主題

인터넷 보안: 가상사설망, 방화벽 그리고 침입탐지시스템

㈜ 퓨쳐시스템 임채훈, 강명희

차 례

요 약

인터넷은 본질적으로 신뢰할 수 없는 네트웍들의 집합체로, 정보의 흐름을 통제하기가 대단히 어렵기 때문에, 인터넷에 산재한 자원을 충분히 활용하면서, 내부의 중요한 자원을 인터넷으로부터 보호해 줄 수 있는 인터넷 보안이 가장 심각한 문제로 대두되고 있다. 본 고에서에서는 현재 인터넷 보안의 주류를 이루는 있는 가상사설망(VPN : Virtual Private Network)과 방화벽(Firewall), 그리고 방화벽의 부족한 부분을 보강해줄 수 있는 침입탐지 시스템(IDS: Intrusion Detection System)에 대한 기본 구현 기술들을 비교 분석해 본다.

I. 서 론

인터넷은 이제 거의 모든 정보를 가장 빠르게, 그리고 손쉽게 얻을 수 있는 정보의 보고로 자리잡게 되었고, 이에 따라 대부분의 회사나 조직체들은 보다 나은 서비스를 제공하고 보다 앞선 경쟁력을 갖추기 위해 대부분 사내망을 인터넷에 접속시키고 있다. 그러나 인터넷은 본질적으로 신뢰할 수 없는 네트웍들의 집합체로, 정보의 흐름을 통제하기가 대단히 어렵다. 따라서 인터넷에 산재한 자원을 충분히 활용하는 반면 내부의 중요한 자원을 인터넷으로부터 보호해 줄 수 있는 인터넷/인트라넷 보안이 가장 심각한 문제로 대두되고 있다.

인터넷 보안은 크게 엑세스 제어 서비스와 통신 보안 서비스의 적절한 조합에 의해 달성될 수 있다. 엑세스 제어는 컴퓨터/네트웍 자원의 접근제한을 통해 외부 혹은 내부의 사용자들로부터 보호하는 기술이며, 통신 보안은 사용자 인증이나 데이터 무결성, 데이터의 비밀보장 등의 암호 기술들을 이용해 인터넷에 유통되는 정보를 불법적인 사용자들로부터 보호하기 위한 것이다. 대부분의 인터넷 보안 제품은

다양한 보안 프로토콜들을 이용하여 엑세스 기능을 구현한 것으로 볼 수 있으며, 현재 인터넷 보안의 주류를 이루는 것이 **가상사설망(VPN : Virtual Private Network)**, 방화벽(Firewall)과 침입탐지시스템(Intrusion Detection System)이다. VPN은 적절한 암호기술을 이용하여 한 조직의 내부 사용자들이 사내나 사외에서 서로 안전하게 통신할 수 있는 채널을 형성해 주며 또한 필요한 접근제어 기능을 제공해 준다. 반면 방화벽은 기본적으로 인터넷과 자신의 네트워 사이에 위치하여 불특정 다수의 인터넷 사용자에 대한 접근제어를 통해 자신의 네트워을 보호하고자 하는 일차적인 방어벽의 역할을 한다. 그러나 방화벽에 의한 접근통제는 주로 TCP/IP 프로토콜 헤더의 정보나 약간의 응용데이타 해석 정도를 이용하므로 복잡한 네트웍 공격이나 호스트에 대한 공격 등을 차단하는 데는 한계가 있다. 또한 내부의 합법적인 사용자에 의한 불법적인 자원 남용에 대해서는 그 역할을 전혀하지 못한다. 이러한 방화벽의 부족한 부분을 보강해줄 수 있는 것이 침입탐지 시스템(IDS: Intrusion Detection System)으로 로컬 네트웍이나 호스트에 위치하여 보다 정밀한 분석을 통해 다양한 공격이나 불법행위 등을 탐지하여 대응방안을 세울 수 있게 해준다.

본 고에서는 VPN과 방화벽을 구축하는 기본 기술들을 비교 분석하고, 기존의 침입탐지 시스템에서 사용되는 침입탐지 기술 및 네트웍 기반 침입탐지(Network-based IDS)와 호스트 기반 침입탐지(Host-based ID)의 장단점 등에 대해 간략히 살펴본다.

Ⅱ. VPN

VPN은 인터넷과 같은 공중망을 이용하여 가상의 사설망을 구성하는 기술로 기존의 전용선을 이용한 사설망에 비해 훨씬 저렴한 비용으로 보다 접속력이 뛰어나면서도 안전한 망을 구성할 수 있다는 면에서 인터넷의 활성화와 더불어 각광을 받기 시작한 보안 솔루션의 하나이다. VPN은 상호 네트워킹 시나리오에 따라 통상 다음의 세가지 형태로 분류하며 각각의 경우에 서로 다른 보안정책 필요하고 다른 구현 기술이 존재할 수 있다:

- ◆ Intranet VPN: 본사와 지사간의 네트워킹
- ◆ Remote Access VPN: 본사와 원격지 허가받은 사용자(직원)간의 네트워킹
- ◆ Extranet VPN: 본사와 사업 파트너, 고객등과의 네트워킹

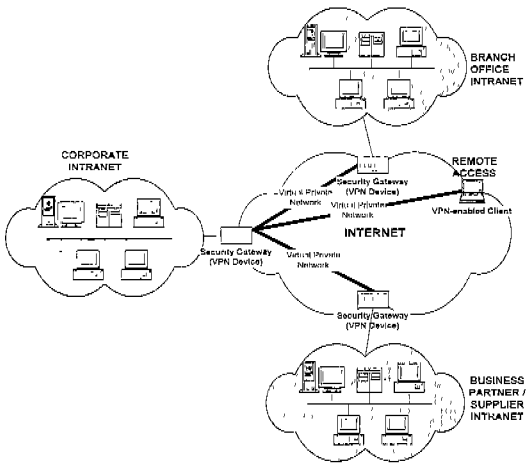

그림 1. Virtual Private Network

1. VPN 시나리오

Intranet VPN : Intranet VPN은 주로 인터넷에 연결된 본사와 각 지사를 안전하게 연결하는 LAN-to-LAN 접속을 위해 사용되는 거의 반 영구적인 사설망이다. 통상 본사의 입장에서 보면 지사는 신뢰할 수 있는 것으로 보기 때문에 주로 인터넷 구간만을 보호하는 것이 일반적이다. 그러나 많은

경우외부보다는 내부에 보안상의 위협요소가 더 많이 존재한다. FBI와 CSI(Computer Security Institute)의 공동 조사에 따르면 실제로 보안침해의 반 이상이 내부에서 일어난다고 한다. 따라서 고의든 실수든 내부자에 의한 정보누출이 더 심각하다고 판단되거나, 혹은 같은 LAN 내부에서도 개인이나 부서에 따라 내부자원의 접근제한을 둘 필요가 있다면, End-to-End 암호화, 사용자/응용 기반의 인증기법등에 의한 내부 보안용으로도 사용될 수 있다.

Remote Access (Dial-up) VPN : 기존의 전화선 등을 이용한 원격지 dial-up 엑세스는 많은 조직체들이 modem pool을 유지하고 장거리 전화비용을 부담해야 했으나, 인터넷을 backbone으로 이용함으로써 보다 저렴하고 구현 및 관리가 쉬운 Remote Access VPN을 구성할 수 있다. Remote Access VPN 시나리오는 mobile-to-LAN 접속으로 그 기본 기능은 이동 사용자로 하여금 사내의 자원에 대한 안전한 접근을 허용함으로써 업무의 능률을 높이고자 하는 것이다. 따라서 이 경우는 대부분의 응용에서 사용이나 관리상의 용이성이 보다 중요한 고려사항이 될 수 있다. 이동 단말의 특성상 클라이언트는 사용하기 쉽게 동작해야 할 것이고, 서버의 입장에서는 Dial-up 엑세스의 특성상 사용자 계정/과금 등과 관련한 사용자 단위의 인증을 기본적으로 제공해야 하므로 기본적인 데이터 암호화/인증 뿐만 아니라, 사용자 계정/ 과금과 관련된 각종 통계자료의 중앙 집중식 관리가 필요하다. 이러한 목적으로 인터넷 표준인 RADIUS (Remote Authentication in Dial-In User Service)가 가장 널리 사용된다.

Extranet VPN : Extranet VPN은 보안정책이 서로 다른 이질적인 subnet들을 상호 연결시켜 주는 Business-to-Business VPN인 만큼, Intranet VPN에 비해 보안상 위험성이 높고, 보다 정교한 엑세스 제어가 필요하며 상호 연동성을 고려해야 하는 등 가장 구성하기 어려운 VPN이라 할 수 있다. End-to-End 보안 및 사용자 단위의 엑세스 제어가 기본적으로 제공되어야 하고, 각 subnet의 경계에는 직접 접속을 피하고, 보다 정교한 엑세스 제어가 가능한 Application Proxy를 설치하여 subnet들을 선택적으로 격리시키는 것이 바람직하다. 그리고 컴퓨팅 환경이 서로 다른 조직체를 상호 연결시켜 주어야 하므로, 상호연동을 위해 표준 보안 프로토콜을 사용해야 하며, 보안정책에 대한 협의와, 가능한 다양한 보안 메카니즘을 지원하는 제품이 유리할 것이다. 특히 Extranet VPN은 한 subnet에서의 취약 요소가 다른 subnet의 보안에 영향을 미치지 않도록 해야 한다.

2. VPN 구현 기술

VPN 구현 기술은 크게 터널링, 암호화/인증, 그리고 엑세스제어 로 나뉘어 진다. 터널링은 VPN내의 두 호스트간에 가상경로을 설정해 주며, 암호화/인증은 이 가상경로를 다른 인터넷 사용자들로부터 격리시켜 안전한 채널로 만들어 준다. 그리고 엑세스 제어는 VPN내의 자원에 대한 접근 통제를 통해 Intranet/Extranet 보안 정책을 구현하게 해 준다. VPN 구현에 가장 널리 사용되는 Tunneling Protocol로는 다음의 4가지가 있으며, VPN 기능은 통상 라우터나 방화벽에 내장되거나 전용 VPN 장비로 구현되기도 한다.

- ◆ PPTP (Point-to-Point Tunneling Protocol: Layer 2 tunneling)
- ◆ L2TP (Layer 2 Tunneling Protocol: Layer 2 tunneling)
- ◆ IPSEC (IP Security protocol: Layer 3 tunneling)
- ◆ SOCKS V5 (Layer 5 tunneling)

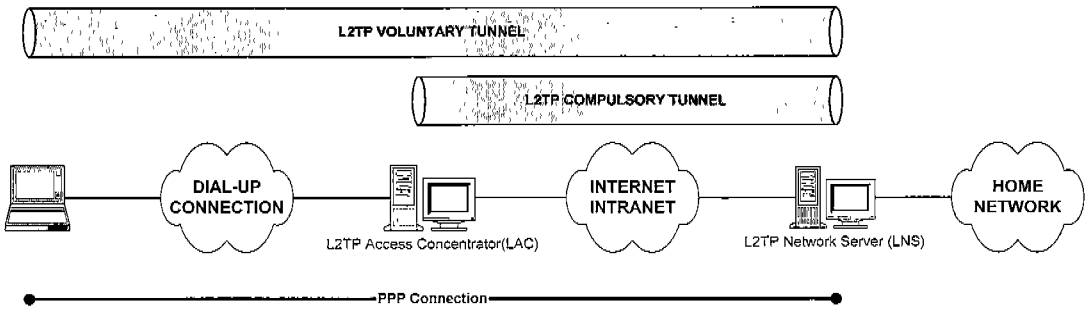

그림 2. Voluntary Tunnel vs. Compulsory Tunnel

PPTP / L2TP : . Microsoft 사의 PPTP와 인터넷 표준인 L2TP (RFC 2661)는 인터넷에서 remote access VPN을 구성하는데 가장 널리 사용되는 클라이언트/서버 기반의 Tunneling Protocol이다. 이 두 프로토콜은 모두 Layer 2의 PPP 트래픽에 대한 Encapsulation을 통해 두 지점간의 터널을 생성, 관리, 소멸시켜주는 것이 기본 기능이며, 보안은 대부분 PPP에서 제공하는 보안기능에 의존하므로 L2TP에서는 보다 강한 보안을 위해 IPSEC을 사용하도록 권고하고 있다.

Layer 2 Tunneling은 크게 tunnel initiator에 따라 **Voluntary Tunneling**과 **Compulsory Tunneling**으로 나눌 수 있다. **Voluntary Tunneling**은 client-initiated Tunneling으로 클라이언트가 직접 Tunnel 서버(보통의 Remote Access Server(RAS), IETF 용어로는 PPTP/L2TP Network Server(PNS/LNS)로 불림)와 Tunnel을 형성하므로 클라이언트간의 End-to-End Tunnel이 형성되며, 클라이언트에 PPTP/L2TP 프로토콜이 탑재되어 있어야 한다. 반면 **Compulsory Tunneling**은 ISP-initiated Tunneling으로 인터넷 서비스 제공자(ISP) Remote Access Switch가 클라이언트를 대신해서 터널을 열어 주는 경우로 클라이언트에 Tunneling Protocol이 탑재되어 있지 않은 경우나, ISP에서 VPN 서비스를 제공해 주는 경우에 사용되며, PAC/LAC-PNS/LNS간에 Tunnel이 형성된다.

PPTP와 L2TP는 유사한 Layer 2 Tunneling 서비스를 제공하지만 이 둘 사이에는 몇 가지 중요한 차이점이 있다:

- ◆ 두 프로토콜 모두 PPP 트래픽을 encapsulation하기 때문에, IP, IPX, NetBEUI, AppleTalk 등의 다양한 상위 로컬 네트웍 프로토콜을 사용할 수 있다. 그러나 PPTP는 transit internetwork이 IP 네트웍일 것을 요구하는 반면, L2TP는 Packet-Oriented Point-to-Point 접속을 제공하는 네트웍만 보장되면, 어떤 전송 프로토콜 상에서도 사용 가능하다 (e.g. IP, Frame Relay PVCs, ATM VCs 등).
- ◆ PPTP는 End-Point들 사이에 하나의 Tunnel만을 지원하나, L2TP는 Multiple-Tunnel을 허용한다. 따라서 L2TP를 사용하면 QoS에 따라 서로 다른 Tunnel을 이용할 수 있다.
- ◆ L2TP는 헤더 압축 및 Tunnel-End-Point 인증 (패킷단위의 인증이 아니라, Tunnel End-Point들의 Identity에 대한 인증) 기

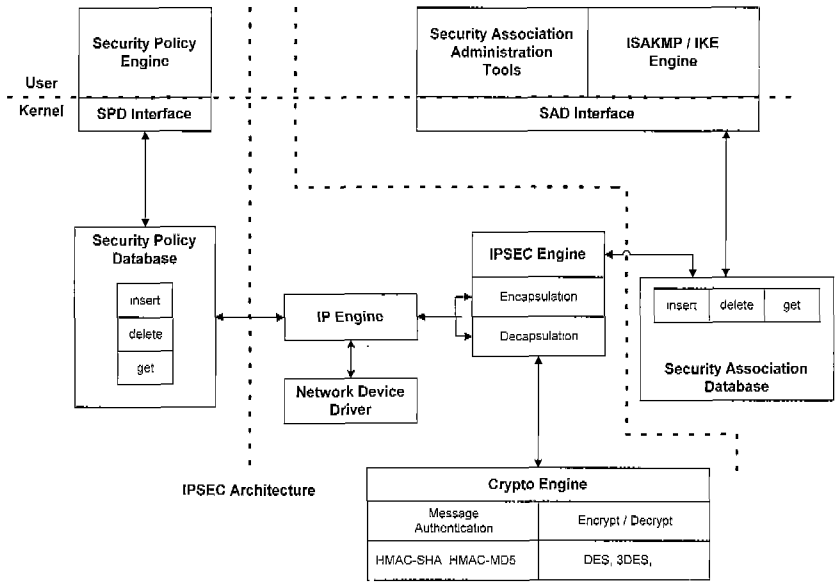

그림 3. IPSEC 구현 모델 예

능을 제공하지만 PPTP에는 이런 기능이 없다. 두 프로토콜 모두에서 사용자 인증(PAP, CHAP, MS-CHAP, EAP)이나 데이터 암호화/압축 (CCP, ECP) 등의 보안 기능은 PPP에서 제공하는 것을 사용한다.

IPSEC : IPSEC은 최근에 일련의 표준화 작업이 완성된 IP 계층의 보안을 위한 인터넷 표준으로 VPN 구현에 가장 널리 사용되는 기술이다. IPSEC은 크게 IP 헤더를 포함하는 전체 패킷에 대한 인증 기능을 제공하는 AH(Authentication Header), IP header 이외의 payload에 대한 암호화/인증 기능을 제공해 주는ESP(Encapsulating Security Payload) 헤더, 그리고 AH/ESP를 포함한 각종 인터넷 보안 서비스에 필요한 Security Association Negotiation 및 Key Management를 담당하는 ISAKMP/IKE(Internet Security Association and Key Management Protocol / Internet Key Exchange)로 구성된다.

IPSEC을 구현상의 관점에서 보면, 크게 IP address를 바탕으로 보안 정책을 담당하는 Policy Engine, Security Association(SA) / Key Management를 담당하는 ISKMP/IKE Engine, AH 및 ESP Encapsulation을 담당하는 IPSEC Engine, 그리고 IPSEC 및 ISAKMP/IKE Engine에 필요한 각종 암호연산 라이브러리 및 API로 구성되는 Crypto Engine으로 나눌 수 있다.

IPSEC의 동작은 크게 **Transport Mode**와 **Tunnel Mode**로 나뉜다. Transport Mode는 End-to-End Tunneling에, Tunnel Mode는 Gateway-to-Gateway Tunneling에 주로 사용되며, 각 Mode는 보안 정책에 따라 AH, ESP 혹은 AH with ESP 등의 다양하게 적용된다.

IPSEC의 Layer 3 Tunneling과 PPTP/L2TP의 Layer 2 Tunneling의 장단점을 비교해 보면 다음과 같다.

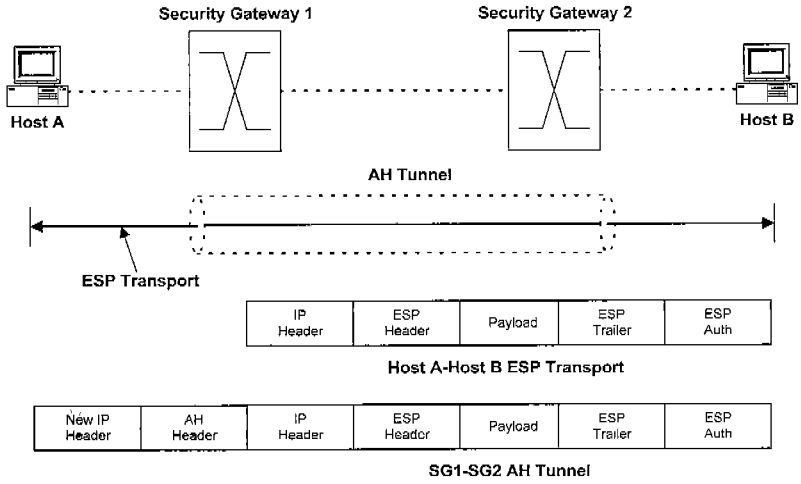

그림 4. IPSEC 프로토콜 동작 과정

◆ PPTP/L2TP는 IP 아닌 트래픽에 대해서도, Tunneling이 가능하지만 IPSEC은 IP Tunneling만을 지원한다. 따라서 IP 외의 다양한 네트웍 프로토콜을 사용하는 인트라넷에 대한 Remote Access를 위해서는 PPTP/L2TP를 지원해야 한다.

◆ PPTP/L2TP는 사용자 단위의 인증, 동적 주소 할당 등이 가능하나, IPSEC은 기본적으로 Host-to-Host 인증을 지원하며, fixed, routable IP address를 가정한다. 엄격한 엑세스 제어를 위해서는 사용자 인증이 필수적이므로 대부분의 IPSEC 개발 업체들은 자체의 사용자 인증 솔루션을 제공하고 있다.

◆ IPSEC은 패킷 단위의 암호화/인증, 자동화된 키 관리 등의 네트웍 계층 보안을 위한 기본 구조를 제공하지만, PPTP/L2TP는 단지 PPP Tunnel의 생성, 소멸, 관리를 주로 하며 PPP가 제공하는 외의 자체적인 보안 기능이 거의 없다. 따라서 PPTP/L2TP는 보안을 위해 IPSEC과 함께 사용할 것을 권장하고 있다.

SOCKS V5 : SOCKS V5는 authenticated firewall traversal을 위한 순수한 Circuit-Level Proxy였던 SOCKS V4에 Client Authentication, Encryption Negotiation, UDP Proxy 등 다양한 보안기능을 보강한 것으로 같은 Seesion Layer Security Protocol인 SSL/TLS와 결합되어 사용될 수 있다 [7, 8, 9]. SOCKS는 물론 어떤 하위계층의 Tunneling Protocol 상에서도 사용될 수 있으며, Session Layer Proxy로서 동작하므로 IPSEC보다 훨씬 정교한 엑세스 제어 및 로깅 기능을 제공할 수 있다. IPSEC 만큼 널리 알려지지는 않았으나, 최근에는 복잡해지는 네트웍에 대한 관리 목적으로 사용되고 있다. SOCKS V5를 이용한 VPN은 하위계층의 Tunneling Protocol에 비해 뛰어난 엑세스 제어 기능을 제공할 수 있기 때문에, 보다 복잡하고 정교한 보안 정책 관리가 요구되는 Extranet VPN과 같은 경우에는 위력을 발휘할 수 있다.

3. VPN 기술 동향

현재 대부분의 VPN 제품들은 라우터나 방화벽 기반의 비표준 프로토콜을 사용한 제품이나,

표 1. VPN 기술 특징 비교

	PPTP	L2TP	IPSEC	SOCKS V5
Standardization	Microsoft	RFC 2661	RFC 2401-2410	RFC 1928, 1929, 1961
OSI Layer	Layer 2	Layer 2	Layer 3	Layer 5
Mode	Client-Server	Client-Server	Peer-to-Peer	Client-Server
Protocols supported	IP,IPX,NetBEUI, AppleTalk, etc.	IP,IPX,NetBEUI, AppleTalk, etc.	IP	TCP,UDP/IP
Tunnel Services	Single PPP tunnel, Per-connection	Multiple PPP tunnels, Per-connection	Multipoint tunnels, Per-SA	Session-by-session basis
User authentication	None (by PPP)	None (by PPP)	None	Provided
Data authentication /encryption	None (encryption provided by PPP)	None (encryption provided by PPP) (refer to IPSEC)	Per-packet auth/enc by AH/ESP headers	Per-message auth/enc through GSS-API
Key management	None	None (refer to IPSEC)	ISAKMP/IKE	GSS-API/SSL
Access control	None (on server)	None (on server)	Packet filtering	Packet/content filtering, proxying
Best practices	Remote access	Remote access	LAN-to-LAN intranet	Extranet
Providers	Remote access vendors	Remote access vendors	Firewall, router, VPN vendors	Firewall, extranet VPN vendors

PPTP/L2F를 이용한 dial-up VPN 제품들이다. 그러나 최근에는 VPN 프로토콜의 표준으로 IPSEC이나 L2TP가 완성됨에 따라 대부분의 제품들이 상호 연동을 위해 IPSEC/L2TP를 기본적으로 지원하고 있는 추세이다. 또한 LAN-to-LAN VPN은 성능이 무엇보다도 중요한 만큼 IPSEC 기반의 전용 하드웨어 VPN 제품들도 속속 출시되고 있다. 다음 표 1은 위에서 살펴보았던 VPN 구현 기술들을 여러 가지 측면에서 비교해 본 것이다.

Ⅲ. 방화벽

방화벽은 통상 인터넷과 내부망의 경계 부분에 존재하여 정보의 흐름을 통제하는 기능을 하며, 현재 대부분의 방화벽은 다음의 4가지 기술의 일부 혹은 조합으로 구성된다[1, 2, 3]:

- ◆ Packet Filtering
- ◆ Application-Level Proxy(or Dedicated Proxy)
- ◆ Circuit-Level Proxy(or Generic Proxy)
- ◆ Stateful Packet Inspection

1. 방화벽 구현 기술

Packet Filtering : Packet Filtering은 가장 간단한 형태의 방화벽의 기능으로, 통상 IP 헤더와 상위 프로토콜 (TCP, UDP, ICMP 등) 헤더의 정보(통상 Source/Destination IP Address, Port Number)만을 이용해 미리 설정된 엑세스 제어 규칙에 따라 해당 패킷의 통과여부를 결정한다. 대부분의 요즘 라우터에는 기본으로 내장되어 있으며, 상용 방화벽 제품들도 어떤 형태로든 Packet Filtering 기능을 기본적으로 제공하고

있다. Packet Filtering은 최소한의 엑세스 규칙만을 적용시켜 처리하기 때문에, 가장 빠른 속도를 낼 수 있고, 구현이 간단하며, 사용자에겐 투명성이 보장된다는 장점이 있다. 그러나 복잡한 네트웍이나 서비스에 대한 정교한 엑세스 규칙을 구현하기가 매우 어려우며, 제한된 정보를 바탕으로 하는 간단한 엑세스 규칙에 따라 접속허가 기능만을 제공하기 때문에, 일단 Packet Filtering을 통과한 위험한 서비스에 대해서는 내부와 외부 망 사이에 접속이 이루어져 내부 자원을 효과적으로 보호해야 하는 방화벽의 가장 기본적인 기능면에서는 많은 취약점을 가진다.

Application-Level Proxy : Application-Level Proxy는 특정 응용 서비스에 대해 내부망과 외부망을 연결시켜 주는 중간 매개자 역할을 하는 것으로, 클라이언트/서버간의 직접적인 접속 대신, 클라이언트를 대신해서 서버와 접속하여 클라이언트/서버 사이의 통신을 중계해 준다. Application-Level Proxy와 다른 종류의 방화벽간의 가장 큰 차이는 Application Proxy는 각각의 응용 프로토콜을 Application Proxy로 구현한다는 것이다. 따라서 응용 프로토콜에 대한 완전한 이해를 바탕으로 한 만큼 보다 정교한 엑세스 제어가 가능하고 그에 따른 상세한 로깅이 가능하다. 또한 Application Proxy는 외부에 알려진 것은 단지 Proxy 서버뿐이므로 내부 망의 시스템 구성(e.g., IP 주소들)을 외부로부터 완전히 숨길 수 있다.

그러나 특정 응용 서비스마다 Proxy 서버가 필요하므로 다양한 인터넷 서비스를 제공하기 위해서는 모든 서비스에 대한 Proxy를 구현해야 하며, 따라서 새로운 인터넷 서비스가 생길 경우는 상당한 시간이 지나야만 해당 Proxy가 제공될 수 있다. 또한 대부분의 Application Gateway Firewall들은 사용자 인증 기능을 제공하는 반면, 이 경우 매 응용 마다 인증이 일어나야 하므로 사용자가 불편을 느낄 수 있으며 대부분의 경우 클라이언트 소프트웨어나 사용법에 변경이 요구되며, 클라이언트와 서버 사이의 모든 트래픽을 분석하고 재전송 해 주어야 하므로 다른 기술에 비해 성능은 훨씬 떨어진다.

Circuit-Level Proxy : Circuit-Level Proxy는 Application Proxy와 유사한 기능을 제공하지만 Application 계층이 아니라, 통상 Session 계층에서 동작한다. 즉 Circuit-Level Proxy는 Session 단위로 클라이언트/서버간의 가상의 circuit을 형성하여 데이터를 전송해 준다. 따라서 Application Proxy와 Circuit Proxy의 차이/장단점은 응용 계층과 세션 계층 처리상의 차이로부터 온다. 즉 Application Proxy는 개별 응용마다 개발되어야 하는 전용 Proxy이지만 Circuit Proxy는 응용 서비스와 무관한 세션계층에서 동작하므로 대부분의 프로토콜을 자동으로 지원해 주는 Generic Proxy이다. 반면 응용 프로토콜을 해석하지 않기 때문에, 특정 응용 기반의 엑세스 제어 및 로깅 기능은 제공하지는 못한다. Circuit-Level Proxy의 대표적인 것으로 인터넷 표준 SOCKS V5를 들 수 있다.

Stateful Inspection : Packet Filtering은 각 패킷을 정해진 엑세스 규칙에 따라 독립적으로 허용/거부하는 방법이다. 반면 **Dynamic Packet Filtering**은 접속에 대한 상태를 관측하여, 패킷의 통과 여부를 결정하는 동적 엑세스 규칙을 적용시킨다. Dynamic Packet Filtering의 보다 발전된 형태로 **Stateful Packet Inspection**이 있다 (Check Point Software Technology사의 Firewall-I). 단순히 헤더 정보만 가지고 필터링 하는 대신 Packet Inspection은 헤더 뿐만 아니라 패킷 내용까지 해석하여 엑세스 규칙을 적용시킨다 (Content Filtering). 즉 네트웍 계층에서 응용 계층까지 모든 프로토콜을 이해하

는 Inspection Module에서 모든 계층으로로부터 관련 정보를 얻은 후 이를 바탕으로 엑세스 제어를 수행한다. 특히 과거의 패킷에 대한 정보, 즉 상태 정보를 지속적으로 유지하여 현재 패킷의 통과 여부를 결정한다.

Packet Inspection은 기본적으로 Packet Filtering기술을 적용하되 (Application Proxy를 모방하여) 적은 부하로 응용 데이터를 해석하여 Filtering을 하는 기술로, Application Proxy보다 훨씬 나은 성능을 가지면서도 이에 버금가는 보호능력을 가지며, 사용자/응용에 투명한 Packet Filtering의 장점을 지닌다고 볼 수 있다.

2. 방화벽 제품 기술 동향

현재 대부분의 방화벽 제품들은 기본적으로 Packet filtering 기능에 Application Proxy나 Stateful Inspection 등 다양한 기술들을 지원하며 응용 서비스에 따라 적절한 기술들을 적용한다. Stateful Inspection의 강점은 융통성과 우수한 성능에 있으며, Application Proxy는 융통성과 성능면에서는 Stateful Inspection에 떨어지지만 응용에서 보다 정교한 엑세스 제어와 로깅이 가능하다는 것이 장점이라 할 수 있다. 반면 Stateful Inspection은 실제로 모든 응용 프로토콜을 완전히 이해하는 Inspection Module을 구현하기는 어렵고, 또한 엑세스 규칙 설정에 상당한 전문지식이 요구되므로 직접적인 접속으로 인한 위험한 서비스가 방화벽을 통과하게 될 가능성도 있다.

방화벽 제품의 안전성이나 신뢰도는 해당 기술을 얼마나 정교하게 사용/관리하기 쉽게 구현하느냐에 의존한다고 보여진다. 방화벽 제품들의 경쟁력은 방화벽 자체의 기술보다 간단한 시스템 관리, 사용자 투명성, 간편한 모니터링, 외부 인증 기법 지원 등과 같은 사용/관리상의 편리성이나 확장성과, VPN 통합, 침입탐지, 보안상 취약성/바이러스 검사등과 같은 기능을 통합 관리하는 통합 엑세스 관리 시스템(Integrated Access Management System)에 초점이 맞추어지고 있는 것 같다. 즉 단일 제품으로서의 방화벽의 중요성은 퇴색되어 가고 있으며, 대부분의 방화벽 제품들이 점점 방화벽 기능에 VPN 기능 및 일부 침입탐지 기능까지를 탑재한 통합 보안 솔루션을 지향하고 있다.

Ⅳ. 침입탐지 시스템(IDS: Intrusion Detection System)

방화벽은 인터넷의 접속점에 위치여 사내망으로 오가는 모든 트래픽을 감당해야 하는 만큼 그 성능상의 제한으로 인해 트래픽에 대한 많은 분석을 하기는 어렵다. 또한 내부의 침투자에 의한 컴퓨터/네트웍 자원에 대한 보호에 있어서는 전혀 역할을 할 수가 없다. 이러한 방화벽의 부족한 부분을 보강해 줄 수 있는 것으로 침입탐지 시스템(IDS: Intrusion Detection System)이 있다. IDS는 로컬 네트웍이나 호스트에 위치하여 보다 정밀한 분석을 통해 네트웍을 통한 공격이나 시스템에 대한 불법행위 등을 탐지하여 대응방안을 세울 수 있게 하는 역할을 한다.

1. 침입탐지 기술

침입탐지 방법은 크게 시스템/자원의 정상적인 사용과 비교하여 비정상적인 행위를 탐지하는 Behavior-based Intrusion Detection (Anomaly Detection)과 이미 알고있는 공격에 대한 지식을 바탕으로 시스템/자원의 남용을 탐지하는 Knowledge-based Intrusion Detection (Misuse Detection)으로 나눌 수 있다. 각각의 장단점들이 있기 때문에, 현재는 두 방법을 결합하여 상호 보완적으로 사용하는 것이 일반적인 추세

이다.

Knowledge-based Intrusion Detection (Misuse Detection) : Knowledge-based ID는 특정 공격이나 시스템 취약성에 대해 축적된 정보(Knowledge Base)를 이용하여 침입을 탐지하여 경고를 낸다. 즉 Knowledge base에 등록된 공격 시나리오에 해당되는 시도만을 침입으로 간주하고 경고를 발한다(Virus Detection과 유사). 따라서 이 방식은 정확도(accuracy)는 상당히 높은 편이나 완전도(completeness)는 Knowledge base가 얼마나 많은 공격유형에 대해서 대처하고, 최신 정보를 얼마나 자주 갱신하느냐에 의존한다. 갱신되지 않은 최신 공격이나 알려지지 않은 공격에 대해서는 속수무책이므로 후술하는 Behavior-based ID에 비해 완전도 낮은 편이다. Knowledge-based ID의 단점은 Knowledge base의 유지보수가 쉽지 않다는 것이다. 즉 알려진 공격에 대한 정보를 수집하거나 새로운 공격유형 혹은 취약점들을 최신의 것으로 유지하는 일이 쉽지 않고, 합법적인 사용자가 권한을 남용하는 내부자 공격 유형을 탐지하는 것도 매우 어렵다.

Knowledge-based ID는 Knowledge base의 구축 형식에 따라 관측된 audit event를 적절한 형식으로 가공하여 비교하게 되는데 가장 간단하고 효율적이어서 commercial IDS에서 가장 널리 사용되는 방법이 Signature Analysis이다 (이런 이유로 IDS 업계에서는 Knowledge-based Intrusion Detection이라는 용어보다 Misuse Detection 혹은 **Signature-based Intrusion Detection**이라는 용어를 더 자주 사용한다). 여기서는 각 공격 시나리오가 그 공격에 고유한 일련의 audit event들의 sequence로 기술되거나 공격에 대한 시스템의audit trail에서 발견되는 데이터의 pattern으로 기술된다. 따라서 침입여부는 간단한 pattern matching algorithm을 이용하여 관측된 event나 audit trail을 Knowledge Base의 내용과 비교함으로써 탐지될 수 있다. 그러나 이 방법이 간단하고 효율적이어서 상용 제품들에서 널리 사용되고 있기는 하지만, 조금만 변경된 공격에 대해서도 그 signature가 다를 수 있고 이 경우는 침입을 탐지하지 못하므로 침입탐지의 Accuracy가 그렇게 높지는 못하다.

학계나 연구계에서 활발히 연구되고 있는 Knowledge-based ID 기술로 Expert System을 이용한 **Rule-based Intrusion Detection** 기술이 있다. 여기에는 공격을 좀 더 상위레벨에서 abstract하게 기술한 Rule Base (a set of rules describing attacks)가 있고, inference engine에서 audit event로부터 유도된 fact와 rule base의 rule에 대한 binding analysis를 통해 침입여부를 판단한다. 이 기술은 아직도 활발히 연구되고 있는 분야이나 그 복잡도나 성능 등의 문제로 아직까지 상용 제품에는 적용되지 못하고 있고 prototype형태에만 사용되고 있다.

Behavior-based Intrusion Detection (Anomaly Detection) : Knowledge-based ID가 알려진 공격이나 시스템의 취약점을 이용하는 반면, Behavior-based ID에서는 평소의 사용자/시스템의 예상되는 행위에 대한 정보를 이용하여 이로부터 벗어난 정도를 바탕으로 침입을 탐지한다. Knowledge-based ID가 알려진 공격에 대해서만 침입 탐지가 가능하고 내부자 공격을 탐지하는 것이 어려운 것에 비해 Behavior-based ID는 알려지지 않은 혹은 새로운 공격도 탐지가 가능하고 내부자/외부자 공격의 구분없이 동일한 기준으로 침입을 탐지할 수 있다는 장점이 있다.

정상적인/합법적인 행위에 대한 사용자/시스템 프로파일은 다양한 방법으로 축적된 참조정보(Reference Information)를 바탕으로 구축되며, 침입은 현재의 사용자/시스템의 행위을 사용자

/시스템 프로파일과 비교하여 일정 수준 이상의 이탈행위가 있을 때, 침입이 일어난 것으로 판단한다. 정상/비정상에 대한 명확한 경계를 정하는 것은 쉽지 않으므로 침입여부의 정확도가 낮을 수 있다. Detection 알고리즘을 좁은 범위로 국한시켜 훈련을 시킨다면 정상적인 행위에 대해서도 False Alarm을 남발할 수 있고, 반면 넓은 범위로 확대하면 침입탐지 기능이 떨어질 수도 있다. 또한 사용자/시스템의 행위가 시간에 따라 변할 수 있으므로 사용자/시스템 행위 프로파일을 주기적으로 갱신 시켜야 하는데 이 과정이 공격자에 의해 악용될 수도 있다. 즉 공격자가 자신의 프로파일을 시간을 두고 서서히 변화 시키면서 detection algorithm을 학습시켜 불법적인 행위조차도 정상으로 판단하게 만들 수도 있다. 결국 Behavior-based ID에서 가장 큰 문제는 사용자/시스템 프로파일의 초기 구축 및 지속적인 관리라 할 수 있다.

Behavior-based ID를 구축하는데 가장 널리 사용되는 방법이 statistics이다 (따라서 혹자는 Anomaly Detection을 **Statistical-based Intrusion Detection**이라 부르기도 한다). 즉 사용자/시스템 프로파일을 일정 시간에 걸쳐 샘플링한 다양한 변수들의 통계를 이용하여 구축하는 것이다 (예를들어 login/logout time, resource (CPU time, disk space, file access)의 사용시간이나 양 등). 다른 방법으로 Expert System이나 Neural Network 등을 이용하여 보다 지능적이고 자동화된 침입탐지 시스템을 구축하려는 시도들이 연구되고 있으나 이들은 너무 많은 양의 계산을 요구하므로 아직까지 널리 사용되지는 않는다 (그러나 기술이 발전함에 따라 차세대 IDS에는 이러한 기술이 일반화될 것으로 보인다).

Behavior-based ID를 구축하는데 가장 널리 사용되는 것이 사용자/시스템 프로파일을 일정 시간에 걸쳐 샘플링한 다양한 변수들의 통계를 이용하는 방법이다(예를들면, login/logout time, resource (CPU time, disk space, file access)의 사용시간이나 양 등). Expert System이나 Neural Network 등을 이용하여 보다 지능적이고 자동화된 침입탐지 시스템을 구축하려는 연구가 되고 있으나 아직까지는 널리 사용되지는 않는다.

2. 침입탐지 시스템(IDS)

IDS는 크게 호스트의 불법 액세스를 탐지하는데 초점을 맞추는 Host-based IDS와 네트웍 공격을 탐지하는데 초점을 두는 Network-based IDS로 나눌 수 있다. 전자는 웹 서버나 데이터베이스 서버 등과 같은 중요한 서버의 보안에 매우 유용하며, 후자는 네트웍 기반구조를 보호하는데 중요한 역할을 한다. 각각이 장단점이 있고 상호보완적으로 사용되어야 대부분의 침입행위를 제대로 탐지 할 수 있다. 즉 네트웍에 산재한 Netowrk IDS sensor와 중요 호스트에 탑재된 Host IDS agent 및 이들을 통합관리하는 중앙의 통합관리센터로 구성되는 침입탐지시스템을 구축하는 것이 바람직하다.

Host-based IDS : Host-based IDS(이하 HIDS)는 개별 host의 OS가 제공하는 보안감사로그, 시스템 로그, 사용자 계정 등의 정보를 이용하여 호스트에 대한 공격을 탐지한다. 대부분의 HIDS는 각 호스트에 상주하는 Agent와 이들을 관리하는 Agent Manager로 구성된다. HIDS는 또한 중요한 시스템 파일이나 실행코드에 대한 무결성 검사 기능이나 시스템의 취약점들을 탐지해 주는 Vulnerability Scanner 등과 결합되어 사용된다. HIDS는 특정 시스템의 OS와 밀접히 결합되어 각종 행위들을 분석하는 만큼 보다 정교한 모니터링 & 로깅이 가능하다. 특히 불법적인 접근뿐만 아니라 합법적인 사용자에 의한 불법행위도 탐지할 수 있으므로 중요 서버의 내부자에 의한 남용을 방지하거나 사후 추적을 가능하게 하며, 또한 이런 상세 기

록들은 공격을 당했을 때 시스템을 복구하거나 사후 대책마련에도 도움을 주며, 네트웍 환경과 무관하여 VPN이나 교환망 환경 등에서도 아무런 영향을 받지 않는다. 그러나 상위계층의 로깅 정보만을 해석하므로 하위계층의 네트웍 Event들은 전혀 알 길이 없으므로 네트웍 공격탐지는 거의 불가능하고, 보호하고자 하는 모든 호스트에 설치되어야 하고, 개별 호스트의 OS나 플랫포옴에 따라 다른 시스템이 요구되므로 개발 비용이 많이 소요된다. 또한 침입탐지를 위한 각종 처리가 해당 호스트의 자원(CPU time, storage 등)를 이용하여 이루어지므로 시스템의 성능을 상당히 저하시키게 되며, IDS에 버그가 있거나 비정상적인 동작 시에는 해당 호스트 자체가 제 기능을 못할 수도 있다.

Network-based IDS : Network-based IDS(이하 NIDS)는 주로 네트웍 패킷이나 SNMP MIB, 응용 프로그램 로그 등을 분석하여 침입을 탐지한다. NIDS는 네트웍 기반의 공격을 탐지하여 네트웍 기반 구조를 보호하고자 하는 것이 목적인 만큼 대부분의 경우 HIDS에서처럼 특정 호스트의 공격은 탐지하거나 상세한 기록을 남길 수는 없다. NIDS는 또한 모든 트래픽의 실시간 분석을 통해 침입을 탐지해야 하는데, 네트웍의 고속화에 비례하여 대용량의 트래픽을 실시간으로 분석할 수 있는 NIDS를 만드는 것은 현실적으로 매우 어렵다. 또한 VPN 환경에서 트래픽이 암호화되는 경우나 교환망과 같이 네트웍이 분할되는 경우는 제 기능을 못하거나 적용범위가 제한되어 실용성이 없는 경우도 있다. 네트웍 패킷 분석에 기반한 NIDS의 또 다른 한계는 일련의 패킷(네트웍 세션)에 대한 NIDS의 해석(Fragmentation-Reassembly, TCP/IP Option Processing, Bad Packet Processing 등)이 Destination 호스트에서의 처리와 다를 수 있다는 점을 이용하는 Packet-Level 공격에 대해 취약하다는 점이다 [10]. 반면 NIDS는 대부분 NIC(Network Interface Card)를 통해 네트웍 패킷을 수집하여 Passive Analysis를 하기 때문에, 기존의 네트웍 자원에 전혀 오버헤드를 주지 않고, 설치가 용이하며, 네트웍 엑세스 지점 등에만 설치하면, 전체 네트웍에 대해서 처리 할 수 있다. 또한 호스트와는 달리 Network-based Monitor들은 능동적으로 프로토콜에 관여하는 일이 없고, 단지 전송되는 패킷을 수동적으로 수집, 분석하는 만큼 공격자가 쉽게 액세스할 수가 없고 따라서 공격에 노출될 위험도 적다.

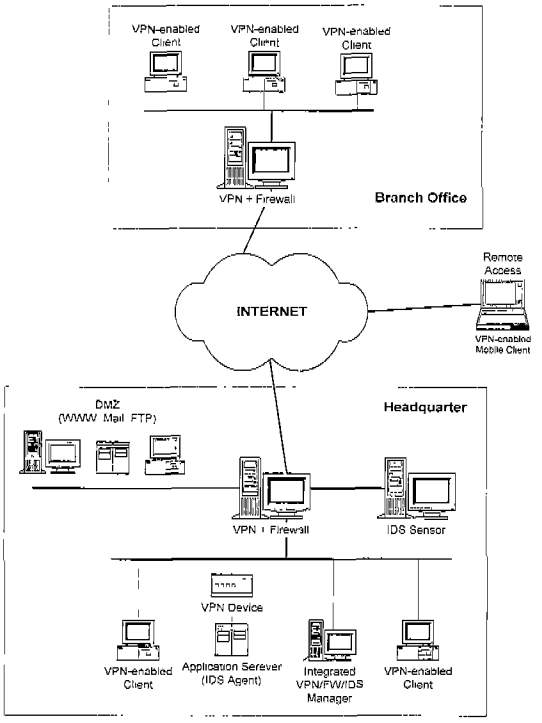

그림 5. VPN, FW과 IDS를 통합한 통합보안시스템의 구축

V. 결 론

컴퓨팅/네트워킹 환경이 다양하고 업무에 따라 보안요구조건이 다른 만큼 네트웍 보안 솔루션은 해

당 조직체의 환경이나 위협요소에 대한 철저한 분석을 통해 최적의 솔루션을 찾아야 할 것이다. 우선 조직 내에서 어떤 네트워 자원을 어떤 용도로 어떻게 사용하고 있는 지 혹은 사용하기를 원하는지를 분석해야 하고, 위험분석을 통해 그러한 네트워 자원의 사용에 있어서 보안 위협 요소가 무엇인지를 파악해야 하며, 이를 바탕으로 보안정책을 설정해야 한다. 위험분석과정에서 우선 보호해야 할 네트워 자원을 파악해서 그 자원이 침해를 당할 가능성 및 침해를 당했을 때 입게 될 유/무형의 손실 등에 대한 분석을 하고, 이에 부합하는 보안정책을 정하고, 보안정책을 가장 잘 구현할 수 있으면서도 가격 대 성능비가 우수한 제품을 선택해야 할 것이다.

일반적으로 방화벽의 주요 기능은 불특정 다수의 인터넷 사용자에 대한 접근통제이고, VPN의 주요 기능은 암호기술을 이용하여 내부 망을 인터넷으로부터 격리시키는데 있으므로 가능한 한 두 보안 제품이 상호 보완적으로 연동할 수 있을 때 보다 높은 보안을 유지할 수 있다. 또한 방화벽을 통과한 외부의 침투자나 내부 네트워에서의 각종 침입행위를 탐지하여 피해를 최소화하고 보안대책을 세우는 데는 침입탐지시스템이 중요한 역할을 할 수 있다. 따라서 이러한 주요 보안제품들이 서로 유기적으로 결합되어 중앙의 통합센타에서 일괄적으로 관리/통제할 수 있는 통합보안시스템으로 나아가는 것이 앞으로의 추세일 것이다.

최근의 아마존이나 CNN 등 대형 웹사이트에 대한 분산 서비스 거부 공격에서 보듯이 현재의 인터넷 보안은 한 사이트에서의 완벽한 보안만으로 이루어질 수는 없다. 각 조직들이 자체의 컴퓨터/네트워 보안에 일관된 대책을 세워 실행해야 할 것이고, 또한 침입행위가 탐지되었을 때는 이를 숨기기보다 이를 전담하는 국제조직들과 상호협조체계를 구축하여 공동대처하는 것이 무엇보다도 중요하다.

※ 참고문헌

[1] W.R.Cheswick and S.M.Bellovin, Firewalls and Internet Security, Addison Wesley, 1994.

[2] D.B.Chapman and E.D.Zwicky, Building Internet Firewalls, O'Reilly & Associates, 1995.

[3] Marcus Goncalves, Firewalls: A Complete Guide, McGraw-Hill, 2000.

[4] C.Scott, P.Wolfe and M.Erwin, Virtual Private Networks, O'Reilly & Associates, 1998.

[5] Casey Wilson and Peter Doak, Creating And Implementing Virtual Private Networks, Coriolis, 2000.

[6] Naganand Doraswamy and Dan Harkins, IPSEC: The New Security Standard for the Internet, Intranets, and Virtual Private Networks, Prentice Hall, 1999.

[7] http://www.ietf.org/ : IETF home page (각종 RFC 문서)

[8] Stephen Northcutt, Network Intrusion Detection: An Analyst's Handbook, New Riders, 1999.

[9] H.Debar, M.Dacier and A.Wespi, A revised taxonomy for intrusion-detection systems, IBM Research Report, Oct.1999.

[10] Thomas H. Ptacek and Timothy N. Newsham, Insertion, Evasion, and Denial of Service: Eluding Network Intrusion Detection, http://www.clark.net/~roesch/ secinfo.html l.11. http://www.cerias.purdue.edu/coast

/archive/data/categ24.html : COAST Security Archive on Intrusion Detection.

임 채 훈

1989년 서울대학교 전자공학과 학사
1992년 포항공과대학교 전자전기공학과 석사 (통신/암호학 전공)
1996년 포항공과대학교 전자전기공학과 박사 (암호학 전공)
1989년~1990년 (주)데이콤 기술본부 사원
1996년~1996년 (주)백두정보기술 암호센타장
1996년~1997년 포항공대 정보통신연구소 정보보안 연구실 위촉연구원
1997년~현재 (주)퓨쳐시스템 암호체계센타장
관심분야: 암호 알고리즘/프로토콜 설계 및 분석, 인터넷 보안, 전자상거래 등

강 명 희

1994년 광운대학교 수학과 졸업 (이학사)
1996년 광운대학교 대학원 전자계산학과 졸업 (이학석사)
1996년~1998년 백두정보기술/㈜, EP&C 팀 주임연구원
1998년~현재 (주)퓨쳐시스템, 암호체계센타 대리
관심분야: 네트웍(인터넷/인트라넷) 보안, 스마트카드, 전자상거래 등