과정

제안된 구조에서는 정책 서버에서 각 이기종 보안 장비의 장비별로 실행 가능한 프로그래머블 보안 정책을 생성함으로써, 동일 AMN내에 있는 보안 장비에 정책을 전송하면 프로그래머블 라우팅 기법을 사용하여 관리하고 있는 모든 AMN내의 보안 장비에 프로그래머블 정책이 전달되어 수행된다.

정책 서버에서 전송한 프로그래머블 보안 정책이 모든 보안 장비에 전달되어 수행되기까지의 시간을 T2라 하고, 관리하고 있는 AMN의 개수를 Nm, 반복전송횟수를 Nr 이라고 할 때 T2는 다음과 같이 표현할 수 있다. 여기서 전송성공확률 Sh는 90%로 가정하고 모든 보안 장비에 보안 정책을 적용하기까지의 목표실패확률 Ft는 5%로 가정한다

$$T2 = \{ C + D \times Nm + E \} \times Nr \quad (6)$$

C는 정책 서버에서 정책을 생성하여 프로그래머블 보안 정책으로 변환하여 동일 AMN내의 보안 장비에게 전송할 때까지의 시간을 의미하며 식(7)에 의해 도출할 수 있다.

$$C = Tpc + Tpt + Tin \quad (7)$$

D는 동일 AMN내에서 보안 장비간의 정책전달 및 수행까지의 시간이라고 할 때 식(8)에 의해 도출할 수 있다.

$$D = (Ns \times Tpp) + (Ns - 1) \times (Tin + Gs) \quad (8)$$

E는 서로 다른 AMN에 위치한 보안 장비간의 정책 전달시 소요되는 시간을 의미하며, 식(9)에 의해 구할 수 있다.

$$E = Tout \times (Nm - 1) \quad (9)$$

식 (7), (8), (9)에 의해 T2는 다음과 같이 표현될 수 있다.

$$T2 = [(Tpc + Tpt + Tin) + \{ (Ns \times Tpp) + (Ns - 1) \times (Tin + Gs) \} \times Nm + \{ Tout \times (Nm - 1) \}] \times Nr \quad (10)$$

Fh와 Nr은 다음과 같이 계산할 수 있으며, 전송실패확률인 Fh가 목표실패확률인 Ft보다 작아야 하므로 이러한 조건에 부합하는 Nr을 도출한다.

$$Nr\ (\ Ns,\ Fh,\ Nm\ )\ =\ (\ Ns\ \times\ Nm\text{-}1\ )\ \times\ Fh,$$
$$Fh\ (\ Nr,\ Sh\ )\ =\ (\ 1\ -\ Sh\ )\ \times\ Nr$$

식(5)와 식(10)을 사용하여 T1과 T2를 산출한 결과를 그래프화하여 그림으로 도시한 결과는 다음과 같다. 그림 13은 Nm은 2로 가정하고 Ns의 개수를 증가시켰을 때 나타나는 결과를 그래프화한 그림이고, 그림 14는 Ns를 2로 가정하고 Nm의 개수를 증가시켰을 때 나타난 결과를 그래프화한 것이다.
그림 13에서 알 수 있듯이 AMN내에 보안 장비의 개수가 증가할수록 제안된 구조가 기존의 PBNM구조에 비해 정책전달 및 메시지 전송시 약 170% 정도의 성능향상이 있음을 알 수 있다. 또한 그림 14의 결과를 살펴보면 제안된 구조가 ANM의 개수가 증가할수록 PBNM 구조에 비해 성능이 평균적으로 약 33% 향상된 것을 볼 수 있다.

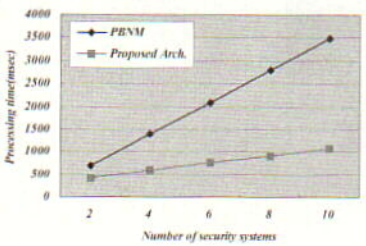

그림13. N_m=2, N_s가 증가할 때 T_1과 T_2값의 변화

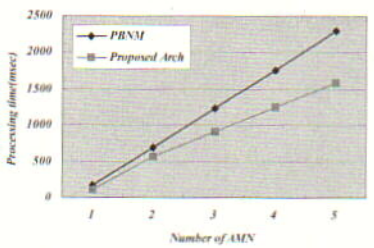

그림14. N_s=2, N_m이 증가할 때 T_1과 T_2값의 변화

V. 결 론

본 논문에서는 이기종 보안 장비를 보다 손쉽게 관리하고, 이기종 보안 장비간의 연동이 가능한 보안 관리 구조를 제안하였고, 기존의 네트워크 위에 프로그래머블 미들웨어를 관리계층으로 둠으로써 보다 지능화된 보안 관리 기법을 제안하였다. 제안된 보안 관리 구조를 사용하여 이기종 보안 장비간의 호환성을 제공할 수 있으며, PBNM 구조와 제안된 구조 간에 정책전달 및 메시지 전송시 소요되는 시간을 비교함으로써 프로그래머블 미들웨어를 통해 이기종 보안 장비 관리시 관리상의 부하를 줄일 수 있음을 검증하였다

참 고 문 헌

[1] "Conceptual model description of Active Security System for Next Generation Network V1.0," Information Security Research Div., ETRI, Korea, Jun. 2002.

[2] Y. Choi, "White paper: The compass of information security ESM Introduction," proceeding of 1st workshop on cyber terrors, Korea, 2002.

[3] OPSEC 개요, "Intro to OPSEC: OPSEC Software Development Kit," http://www.opsec.com/intro/sdkds.html.

[4] Gerhard Eschelbeck, "Active Security- A proactive approach for computer security systems," Journal of Network and Computer Applications 2000, pp.109-130, 2000.

[5] Alex Galis, et al., "A Flexible IP Active Networks Architecture," Proceedings of International Workshop on Active Networks, Oct. 2000.

[6] Alex Galis, et al., "Policy-Based Network Management for Active Networks," IEEE ICT 2001 Conference proceedings, Bucharest, Romania, 4-7 June 2001.

[7] Kei Kato, et al., "Application of Active Networking to policy networking," Japan.

[8] D. Galand, O. Marce, "Active Router Information in Routing Protocols," IETF Internet Draft, 2000.

[9] 채기준, "최종보고서: 센서 통신 구조 최적화 모델링 연구", 이화여대, Funded by ETRI, 2002

[10] J. Postel, "Internet Protocol," IETF RFC 791, 1981.

[11] S. Deering, R. Hinden, "Internet Protocol,

Version 6 (IPv6)," IETF RFC 2460, 1998.
[12] D. Scott Alexander, Bob Braden, Carl A. Gunter, Alden W. Jackson, Angelos D. Keromytis, Gary J. Minden, and David Wetherall, "Active Network Encapsulation Protocol (ANEP)," http://www.cis.upenn.edu/switchware/ANEP/docs/ANEP.txt, 1997.
[13] Jiyoung Lim, "Design of security enforcement engine for active nodes in active networks," The International Conference on Information Networking (ICOIN) 2003, vol.3, 2003.

김 명 은(Myung-Eun Kim) 정회원

1996년 숭실대학교 소프트웨어공학과 공학사
1998년 서강대학교 전산학과 공학석사
1998년 4월 ~ 1999년 9월 동양시스템하우스 근무
2000년 9월 ~ 현재 한국전자통신연구원 정보보호연구본부 네트워크보안구조팀 연구원

<주관심분야> 네트워크보안, 프로그래머블네트워크, 차세대 인터넷 등

오 승 희(Seung-Hee Oh) 정회원

1999년 전북대학교 컴퓨터과학과 이학사
2001년 이화여자대학교 컴퓨터학과 공학석사
2001년 ~ 현재 한국전자통신연구원 정보보호연구본부 네트워크보안구조연구팀 연구원

<주관심분야> 정보보호, 차세대네트워크보안, Active Network 등

김 광 식(Kwang-Sik Kim) 정회원

1991년 경북대학교 전자공학과 졸업
1997년 충북대학교 정보통신공학과 석사
2000년 충북대학교 정보통신공학과 박사
1991년 1월~2000년 6월 한국전자통신연구원 무선방송연구소 선임연구원
2000년 11월~2002년 2월 (주)투니텔 연구소장
2002년 3월~현재 한국전자통신연구원 정보보호연구본부 선임연구원

<주관심분야> CDMA이동통신, 네트워크정보보호

남 택 용(Taek-Yong Nam) 정회원

1987년 충남대학교 계산통계학과 이학사
1990년 충남대학교 대학원 계산통계학과 이학석사
1987년 ~ 현재 한국전자통신연구원 정보보호연구본부 네트워크보안구조연구팀 팀장

<주관심분야> 정보보호, 능동보안, 차세대네트워크 구조 등

손 승 원(Sung-Won Sohn) 정회원

1984년 경북대학교 전자공학과 공학사
1994년 연세대학교 전자공학과 공학석사
1999년 충북대학교 컴퓨터공학과 공학박사
1983년 ~ 1986년 삼성전자 연구원
1986년 ~ 1991년 LG 전자(주) 중앙연구소 HI8mm 캠코더 팀장
1991년 ~ 현재 한국전자통신연구원 정보보호연구본부 네트워크보안연구부 부장

<주관심분야> 네트워크보안, 차세대인터넷, Active Internet 등