

차세대 인터넷 구조 진화

방영철, 이상호, 홍인기, 정성택(한국산업기술대학교 컴퓨터공학과)

1. 서론

인터넷의 거대한 성장은 글로벌 세계라는 새로운 장을 연 반면, 패킷 통신 시스템에 있어 사용자 요구사항의 증가를 야기시켰다. 이는 네트워크 구조의 변화와 관리 및 확장의 용이성이 요구되는 인터넷 환경을 만들고, 인터넷의 지속적인 진화를 야기시키고 있다.

이와 같은 인터넷 진화의 한 예로, 새로운 라우팅 프로토콜, 알고리즘, 그리고 네트워크 구조들이 지속적으로 개발되고 있으며, 새로운 인터넷 계층 프로토콜과 기존 프로토콜에 대한 수정이 고안되고 있다. 특히, 라우터 분야에서는 사업자와 네트워크 운영에 관련된 조직들이 광범위하게 참여하여 새로운 표준을 조심스럽게 계획 및 관리를 하고 있다. 이는 라우터가 인터넷에서 핵심적인 역할을 하고 있을 뿐 아니라, 인터넷에 설치된 많은 라우터들은 호스트의 수보다 적다는 사실과 라우터에 관련된 표준이 호스트 관련 표준보다 빠른 속도로 진화할 것이라는 기대 때문이다.

인터넷 고안자와 초창기 사용자들은 전 세계의 컴퓨터들을 한데 묶은 인터넷 기반 위에 운

영될 새로운 응용을 창출하고자 하는 공통된 목적을 가지고 있었다. 즉, 설계자, 사용자 혹은 운영자들은 일관적인 비전과 같은 목적을 공유하였다. 이는 더 이상의 설명이 필요 없는 인터넷 성공의 원동력이었다.

인터넷 구조는 자체 기술 데이터그램(self-describing datagram) 패킷, 종단간 인수(end-to-end arguments), 기술의 다양성, 그리고 글로벌 어드레싱(global addressing) 등을 포함한 여러 원리에 근거를 두고 개발 및 진화하였다. 그러나, 인터넷의 주 사용 계층이 연구중심의 엔지니어에서 일반 사용자로 이동하면서 새로운 설계원리를 제안하도록 하는 요구사항들이 출현하게 되었다.

인터넷은 하나의 커뮤니티로서의 복잡성과 구성요소들의 고장 등과 같은 여러 난제들에도 불구하고 안정성, 확장성, 그리고 관리의 효율성 등을 제공하는 것을 설계원리로 하여 발전하여 왔다. 그러나, 오늘날 인터넷의 주류가 엔지니어에서 일반 사용자로 옮겨가면서 엔지니어에 의해 설계되고 개발된 인터넷의 진화 방향을 예측할 수 없게 되었다. 즉, 사용자 집단들간의 이익 창출을 위한 사이버 공간(tussle)^[1]으로 발전된 오

늘날의 인터넷은 이해관계가 다른 여러 사용자 집단들에 의해서 복잡한 구조로 진화되고 있다.

인터넷에 관련된 연구와 개발이 직면한 도전은 이익을 창출하기 위한 사용자들의 요구사항을 인식하고 반영하는데 있다. 즉, 인터넷의 기술적인 구조는 기존의 목적인 안정성, 확장성, 관리의 효율성 및 진화력을 만족시켜야 할 뿐 아니라, 오늘날 사용자 집단들의 요구사항을 반드시 수용해야만 한다. 이를 위해, 새로운 인터넷 구조가 요구되고 있으며, 기술적으로 어려운 숙제로 남아있다.

본 고에서는 네트워크 구조 진화의 필요성^[1]에 대하여 논하고 현재 진행중인 세가지 대표적인 프로젝트인 Role-Based Architecture^[2], A New Internet Routing Architecture^[3], 그리고 Reorganizing the Addressing Architecture^[4]에 대하여 기술하고자 한다.

II. 네트워크 구조 진화의 필요성

네트워크 구조는 통신 메커니즘이나 프로토콜을 설계할 때에 일반적으로 사용하는 용어이다. 이것의 목적은 사용자의 요구를 만족시키기 위해서 필요한 결정들을 일관되게 할 수 있도록 하는 것이다. 그런데, 기술의 발전과 더불어 사용자의 요구가 새롭게 많이 등장하였다. 기존의 인터넷 구조에서는 이들 요구를 만족시키기 위해 많은 기능을 추가하는 노력을 하였다. 이러한 노력은 기존의 네트워크 구조를 더욱 복잡하게 하여 유지/보수 측면에서 많은 문제점을 야기시켰을 뿐만 아니라, 기능적인 측면에서도 특정 기능의 손실을 야기시켰고, 인터넷 구조 측면에서도 문제를 야기시켰다.

기존 네트워크(또는 인터넷) 구조의 문제점은

아래와 같다. 첫째, 외부로부터 네트워크를 보호하기 위해서 개발된 방화벽 (firewalls)은 인터넷의 종단간 기능을 손상시켰다. 둘째, 보안을 위해서 개발한 IPSEC는 전송계층 (transport-layer) 헤더를 암호화함으로써, 네트워크 공급자가 포터 번호와 상위 계층의 프로토콜 헤더를 볼 수 없게 하여 기존의 인터넷 구조의 계층 모델에 모순되게 하였다. 셋째, 부족한 IP 주소 공간을 보완하기 위해서 개발된 NAT(network address translation)은 기존 구조의 peer-to-peer 모델과 모순되게 하였다^[5]. 넷째, 사용자의 요구를 만족시키기 위해서 단순히 확장한 기능들은 상호간에도 불협화음을 조장하였다. 예를 들면, NAT 장치는 전송계층 (transport-layer) 헤더를 암호화하는 IPSEC와 같이 사용될 수가 없었다. 이외에도 기존 네트워크는 사용자 요구 사항을 만족시키기 위해서 첨가한 기능에 의해서 많은 문제들을 야기시켰다.

이러한 문제를 해결하기 위해서 현재의 요구 사항뿐만 아니라 미래의 요구 사항까지도 고려한 새로운 네트워크 구조가 필요하게 되었다. 새로운 네트워크 구조가 고려해야 할 사항으로는 (1) 이동성, (2) Policy-driven auto-configuration, (3) 이동성 등으로 인한 짧은 시간내의 전송 매체(backbone link 또는 mobile device) 변경 등이 있다.

III. 역할 기반 구조 (Role-Based Architecture)

계층기반으로 구성된 네트워크 구조에 많은 문제점이 제기되면서 네트워크 프로토콜의 설계 및 구현에 있어 비계층구조가 소개되고 있다. 그 목적은 통신시 발생하는 문제점을 보다

유연하게 제어하는 것이다. 본 절에서는 역할기반구조(role-based architecture)라 불리는 비계층 프로토콜에 대하여 기술한다.

현재 운용중인 패킷 기반 네트워크 구조에서 통신기능들은 프로토콜 계층⁶⁾으로 구성하고, 패킷 전송을 제어하는 메타데이터는 각 프로토콜 계층마다 하나씩 존재하는 프로토콜 헤더에 저장된다.

프로토콜의 계층적 구조는 네트워크의 원리를 구성하는데 적합하고, 특히, 현재 사용되는 인터넷 구조에 비해 원래 인터넷구조의 엄격한 종단간 모델에 더욱더 적합하다. 그러나, 오늘날 인터넷의 사용이 일반인에게 빠르게 전파되면서, 사용자의 높아지는 요구사항을 만족시키기 위해 개발된 방화벽, NAT 박스, 플록시, 그리고 캐쉬 메모리등과 같은 미들 박스(middle boxes)의 빠른 확산은 예기치 못한 많은 문제점들을 발생시키고 있다. 또한, 멀티캐스트(multicast), 오버레이 라우팅(overlay routing), 그리고 터널링(tunneling)과 같은 기능들은 네트워크 구조를 더욱 복잡하게 만들고 있다.

앞에서 논한 것과 같이 오늘날 네트워크 구조는 사용자의 요구사항들을 만족시키기 위한 미들박스와 많은 기능들이 첨가됨으로써 복잡하게 변화되었다. 즉, 네트워크 구조 설계자들은 이미 작동중인 각 계층에 존재하는 프로토콜들과 오랫동안 지속되어온 계층 사이의 인터페이스를 수정하기 보다는 존재하는 계층 사이에 새로운 기능을 첨가시켰다. 계층 2.5의 MPLS, 계층 3.5의 IPsec, 그리고 계층 4.5의 전송계층 보안(Transport-Layer Security)등이 이에 해당한다. 이로 인해, 네트워크 구조는 새로운 서비스를 기존의 계층구조에 적용하는데 더 이상 적합하지 않게 되었다.

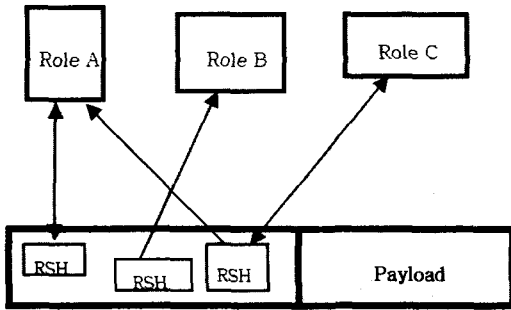
뿐만 아니라, 프로토콜 설계자들은 이익창출을 위한 사이버 공간⁷⁾상에서 다른 프로토콜에 의해 고려된 가정(assumptions)들을 위반하는 근시안적인 설계를 계속적으로 제안하고 있다. 이로 인해, 네트워크 프로토콜의 계층화는 네트워크 소프트웨어 모듈화를 위한 추상적 개념에 충분히 유연하지 못하게 되었다. 즉, 현재의 프로토콜 계층화는 기존의 프로토콜 표준들과 함께 유연성에 관련된 문제를 고려하여야 한다.

결론적으로, 통신의 추상적인 개념들을 기반으로 하여 기존 방법보다 더욱 명확하고, 일반적이면서, 확장성과 유연한 관계를 제공하는 진화된 네트워크 구조가 절실하게 필요하게 되었다.

최근에 연구가 되고 있는 비계층 네트워크 구조인 역할기반구조(Role-Based Architecture)에서 역할(roles)은 잘 정의되고 잘 알려진 블록을 구성한다. 프로토콜간 상호 연동을 위하여, 역할기반구조를 사용하는 네트워크에서는 상대적으로 적은 수(수천)의 역할만을 정의하고 표준화하는 것을 요구한다. 하지만, 특수목적, 실험적, 혹은 국지적으로 정의된 역할을 고려하면 역할의 수는 훨씬 많을 수 있다.

역할기반구조의 일반적인 아이디어는 기능성이 명시된 신호(signaling)들을 설계하자는 것이다. 설계된 신호의 부족은 미들박스가 현재의 계층적 구조에 적합하지 않은 주된 이유 중 하나이다. 예를 들어, 패킷이 사이트를 보호하는 방화벽을 통과할 때, 종단 시스템에 신호를 보낼 정의된 방법이 없거나, 종단 시스템이 웹 캐쉬에 의해서 리디렉트(redirect)된 요청(request)를 원치 않는다는 것을 신호할 방법이 없게 된다.

역할기반구조는 네트워크를 구성하는 요소들을 명시적으로 확인하고, 어드레싱(addressing)하여 통신을 할 수 있게 한다. 뿐만 아니라, 이



〈그림 1〉 역할(Roles)과 Role-Specific Headers

것은 IP, TCP, 그리고 HTTP등과 같은 방대한 크기의 프로토콜들을 구체적인 임무만을 수행하는 packet forwarding 및 fragmentation 등과 같은 다소 작은 크기의 단위로 재모듈화 (re-modularization)할 수 있게 한다. 이를 통해, 이러한 작은 단위 모듈도 역할기반구조에서는 구체적인 역할을 수행할 수 있게 된다.

앞에서 언급한 비계층구조로서의 역할기반구조는 역할이라 불리는 프로토콜 단위로 구성되어 있다. 역할은 패킷 전송과 처리에 관련된 구체적인 기능을 수행하는 블록(block)으로 생각할 수 있으며, 개체(entities)의 추상적 개념을 형식화하는 것을 가능하게 한다.

역할은 특별한 특성들을 갖는다고 단순히 말할 수 있으며, actor라 불리는 코드에 의해 노드에서 임무를 시작한다. 다양한 목적을 위해 역할과 actor사이의 구별은 중요하지 않으며, 역할은 논리적으로 다수의 노드에서 수행될 수 있다. 따라서, 각 actor가 특정한 노드에서 수행되는 동안, 역할은 추상적 개념으로 여러 노드에 분산될 수 있다.

역할 데이터(role data)라 불리는 메타데이터(meta data)는 role-specific headers (RSH)로 나뉘어진다. 그림1은 힙(heap)에 존재하는 세계

의 RSH와 이들 RSH를 읽고 쓰는 세계의 역할을 포함하는 패킷을 보여주고 있다.

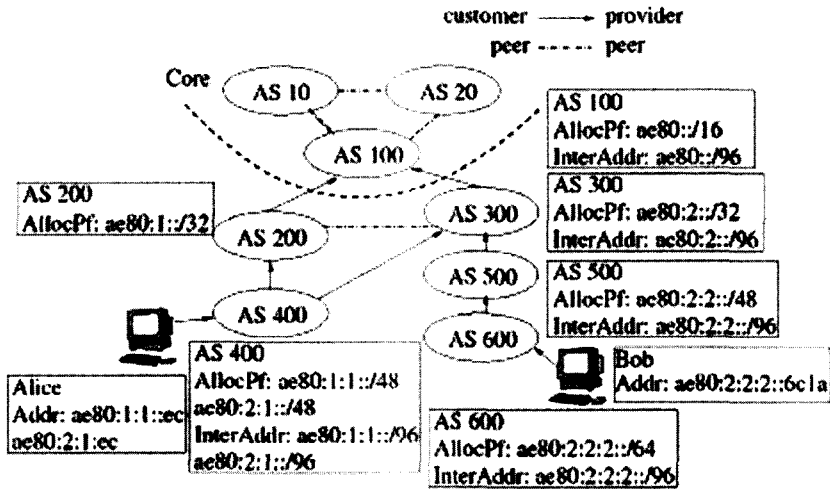
종단 시스템은 자신이 보낸 패킷이 어떤 특정한 역할을 하는 노드와 만나는 지를 알 필요가 없으며, 패킷이 전송되는 경로상의 어떤 노드도 통과하는 패킷에 RSH를 첨가시킬 수 있다.

IV. New Internet Routing Architecture (NIRA)

현재 사용되고 있는 인터넷에서 사용자들은 ISP는 선택할 수 있으나, 패킷이 네트워크에 들어오면 그 패킷이 어떤 라우팅 경로를 통해 전송되는 지는 선택할 수 없다. 여기서, 사용자는 인터넷을 사용하고 있는 컴퓨터에서 작동하는 소프트웨어를 뜻한다.

새로운 인터넷 라우팅 아키텍처인 NIRA가 추구하는 목표는 패킷의 경로를 사용자가 선택할 수 있게 하는 것이다. 이는 사용자들의 선택에 의해 경쟁을 유도함으로써 경제의 논리를 인터넷 사업자에게 적용시킬 수 있으며, 새로운 서비스와 기술 혁신을 가져오게 할 것이다.

NIRA에서 해결해야 할 문제 점은 BGP(Border Gateway Protocol)가 디폴트 라우터를 선택하기 때문에 사용자는 라우팅에 대한 선택권이 전혀 없다는 것이다. 뿐만 아니라, BGP는 느리게 동작하고, 간단한 메트릭에 의존해서 경로를 선택하기 때문에 라우팅 경로를 찾지 못했을 경우에 더욱 느리게 반응한다는 것이다. 그러나, NIRA에서와 같이 라우터 경로 선택이 지역내의 ISP들에 의해 수행되게 함으로써 End-to-End QoS와 같은 새로운 서비스를 개발할 수 있게 하고, 이와 같은 서비스를 통해 라우팅 경로가 더 빠르고 안정되게 동작할 수



〈그림 2〉 NIRA의 네트워크 모델과 어드레싱 방법

있게 한다.

사용자의 라우팅 경로 선택을 효과적으로 지원하기 위해 NIRA는 여러 요구 사항들을 만족시켜야 한다. 이들 요구 사항들은 아래와 같다.

첫째, NIRA는 Scalability 성질을 가져야 한다. 이는 빠른 인터넷의 성장과 수많은 기기들이 인터넷에 접속되기 때문에 하나의 라우터가 모든 것을 처리할 수 없기 때문이다. Scalability를 위해서는 Name-to-Route Resolution 프로토콜과 Topology Information Propagation 프로토콜이 필요하고, 라우팅 경로를 찾는 프로세서는 두 개로 나누어져 실행되어야 한다.

둘째, NIRA는 Robustness 성질을 가져야 한다. 즉, 라우터가 존재하지 않을 때에는 사용자가 직접 다른 경로의 라우터를 선택할 수 있게 하며, 또한 하나의 라우터가 여러 가지 이유로 동작하지 않더라도 전체적인 라우터에 영향을 미치지 않게 해야 한다. 이러한 Robustness를 위해서는 라우팅 경로를 발견하지 못했을 경우

를 미리 예측할 수 있어야 하고, 빠른 피드백을 제공해야 한다.

셋째, NIRA는 Efficiency성질을 가져야 한다. 즉, 도메인 레벨의 라우팅 선택을 위한 여러 가지 오버헤드를 최소화해야 한다. 이를 위해서는 계층적인 어드레싱이 필요하다.

넷째, NIRA는 Heterogeneous User Choices를 지원해야 한다. 즉, 한 도메인에 있는 각각의 사용자가 다른 종류의 ISP를 선택할 수 있게 해야 한다. 이를 위해서는 Scalable 라우팅이 필요하다.

NIRA와 관련된 현재까지 진행된 연구에는 NIMROD에서 제안한 Hierarchical Map Distribution, IDPR(Inter-Domain Policy Routing), SIDRA(Scalable Inter-Domain Routing Architecture), SIPP(Simple IP Plus), TRIAD(Translated Relaying Internet Architecture integrating Active Directories), 그리고 Feedback based Routing 등 있다. NIRA와 위의 선행 연구들과의 차이점은 NIRA에는

Scalable route discovery, Efficient Route Representation, Heterogeneous User Choices라는 부분이 있다는 것이다.

그림2에서, 코어(core)는 네트워크에 고밀도로 접속되는 영역으로서, AS 10, AS 20, 그리고 AS 100이 이에 해당된다. 이러한 코어에서는 패킷이 더 이상 올라가지 못하고 멈추게 되어 코어의 외부 영역에 접속되는 기기의 수가 적게 된다.

그림에서 AS 400-200-100-300-500-600 과 AS 400-200-300-500-600은 “Valley-free”라 하는데, 이것은 도메인 레벨의 라우팅으로서 사용자가 보낸 패킷이 사용자의 ISP 체인을 따라 올라가다가 리서버부분에서 ISP 체인을 따라 내려가는 현상을 의미한다.

“Valley-free” 라우팅은 두 영역으로 나뉘어진다. 즉, 소스 어드레스를 모으는 ISP영역과 목적지 어드레스를 모으는 ISP 체인 영역으로 나누어 지게 된다. 이 라우팅은 TIPP(Topology Information Propagation Protocol)을 이용해 ‘up-hill’ 라우터를 구하고, NRRS(Name-to-Route-Resolution Service)를 이용해 목적지 어드레스를 알려 준다.

TIPP는 프로토콜과 비슷한 개념으로서 라우터로 하여금 토폴로지 정보를 선택적으로 사용해서 네트워크를 진행하게 만든다. 또한, TIPP는 라우팅 정보의 업데이트가 각각의 사용자에 의해 선택될 수 있게 함으로써, 전체 네트워크의 라우팅에 영향이 미치지 않게 한다. NRRS는 사용자로 하여금 다른 사용자의 라우팅 세크먼트를 찾을 수 있게 해 준다.

NIRA는 라우팅을 만들고 표현하는데 드는 오버헤드를 줄이기 위해서 계층적인 provider-rooted 어드레싱 스킴을 사용한다. 이 어드레싱

은 IPv6와 같은 방식으로 표현되며, 각각의 ISP마다 고유한 접두사 부분을 제공하여 각각의 사용자는 그 ISP가 사용자에게 다른 어드레스를 할당하게 한다.

이러한 NIRA의 최종 목적은 사용자에게 도메인 레벨의 라우팅을 제공하고, 사용자로 하여금 자기 자신의 라우팅을 결정할 수 있도록 함으로써, 더 나은 네트워크 환경을 구축하는데 있다.

V. FARA(Forwarding directive, Association, and Rendezvous Architecutre)

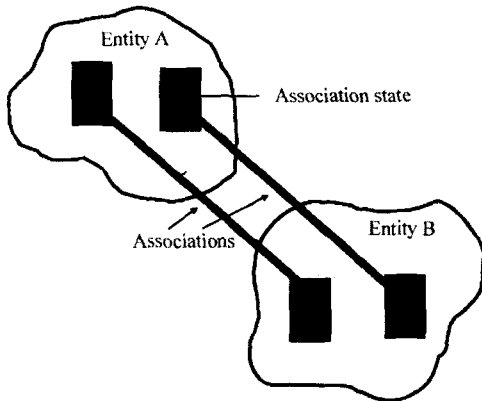
FARA는 추상적인 네트워크 구조 모델로서, 여러 개념적인 요소를 정의하였으나, 많은 세부적인 기술적 결정 사항들은 개발자가 하도록 하였다. FARA의 목표는 (1)network-layer forwarding 메커니즘으로부터 end-system 식별을 분리하며, (2)새로운 전역 namespace를 불필요하게 하고, (3) 보증 단계를 통한 보안을 제공하는 것이다. 이들 목표를 달성하기 위해서 FARA에서 정의한 기본 요소들은 Entity, Association, 통신 기반 (communication substrate), Forwarding Directive, 그리고 Rendezvous이다. 이들에 관해서 각각 논하도록 하겠다.

* Entity

Entity는 네트워크 통신의 끝단에 존재하는 응용을 일반화한 것이다. 즉, entity는 프로세스, 쓰레드, 또는 완전한 컴퓨터, 휴대폰 등이 될 수 있다.

* Association

Association은 각 entity들이 통신을 하기 위해서 사용하는 논리적인 통신 링크(communi-



〈그림 3〉 Associations

cation links)로서, 그림3에서와 같이 링크된 entity들의 내부에 존재하는 통신 상태 (communication state)와 entity들 사이에 전달되는 패킷의 흐름으로 정의된다. 수신 entity가 수신한 메시지를 해당 association으로 전달하기 위해, FARA에서의 각 패킷은 association ID(AId)를 가지고 있다.

* 통신 기반 (Communication Substrate)

FARA에서는 패킷 전달 메커니즘으로서 특정 메커니즘을 제안하지는 않지만 connectionless 패킷 전달 메커니즘을 가정한다. 즉, hop/hot delivery, label swapping등이 모두 가능하다. FARA에서는 오직 connectionless 패킷 전달 메커니즘만을 가정하였기 때문에 entity들이 패킷 전달의 신뢰성을 보장해야 한다.

* Forwarding Directive

FARA에서의 패킷 전달은 노드도 아니고 목적지 호스크의 프로토콜 스택(stack)도 아닌 association의 반대편 entity이다. 따라서, entity들이 패킷을 전송할 때에 목적지 entity에 관한 정보를 패킷의 헤드 필드에 기록하게 되는데,

이를 Forwarding Directive(FD)라 부른다.

* Rendezvous

FARA에서는 전역 namespace를 사용하지 않기 때문에 패킷 전송을 시작할 때에는 목적 entity내의 AId등을 알 수 없다. 따라서, 최초 전송 패킷에는 AId를 포함하지 않고, AId를 할당하고, 이후 패킷 전송을 원활히 할 수 있도록 하는 정보가 들어 가며, 이 정보를 rendezvous information이라 한다. 그리고, 이 메커니즘을 rendezvous라 한다.

VI. 결론

본 논고에서는 많은 새로운 요구사항의 등장으로 인해 필요성이 대두되고 있는 새로운 네트워크 구조의 개발에 관해 논하였다.

본 논고에서는 기존 인터넷 구조가 새로운 요구사항들을 만족시키기 위해서 추가한 많은 기능들이 효과적으로 동작하지 않음에 관해 논하고, 이와 같은 이유로 새로운 네트워크 구조가 필요하게 되었음을 논하였다. 그리고, 새로운 네트워크 구조에 관한 대표적인 프로젝트인 역할 기반 구조 (Role-Based Architecture), New Internet Routing Architecture (NIRA) 그리고 FARA(Forwarding directive, Association, and Rendezvous Architecutre)에 관해 논하였다.

참고문헌

- [1] Clark, D., Wroclawski, J., Sollins, K., Braden, R., "Tussle in Cyberspace: Defining Tomorrow's Internet", ACM SIGCOMM 2002, August 2002.
- [2] Braden, R., Faber, T., Handley, M., "From Protocol Stack to Protocol Heap - Role-Based Architecture",

- HotNets-I, Princeton, NJ, October 2002.
- [3] Xiaowai Yang, "NIRA: A New Internet Routing Architecture", ACM SIGCOMM FDNA 2003 Workshop, Karlsruhe, August 2003.
- [4] Clark, D., Braden, R., Falk, A., and Pingali, V., "FARA: Reorganizing the Addressing Architecture", ACM SIGCOMM 2003 FDNA Workshop, Karlsruhe, August 2003.
- [5] Braden, R., Clark, D., Shenker, S., and Wroclawski, J., "Developing a Next-Generation Internet Architecture", <http://www.isi.edu/newarch/DOCUMENTS/WhitePaper.pdf>, July 15, 2000.
- [6] E. Meyer. ARPA Network Protocol Notes. RFC 46, Network Working Group, April 1970.

저자소개



박병일

1994년 University of Oklahoma 전자공학(박사)
 1997년 University of Oklahoma 전자공학(석사)
 2000년 University of Oklahoma 전자공학(박사)
 2001년~2002년 한국산학기술연구원 과장 겸 포스트
 도플링
 2002년~현재 한국산업기술대학교 컴퓨터공학부
 조교수

저자소개



이상훈

1991년 경북대학교 전자계산학과(학사)
 1992년~1993년 한국과학기술원 전자공학(석사)
 2001년 한국과학기술원 전자공학(박사)
 2001년~2003년 LG전자기술원 선임 연구원
 2003년~현재 한국산업기술대학교 컴퓨터
 공학과 전임강사



홍인기

1991년 KAIST 전기 및 전자공학(박사)
 1992년~1999년 UC Irvine, Radiological
 Science 박사후 연구원
 1999년 광주과학기술원 정보 및 통신
 공학과 연구교수
 2003년~현재 한국산업기술대학교 컴퓨터
 공학과 전임강사



정성택

2002년 KAIST 전기 및 전자공학(박사)
 1995년~2000년 메디슨 MRI연구소 선임연구원
 2000년~2004년 메디너스 연구소장
 2004년~현재 한국산업기술대학교 컴퓨터
 공학과 전임강사