

공개 소프트웨어 기반 IP 네트워크 시스템 실습 플랫폼 및 실습 시나리오

김태준
공주대학교 정보통신공학부

A Study on Open Software based IP Network System Practice Platform and Practicing Scenario

Tae-Joon Kim

Division of Information and Communication, Kongju National University

요약 4차 산업혁명 시대의 핵심 인프라로 자리 매김하는 IP(Internet Protocol) 네트워크 시스템을 실제 구축, 시험 및 운영하면서 공부한 내용을 확인하고 이해하는 것은 네트워크 입문자들에게 매우 중요하다. IP 네트워크 시스템은 인터넷 계층 이하의 기능을 담당하는 IP 네트워크 자체와 DNS, DHCP, email 및 WWW로 기본적인 네트워크 서비스를 제공하는 기본 네트워크 시스템, 그리고 VoIP와 같은 응용 서비스 시스템으로 구성된다. 본 논문에서는 공개 소프트웨어 기반의 최소한의 네트워크 형상으로 IP 네트워크 시스템의 제반 기능을 실습해볼 수 있는 플랫폼을 제안하고, 실습 시나리오와 결과를 제시한다. 그리고 ns(network simulator)에 IP 주소 할당과 라우팅 테이블 관리 기능을 추가하여 테이블 기반 데이터그램 라우팅 기법을 사전 학습할 수 있도록 하였다. 본 연구에서 제안된 실습 플랫폼으로 별도의 장비와 비용 부담 없이 개인용 컴퓨터에서 간편하게 실습을 수행할 수 있어 네트워크 입문자에게 유용하리라 기대된다.

Abstract It is very important for beginners in networking to build, test, and operate an IP network system, which is the key infrastructure of the fourth industrial revolution. An IP network system consists of the IP network itself, covering both the internet and underlying layers, plus the basic network system supporting fundamental network services, with a DNS, the DHCP, email, WWW, and application service systems, such as VoIP. This paper suggests an open software-based practice platform with a minimal network configuration on which beginners can practice on nearly all of the IP network system, and it presents a practice scenario and results. In addition, it adds both IP address allocation and routing table manipulation functions to the network simulator software, which will allow beginners to learn table-based datagram routing schemes in advance. This platform may be useful for networking beginners in order to practice on an IP network system on their own personal computers without additional equipment and costs.

Keywords : Internet, Practice platform, Practice scenario, Open software, Network simulator

1. 서론

인공지능, 사물 인터넷, 빅 데이터, 모바일 등의 첨단
정보통신 기술이 경제·사회에 융합되어 혁신적인 변화를

불러오는 4차 산업혁명이 시작되고 있다. 이러한 산업혁
명의 핵심은 초 연결성으로 이는 IP 네트워크 시스템에
의해 실현되고 있다. 따라서 4차 산업혁명 시대를 이끌어
갈 첨단 기술에 능동적으로 대응하기 위해서는 IP 네트

*Corresponding Author : Tae-Joon Kim(Kongju National Univ.)

email: tjkim@kongju.ac.kr

Received July 6, 2020

Accepted October 5, 2020

Revised August 4, 2020

Published October 31, 2020

워크 시스템을 이해하는 것이 매우 중요하며, 특히 실제 구축, 시험 및 운영의 실습을 통해 시스템의 동작을 이해하고 학습하는 것은 입문자들에게 필수적인 과정이라 할 수 있겠다.

많은 시장 수요로 여러 교육용 실습장치가 개발되어 왔다[1-8]. [1]은 임베디드 장비로 실습환경을 구축해야 하므로 비용 부담이 크고, [2]는 IP 네트워크 실습을 위해 별도로 개발된 에이전트 소프트웨어를 사용해야 하므로 제약이 많다. UML(User Mode Linux) 기반 가상환경에서 IP 네트워크를 실습할 수 있는 환경이 제안되었고[3], 자동접수기능과 작업지시 지원기능을 개발하여 강사 맞춤형 실습 지원 환경을 제시하였다[4]. 실습 파트너를 대신하는 동료 대리자 소프트웨어로 다수의 학생이 상호 협력하면서 네트워크를 구축하는 실습 환경을 효과적으로 제공하였고[5], 실습하는 학생을 원격 지원할 수 있는 강사 지원 기능이 개발되었다[6]. 그러나 [1-6]은 모두 IP 네트워크 자체만 대상으로 하고 있어 네트워크 서비스 시스템의 실습에는 적용할 수 없다. [7]에서 ns[9]에 IP 주소할당 기능과 라우팅 테이블 관리 기능을 추가하였고, [8]에서 인터넷 시스템 실습 플랫폼을 설계하였다. 최근에는 엄청난 자료를 축적하고 있는 검색 사이트에서 IP 네트워크 시스템의 실습에 관한 많은 단편 정보를 얻을 수 있으므로 실습 플랫폼만 제공되면 입문자들도 독자적으로 네트워크를 실습해볼 수 있는 좋은 여건이 조성되고 있다. 따라서 실습 플랫폼의 중요성이 점차 강조되고 있다.

IP 네트워크 시스템은 인터넷 계층 이하의 기능을 담당하는 IP 네트워크 자체와 DNS, DHCP, email 및 WWW로 기본적인 네트워크 서비스를 제공하는 기본 네트워크 시스템 그리고 VoIP와 같은 응용 서비스 시스템으로 구성된다. 본 연구에서는 [7]과 [8]의 결과를 확장하여 이러한 IP 네트워크 시스템 전체를 실습할 수 있는 실습 플랫폼을 제안하고, 테이블 기반 라우팅 기법을 사전 학습할 수 있는 시뮬레이터를 개발하며, 제안된 플랫폼에서의 실제 실습을 바탕으로 실습 시나리오와 실습 결과를 제시한다.

2. 본론

2.1 IP 네트워크 시스템 실습 플랫폼

먼저 실습 플랫폼이 커버할 실습범위에 대해 살펴본다. 인터넷 시스템은 Fig. 1과 같이 링크, 인터넷, 전송

및 응용의 4 계층으로 모델링되며, 그림에 도시된 프로토콜들이 모두 플랫폼에서 취급될 대상이다. 구체적으로 살펴보면 링크계층에서는 널리 사용되는 이더넷 프로토콜, 인터넷 계층에서는 IP와 이의 동작을 도와주는 ARP(Address Resolution Protocol), ICMP(Internet Control Message Protocol), RIP (Routing Information Protocol) 및 OSPF(Open Shortest Path First), 그리고 전송계층에서는 TCP(Transport Control Protocol)와 UDP(User Datagram Protocol)가 실습대상이다. 응용계층에서는 기본적인 네트워크 서비스를 제공하는 DNS(Domain Name System), DHCP(Dynamic Host Configuration Protocol), SMTP(Simple Mail Transfer Protocol) 및 HTTP(Hyper Text Transfer Protocol)와 최근에 널리 보급되고 있는 인터넷 전화를 위한 SIP(Session Initiation Protocol)/SDP(Session Description Protocol)가 실습대상에 포함된다.

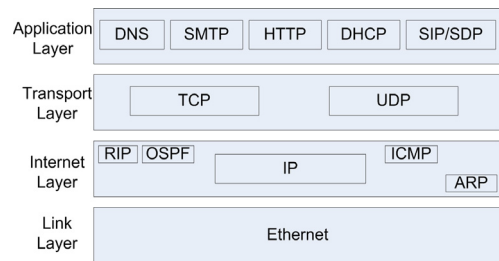


Fig. 1. Internet Model and Practice Subjects

실습 플랫폼을 설계함에 있어 우선적으로 고려한 사항은 별도의 장비나 추가적 비용 없이 학생들이 소지한 노트북 또는 개인용 컴퓨터에 플랫폼을 탑재하여 실습을 할 수 있도록 해야 한다는 것이다. 이를 위해 앞에서 살펴본 대상 프로토콜들을 모두 실습할 수 있으면서 최소한의 플랫폼 네트워크 형상을 갖도록 해야 한다.

IP 네트워크 시스템은 서비스를 이용하는 가입자 부분과 서비스를 제공하는 서버 부분으로 나눌 수 있다. 가입자 부분은 1인 가구부터 대규모 기업까지 다양하고, 서버 역시 소규모 서버 장치로부터 대규모 서버 집단까지 광범위하다. 이러한 가입자와 서버 부분을 실습할 수 있는 융통성 있으면서도 간단한 실습 도메인 체계의 설계가 매우 중요하다. 또한 공중 인터넷과 동일한 환경에서 실습할 수 있어야 하나 기존 인터넷의 루트 네임서버와 도메인 체계를 함부로 이용할 수 없다. 따라서 공중 인터넷 도메인과 동일한 도메인 체계를 플랫폼이 갖도록 해야 한다. 본 연구에서는 편의상 서비스를 제공하는 서버

사이트를 kongju로 하였고, 가입자 사이트의 ISP(Internet Service Provider) 사업자를 kt로 하였다. 실습 플랫폼에 탑재할 도메인 체계는 광범위한 서버를 대표할 수 있는 s.kongju.ac.kr과 다양한 가입자를 대표할 수 있는 a.kt.co.kr을 수용하는 간단한 체계로 설계하였으며 구체적으로 Fig. 2와 같다.

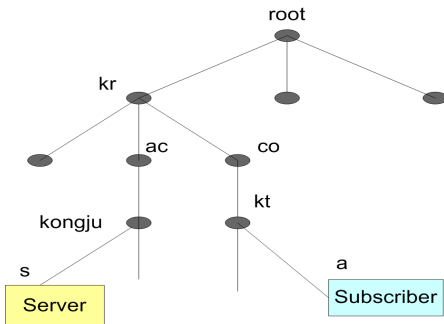


Fig. 2. Platform Domain Name System

IP 네트워크 시스템은 가입자 호스트, 서버 시스템 및 이들을 상호 접속해주는 라우터의 세 부분으로 나눌 수 있다. 호스트는 링크, 인터넷 및 전송 계층 프로토콜들을 내장하고 있는 운영체제와 응용 계층의 클라이언트 소프트웨어를 탑재하고 있고, 서버 시스템 역시 마찬가지로의 운영체제와 응용 계층의 서버 소프트웨어를 탑재하고 있으며, 라우터는 인터넷 계층 이하 프로토콜을 탑재하고 있다. 따라서 Fig. 1에 도시된 실습 대상들을 실질적으로 구분해보면 인터넷 계층 이하의 기능을 담당하는 IP 네트워크 자체와 IP 프로토콜 환경변수의 공급 및 도메인 네임 해석으로 URL(Uniform Resource Locator) 기반 네트워크 시스템이 동작하도록 해줄 뿐만 아니라 메일과 웹 등의 기본적인 서비스 기능을 제공하는 기본 네트워크 시스템, 그리고 인터넷 전화 같은 정보통신 서비스를 제공하는 응용 서비스 시스템으로 나눌 수 있다

IP 네트워크 부분을 담당하는 프로토콜들은 모두 운영체제에 내장되므로 이의 실습을 위해서 별도의 작업이 필요 없고 프로토콜 분석기로 통신 프로토콜을 분석하는 과정만 요구된다. 기본 네트워크와 응용 서비스 시스템 실습을 위해서 서비스를 제공하는 서버를 탑재해야 한다. Fig. 2에 정의된 플랫폼의 각 도메인에 탑재할 서버는 Fig. 3과 같다.

Fig. 2의 도메인 체계를 수용하기 위해서는 도메인 네임을 해석하는 네임 서버를 탑재해야 하는데, 본 연구에

서는 혼란을 방지하기 위해 네임서버가 커버하는 서비스 범위의 존(zone)을 도메인과 동일하게 하므로 Fig. 3과 같이 루트 도메인부터 모든 도메인 각각에 별도의 네임 서버, 즉 NS를 탑재한다. 가입자를 수용하는 kt 도메인에는 가입자들에게 IP 프로토콜 환경 변수를 제공해주는 DS, 메일 서비스를 제공해주는 MX 및 자체 웹서비스를 위한 WS를 탑재한다. 그리고 서비스를 제공하는 kongju 도메인에는 WS와 MX 및 인터넷 전화 서버인 VS를 탑재한다.

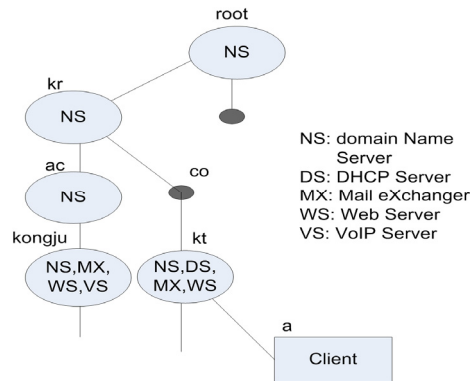


Fig. 3. Server Deployment in Platform

Fig. 3과 같이 배치된 서버들은 가상 운영체제 환경을 이용하여 실현한다. 본 연구에서는 윈도 호스트에 가상 운영체제를 지원하는 shareware인 VMware를 설치하고 여기에 리눅스 가상머신을 탑재하여 구축한다. 본 논문에서는 없어도 실습에는 무관한 ac 도메인을 생략하여 4개의 리눅스 가상머신과 클라이언트를 수용할 윈도 호스트로 플랫폼을 구성한다. 설계된 플랫폼의 네트워크 형상은 Fig. 4와 같다. 구체적으로 윈도 호스트의 공중 인터넷 회선을 공유하여 실습에 필요한 오픈 소프트웨어를 다운 받을 수 있는 NAT(Network Address Translation) 유형의 192. 168.137.0과 각각 LAN1~3에 할당된 10.0.10.0, 10.0.20.0 및 10.0.30.0의 4 네트워크의 토폴로지를 갖는다. 각 node에 할당된 IP 주소는 편의상 4개의 십진수 중 호스트 식별자에 해당하는 끝 십진수만 표시하였다.

플랫폼에 사용되는 리눅스는 범용 운영체제로 실습 플랫폼의 용도로 사용하기 위해서는 필요한 공개 소프트웨어를 다운받아 설치하고 설치한 소프트웨어의 형상설정 파일의 수정과 restart를 통해 네트워크 장치로 동작시켜야 한다. 이를 위해 pnet.init라는 셸 스크립트를 사용한

다. 이를 수행시키면 Fig. 5와 같이 hostname을 읽어 해당 node의 형상설정 파일을 작성하고 형상 변경된 소프트웨어를 restart를 시킨다. 형상설정 파일은 “echo” 구문을 사용하여 쉽게 작성할 수 있다.

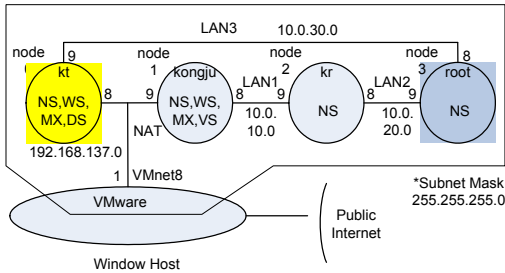


Fig. 4. Platform Network Configuration

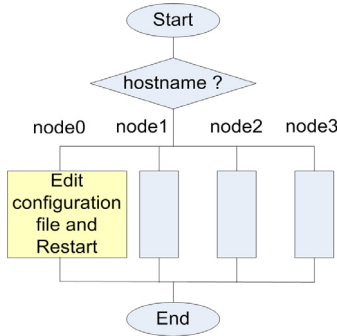


Fig. 5. The Flowchart of pnet.init

2.2 라우팅 시뮬레이터 개발

IP 네트워크를 실습하기 위해서는 라우팅 테이블의 조작으로 네트워크 형상을 자유자재로 변경할 수 있어야 하므로 테이블 기반 라우팅 기법에 대한 사전 학습이 필요하다. 이러한 학습을 위해 ns에 IP 주소할당과 라우팅 테이블 관리 기능을 추가한 라우팅 시뮬레이터를 개발하였다. 기능 추가를 위해 할당된 IP 주소 정보를 담은 node_table[]과 라우팅 테이블 정보를 담은 route_table[]의 자료구조를 도입하였다. 이들 자료구조로 각 노드의 IP 주소와 라우팅 테이블 정보를 기존 ns 프로그램의 자료구조와 완전 분리하였다. 그 대신 route_table[]에 새로운 레코드를 추가 또는 삭제할 경우 ns가 제공하는 tcl 라이브러리 함수를 이용하여 ns 프로그램의 커널 라우팅 테이블에 해당 루트를 추가 또는 삭제하도록 하였다.

추가된 자료구조를 상세히 살펴보자. node_table[]

의 각 레코드는 다음의 5개의 영역으로 구조화 된다: node ID(IDentification), IP address, subnet mask, interface 및 flag. 각 레코드는 node ID와 interface가 지시하는 노드의 인터페이스에 IP address와 subnet mask가 지시하는 IP 주소와 서브넷 마스크가 할당됨을 나타낸다. 여기서 flag는 해당 레코드의 valid/invalid 여부를 표시한다. route_table[]의 각 레코드는 다음의 6개 영역으로 구조화 된다: node ID, destination address, subnet mask, gateway, interface 및 flag. 각 레코드는 node ID가 지시하는 노드에 destination address가 지시하는 목적지에 대한 경로 정보를 나타내는데, 경로상의 다음 단 주소와 다음 단으로 전달하기 위해 사용되는 인터페이스가 각각 gateway와 interface에 수록된다. 마찬가지로 flag는 해당 레코드의 유효성 여부를 표시한다. 라우팅 시뮬레이션을 위해 추가된 명령어는 Table 1과 같다.

Table 1. Commands added

Commad	Description
add-ip	Add IP address
delete-ip	Delete IP address
print-ip	Print IP address
add-route	Add route
delete-route	Delete route
print-route	Print route

실제 사례를 들어 개발된 라우팅 시뮬레이터의 동작을 살펴본다. 할당된 210.125.213.0/24를 5개의 서브넷으로 나누고 각 서브넷은 30대의 호스트를 수용하도록 하였다. 그러면 서브넷 마스크는 모두 255.255.255.224가 된다. 학습 네트워크의 토폴로지는 Fig. 6과 같이 6개의 노드로 구성한다.

먼저 add-ip 명령을 사용하여 각 노드의 인터페이스에 IP 주소와 서브넷 마스크를 할당한다. add-ip 명령어 사용법은 다음과 같다: add-ip 노드번호 IP주소 서브넷 마스크 인터페이스. 예로 노드 2의 인터페이스 2에 IP 주소 210.125.213.97과 서브넷 마스크 255.255.255.224를 할당하고자 할 때 “add-ip 2 210.125.213.97 255.255.255.224 2”를 입력하면 된다. 각 노드의 모든 인터페이스에 IP 주소와 서브넷 마스크를 할당한 후 “print-IP” 명령어를 입력하면 IP 주소 할당 내역이 출력되므로 제대로 할당되었는지 확인할 수 있다.

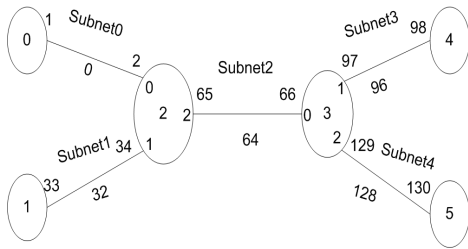


Fig. 6. Learning Network Topology

다음으로 add-route 명령어를 사용하여 각 노드의 라우팅 테이블을 작성한다. 명령어 사용법은 다음과 같다; add-route 목적지네트워크 서브넷마스크 gateway 인터페이스. 예로 노드 2에서 목적지 210.125.213.160 네트워크로 가는 레코드를 추가하고자 할 때 “add-route 210.125. 213.160 255.255.255.224 210.125.213.98 2”를 입력하면 된다. 각 노드에서 모든 목적지에 대한 라우팅 레코드를 추가한 후 “print-route” 명령어를 입력 하면 각 노드의 라우팅 테이블 내용이 출력되므로 제대로 추가되었는지 확인할 수 있다.

마지막으로 ns-2가 제공하는 tcl 스크립트를 이용하여 노드 0과 1, 노드 1과 2, 노드 2와 3, 노드 3과 4, 및 노드 3과 5사이를 각각 바인딩 하고 전이중 링크들을 구성한 후 노드 0에서 노드 5로 ping 시험을 하면 실제 패킷이 움직이는 모습을 관찰하면서 네트워크의 라우팅 동작을 확인할 수 있다. Fig. 7은 ping 패킷이 실제 배달되는 모습을 보여준다.

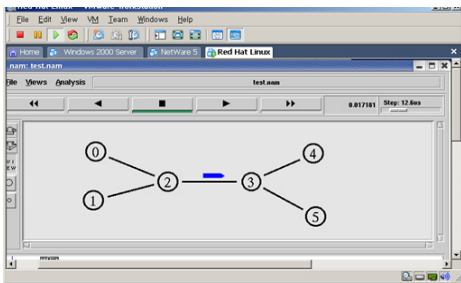


Fig. 7. Ping test result

2.3 실습 시나리오

실습에 사용된 Fig. 4의 플랫폼, 즉 실습 환경을 살펴 보면 윈도 10 운용체제가 설치된 윈도 호스트에 VMware Workstation 15 Pro를 탑재하고 여기에 4개의 리눅스 가상머신을 생성하여 플랫폼 네트워크를 구성

하였다. 많은 종류의 리눅스 배포판이 있으나 네트워크 기능은 모두 동일하므로 가벼운 레드햇 9.0을 사용하였다. 통신 절차를 확인하고 분석하기 위한 프로토콜 분석기로 윈도 호스트용의 wireshark과 리눅스가 제공하는 tcpdump를 사용하였다. 가상머신의 4개의 호스트 네임을 각각 node0~3으로 설정한 후 가동한 플랫폼의 화면 모습은 Fig. 8과 같다.

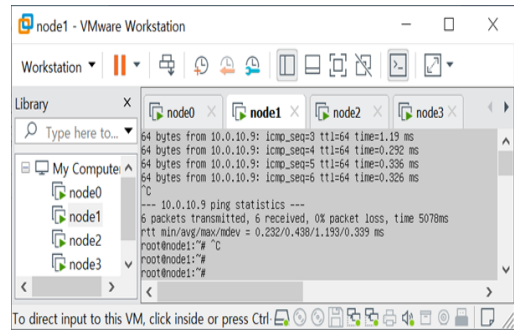


Fig. 8. Graphic User Interface of Platform

플랫폼에서 수행할 실습은 인터넷 통신, 기본 네트워크 서비스 및 응용 서비스 시스템의 3 단계로 나누어 진행한다.

2.3.1 인터넷 통신 실습

pnet.init으로 Fig. 3의 형상에 맞게 각 node별 IP 환경을 설정하면 LAN 내에서만 통신이 가능한 인트라넷 실습을 수행할 수 있다. node0에서 192.168.137.9의 물리주소가 ARP 캐시에 있으면 삭제한 후 “ping 192.168.137.9”의 명령을 실행시키면 Fig. 9와 같이 ARP 동작으로 192.168.137.9의 물리주소가 00:0c:29:65:3e:f5임을 알아낼 수 있고, 프레임 헤더는 목적지와 근원지 주소 및 type 부분으로 구성됨을 확인할 수 있다.

```
ARP ... Who has 192.168.137.9? Tell 192.168.137.8
ARP ... 192.168.137.9 is at 00:0c:29:65:3e:f5
```

(a)

```
Ethernet II, Src: Vmware_20:7a:49 (00:0c:29:20:7a:49)
> Destination: Broadcast (ff:ff:ff:ff:ff:ff)
> Source: Vmware_20:7a:49 (00:0c:29:20:7a:49)
Type: ARP (0x0806)
```

(b)

Fig. 9. ARP Practice Result and Ethernet Head
(a) ARP Request and Reply (b) Ethernet Frame Head

인트라넷 통신이 확인되면 라우팅 시뮬레이터로 라우팅 기법을 학습한다. add-ip 명령어로 각 node의 IP 주소와 서브넷 마스크를 설정하고, add-route 명령어로 Fig. 3의 각 네트워크들에 대한 정적 루트를 추가한다. 그리고 ping 시험으로 정적 라우팅 테이블의 오류 유무를 확인하고 오류 시 수정 및 재확인을 통해 테이블 기반 데이터그램 라우팅 기법을 학습한다.

라우팅 학습 결과를 바탕으로 pnet.init에 정적 라우팅 테이블 작성 구문을 추가하고 리눅스를 라우터로 동작시켜 IP, ICMP, RIP 및 OSPF 프로토콜의 인터넷 통신 실습을 해볼 수 있다. IP 통신은 간단한 비연결 및 비확인형 통신으로서 단편화 과정에 대해서만 실습한다. node0에서 “ping 10.0.20.9 -l 5000 -t 1” 명령 실행시 ethernet의 MTU(maximum transfer Unit)가 1500 바이트이므로 4개의 데이터그램으로 단편화되어 전달된다. 이러한 단편화 과정을 Fig. 10과 같이 확인할 수 있다. Fig. 10에서 모든 단편들의 ID 값이 cd41로 원 데이터그램의 값을 상속받고, 각 단편의 offset 값이 0, 1480, 2960 및 4440이 됨을 관찰할 수 있다.

대표적인 목적지 도달불가, 경로 재지정 및 TTL(Time To Live) 시간초과 ICMP 메시지에 대한 실습을 살펴본다. 이러한 메시지들이 발생할 수 있는 비정상 상태를 만들어 줘야 하므로 다소 복잡한 준비과정이 필요하다.

```
IPv4 ... Fragmented IP protocol (proto=ICMP 1, off=4440, ID=cd41)
IPv4 ... Fragmented IP protocol (proto=ICMP 1, off=2960, ID=cd41)
IPv4 ... Fragmented IP protocol (proto=ICMP 1, off=1480, ID=cd41)
ICMP ... Echo (ping) request id=0x840c, seq=1/256, ttl=64 (reply)
```

Fig. 10. Datagram Fragmentation Practice Result

node0의 라우팅 테이블에서 10.0.10.0/24와 0.0.0.0/32 레코드를 삭제하고 원도 호스트의 라우팅 테이블에서 목적지 10.0.10.0/24 에 대한 다음 홉을 비최적 경로인 192.168.137.8로 변경한 후 원도 호스트에서 “ping 10.0.10.8” 명령을 실행하면 이 메시지를 받은 node0는 라우팅 테이블에 해당되는 레코드가 없어 목적지 도달불가 메시지를 반송하는데 이때 Fig. 11의 (a)와 같은 메시지의 발생을 관찰할 수 있다. node0의 라우팅 테이블을 원상복구 한 후 다시 “ping 10.0.10.8” 명령을 실행하면 원도 호스트에게 비최적 경로이므로 10.0.10.0/24 에 대한 다음 홉을 최적 경로인 192.168.137.9로 경로를 재지정하라는 메시지를 보내게 됨을 Fig. 11의 (b)와 같이 확인할 수 있다. TTL 시간초과를 실습하기 위해서 먼저

Fig. 4의 원도 호스트에서 node0, node3, node2를 거쳐 node1의 10.0.10.8과 양방향 통신하도록 모든 노드의 라우팅 테이블을 변경한 후 원도 호스트에서 “tracert 10.0.10.8” 명령을 실행한다. 그러면 경로가 추적되면서 생성된 경로추적 관련 ICMP echo 요청과 TTL 시간초과 메시지를 관찰할 수 있다. Fig. 11의 (c)는 경로상의 3번째 라우터가 10.0.20.8임을 보여주는데, 이는 경로상의 3번째 라우터인 10.0.20.8에서 TTL 값이 0이 됨을 발견하여 반송하는 시간초과 메시지가기 때문이다.

동적 라우팅 기반 인터넷 통신 실습을 위해 quagga 패키지를 설치한다. quagga와 RIP 및 OSPF 프로토콜의 동작 환경 설정 구문을 추가하고 pnet.init를 재 실행하면서 각 노드의 라우팅 테이블이 완성되어 가는 과정을 살펴보고 프로토콜 분석기로 RIP와 OSPF 메시지를 캡처하여 분석해본다. 그리고 각 노드들 사이에 ping 시험으로 통신이 되는지 확인해본다.

추가적으로 라우팅 프로토콜의 수렴시간이 매우 중요하므로 RIP 프로토콜에 대해 Fig. 4의 LAN1에 장애가 발생할 때의 수렴시간, 그리고 그 장애가 복구될 때의 수렴시간을 측정하고 분석해본다. 먼저 node1의 10.0.10.8 인터페이스(이하 A지점)와 node3의 10.0.20.9(이하 B지점)에 tcpdump로 ping 트래픽을 감시하도록 설정한 후 node2에서 “ping 10.0.10.8” 명령 실행으로 무한 ping 시험을 실시한다. node2에서 LAN1을 down시키면 경로가 차단되므로 A 지점의 ping 패킷 발생이 중단되고, RIP 프로토콜에 의해 node2-node3-node0-node1 사이의 양방향 통신이 가능하도록 모든 노드의 라우팅 테이블이 자동 갱신된다. 갱신되는데 소요되는 시간이 바로 장애시 수렴시간이 된다. 수렴시간이 지나면 B지점에서 ping 패킷이 관찰된다.

node2에서 LAN1을 up하면 node2와 node1사이의 직접경로가 추가되고, 추가된 경로가 RIP 프로토콜에 의해 모든 노드의 라우팅 테이블에 반영되면 B지점의 ping 패킷이 사라지고 대신 A 지점의 ping 패킷이 관찰되는데, LAN1 up 시점부터 A지점 ping 패킷 관찰 때까지의 시간이 바로 복구시 수렴시간이 된다. 실습결과 장애발생시 수렴시간의 최소, 최대 및 평균값은 각각 2초, 47초 및 15초로 측정되었고, 장애 복구시 수렴시간의 최소, 최대 및 평균값은 각각 1초, 48초 및 11초로 측정되었다. 이러한 결과는 1초 이내의 감지시간을 갖는 리눅스의 링크 down/up 상태 감지 메카니즘과 매 30초 마다 자신의 라우팅 테이블 값을 주변에 통지하는 RIP 프로토콜의 동작 방식으로부터 설명될 수 있다

```
Internet Control Message Protocol
Type: 3 (Destination unreachable)
Code: 0 (Network unreachable)
Checksum: 0xfcff [correct]
[Checksum Status: Good]
Unused: 00000000
> Src: 192.168.137.1, Dst: 10.0.10.9
(a)
```

```
Internet Control Message Protocol
Type: 5 (Redirect)
Code: 1 (Redirect for host)
Checksum: 0xb14c [correct]
[Checksum Status: Good]
Gateway address: 192.168.137.9
(b)
```

```
> Internet Protocol Version 4, Src: 10.0.20.8,
▼ Internet Control Message Protocol
Type: 11 (Time-to-live exceeded)
Code: 0 (Time to live exceeded in transit)
(c)
```

Fig. 11. ICMP Practice Result

(a) Destination Unreachable (b) Route Redirection
(c) TTL Time Exceeded

2.3.2 기본 네트워크 시스템 실습

먼저 DNS 실습에 대해 살펴본다. DNS 구축을 위해 수정하여야 할 파일은 도메인을 지정하는 /etc/named.conf, 루트 네임서버의 IP 주소 정보를 등록하는 /var/named/named.ca 및 해당 zone의 IP 주소 해석 데이터베이스를 등록하는 /var/named/domain.zone 의 3개이다. 모든 네임 서버가 갖고 있는 루트 네임서버의 주소 정보를 실습 플랫폼의 루트 네임서버의 주소로 변경한다. 구체적으로 /var/named/named.ca 파일내 13개 루트 네임서버 중 B ~ M의 12개는 삭제하고 A 루트 네임서버의 IP 주소를 10.0.30.8로 변경한다. 루트 및 kr 네임서버에 각각 kr과 kongju 도메인에 대한 위임내역을 작성하고 각 네임서버의 존 파일 작성 후 네임 서비스를 가동시킨다. 그리고 a.kt.co.kr에 해당하는 윈도 호스트에서 "ping www.kongju.kr" 명령을 실행하면 Fig. 12와 같은 DNS 메시지를 관찰할 수 있다. 이는 Fig. 13과 같이 kt 네임서버가 계층적 질의를 통해 네임을 완전하게 해석한 결과인 192.168.137.9의 IP 주소 값을 a.kt.co.kr에게 반환해주는 재귀적(recursive) 네임해석의 결과이기 때문이다.

DHCP 실습에 대해 살펴본다. DHCP 서버를 탑재한 node1 에서 dhcpd.conf 파일을 편집하여 임대할 주소의 범위, 임대 기간 및 기타 네임서버, 도메인 명 등의 클라이언트에게 제공할 IP 환경정보를 입력하고 dhcpd를

가동시킨다. 윈도우 호스트를 IP 주소 자동공급으로 설정하면 DHCP 프로토콜이 가동되어 IP 환경정보가 공급된다. Fig. 14와 같이 DHCP 서버를 발견하는 Discover와 Offer, 발견된 서버로부터 정보를 공급받는 Request와 Ack 메시지에 의해 IP 환경정보 제공 절차가 이루어짐을 확인할 수 있다.

```
DNS query 0xf6e3 A www.kongju.kr
DNS query 0x7736 A www.kongju.kr OPT
DNS query response 0x7736 A www.kongju.kr NS ns.kr A 10.0.20.8 OPT
DNS query 0x3b9b A www.kongju.kr OPT
DNS query response 0x3b9b A www.kongju.kr NS ns.kongju.kr A 10.0.10.8
DNS query 0xb220 A www.kongju.kr OPT
DNS query response 0xb220 A www.kongju.kr A 192.168.137.9 NS kongju.kr
DNS query response 0xf6e3 A www.kongju.kr A 192.168.137.9 NS kongju.kr
```

Fig. 12. DNS Practice result

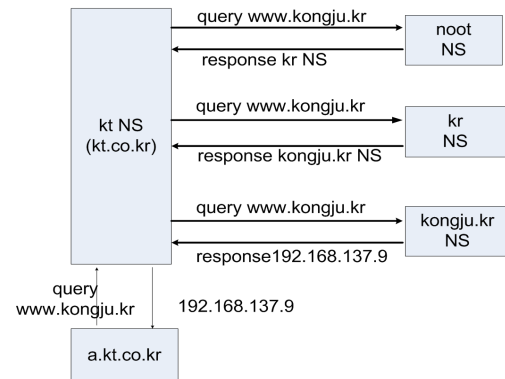


Fig. 13. Recursive Name Resolving

DHCP	...	DHCP Discover
DHCP	...	DHCP Offer
DHCP	...	DHCP Request
DHCP	...	DHCP ACK

Fig. 14. DHCP Practice Result

웹 서비스는 apache2 서버를 사용하여 실습한다. node0와 node1에 apache2를 설치하고 윈도 호스트의 웹브라우저에서 이의 동작을 확인한다. 웹 서비스 실습 시 tcp 실습도 같이 병행할 수 있다. Fig. 15의 (a)와 같이 웹 서비스에 사용되는 HTTP 메시지를 확인할 수 있고, (b)와 같이 3-way handshake에 의해 tcp 연결이 설정됨을 확인할 수 있다. 웹서버에의 접속이므로 목적지 포트번호가 웹서버에 할당된 80이 됨을 확인할 수 있다.

```

HTTP ... GET / HTTP/1.1
HTTP ... HTTP/1.1 200 OK (text/html)
    (a)

TCP ... 50704 → 80 [SYN] Seq=0 Win=64240 Len=0
TCP ... 80 → 50703 [SYN, ACK] Seq=0 Ack=1 Win=1460
TCP ... 50703 → 80 [ACK] Seq=1 Ack=1 Win=131328
    (b)
    
```

Fig. 15. Web Practice Results
(a) HTTP protocol (b) 3-way handshake connection

Email은 레드햇 9.0이 제공하는 sendmail과 pop3를 이용하여 실습한다. 이들 파일의 환경설정 파일 작업 후 DNS zone 파일에서 MX 레코드를 추가하여 도메인 내 메일서버를 추가한다. 메일링의 동작확인은 윈도우 호스트에서 kongju.kr과 kt.co.kr 메일서버에 각각 kdhong1과 kdong2의 메일 id를 등록한 후 kdong1에서 kdong2로 메일을 보내고, kdong2에서 회신을 보내어 kdong1이 그 회신을 수신하였는지 여부로 메일의 동작 결과를 확인할 수 있다.

2.3.3 응용 서비스 시스템 실습

인터넷 응용 서비스 시스템은 매우 다양하나 본 연구에서는 최근에 각광을 받고 있는 인터넷 전화 시스템에 대해서 실습한다. 사용한 공개 소프트웨어는 asterisk SIP server 17.5.1과 OfficeSIP Softphone 1.0 이다. asterisk sip server를 탑재하기 위해 node1을 ubuntu 18.0.4로 업그레이드하였다. 양자간 통화를 위해 윈도 노트북을 추가하였으며, 이를 실습 플랫폼 네트워크에 수용하기 위해 Fig. 4에서 192.168.137.0/24의 NAT 네트워크를 192.168.0.0/24의 bridged 네트워크로 변경하였다. bridged 유형은 외부 네트워크에 직접 연결하는 것으로서 외부 컴퓨터의 수용에 적합하다. 192.168.0.0/24에 연결되는 윈도 호스트, 윈도 노트북 및 node1에 각각 12, 11 및 17의 호스트 식별자를 부여하였다.

윈도 호스트 및 노트북에 Softphone을 설치한 후 이들에 각각 tjkim1과 tjkim2의 이용자를 등록하였다. tjkim2에서 tjkim1으로 proxy 서버 기반 VoIP 호를 발생시켜 통화를 한 결과 Fig. 16과 같은 메시지를 관찰할 수 있었고, 이로부터 Fig. 17의 Proxy 서버 기반 호 절차와 일치함을 확인할 수 있었다.

```

SIP/SDP ... Request: INVITE sip:tjkim1@kongju.kr |
SIP ... Status: 100 Trying |
SIP/SDP ... Request: INVITE sip:192.168.0.12:50234;tr
SIP ... Status: 100 Trying |
SIP ... Status: 180 Ringing |
SIP ... Status: 180 Ringing |
SIP/SDP ... Status: 200 OK |
SIP/SDP ... Status: 200 OK |
SIP ... Request: ACK sip:tjkim1@192.168.0.17:5060
RTP ... PT=ITU-T G.711 PCMU, SSRC=0x959A6D92, Seq
RTP ... PT=ITU-T G.711 PCMU, SSRC=0x15F03449, Seq
SIP ... Request: BYE sip:tjkim1@192.168.0.17:5060
SIP ... Status: 200 OK |
SIP ... Request: BYE sip:192.168.0.12:50234;addr
SIP ... Status: 200 OK |
    
```

Fig. 16. VoIP Practice Result

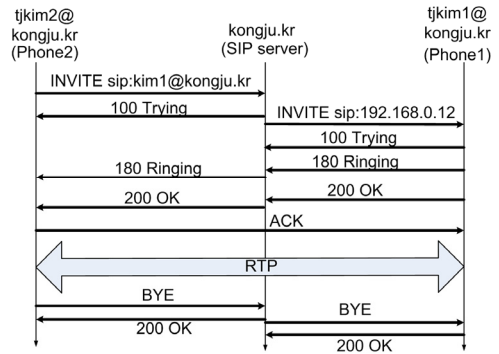


Fig. 17. VoIP Call Flow

3. 결론

본 논문에서는 IP 네트워크 자체와 기본 네트워크 시스템 및 응용 서비스 시스템으로 구성되는 IP 네트워크 시스템의 제반 기능을 실습할 수 있는 실습 플랫폼을 제안하였고, 제안된 플랫폼에서 수행한 실습 결과를 바탕으로 실습 시나리오를 제시하였다.

인터넷의 가입자와 서비스 제공자 및 서비스 유형 분석으로부터 이들을 모두 수용할 수 있는 간단하고 융통성 있는 실습 도메인 체계를 도입하고, 각 실습 도메인에 배치할 서버들을 도출하였다. 실습 도메인 체계와 도출된 서버들을 모두 수용할 수 있는 최소한의 형상을 갖는 플랫폼 네트워크 구조를 설계하였다. 추가적으로 널리 활용되고 있는 ns에 IP 주소 할당과 라우팅 테이블 관리 기능을 추가하여 테이블 기반 데이터그램 라우팅 기법을 사전 학습할 수 있는 라우팅 시뮬레이터를 개발하였다.

본 연구에서 설계된 실습 플랫폼은 별도의 장비나 비

용의 부담 없이 개인용 컴퓨터에서 간편하게 실습을 수행할 수 있어 네트워크 입문자에게 유용하리라 기대된다.

References

- [1] K. Park, J. Jeong and K. Jeong, "Implementation of Embedded Education Router System", Journal of the Korea Society of Computer and Information, Vol.18, No.5, pp.9-17, 2013.
- [2] N. Iguchi, "Virtual IP Network Practice System with Software Agent", Proceeding of Conference on Complex, Intelligent, and Software Intensive Systems, pp.711-720, 2017.
- [3] N. Iguchi, M. Kurauchi, Y. Tsujimoto and T. Nishijima, "IP Network Construction Learning System utilizing Virtual Router", Proceedings of the IEEE Pacific Rim Conference on Communications, Computers and Signal Processing, pp.107-112, 2011
- [4] N. Iguchi and K. Kitazawa, "Development of Hands-on IP Network Practice System with Automatic Scoring Function", Proceeding of the Seventh International Conference on Complex, Intelligent, and Software Intensive Systems, July 2013
- [5] T. Karasuno, K. Masuga, Y. Taniguchi and N. Iguchi, "A Study on Pseudo Cooperative Practice in a Cloud based Hands-on IP Network Practice System", Proceeding of the Global Conference on Consumer Electronics, October 2016
- [6] S. Hirahata, Y. Taniguchi, and N. Iguchi, "Development of Instructor Support Function for Cloud-based IP Networking Practice System", Proceeding of the International Conference on Consumer Electronics, May 2019.
- [7] T. Kim, "An Implementation of IP Network Practice System using Network Simulator", Proceeding of Autumn Conference of Korea Academia-Industrial Cooperation Society, pp. 318-320, November 2018.
- [8] T. Kim and S. Ryoo, "Design of Open Software based Internet Practice Platform", Proceeding of Spring Conference of Korea Academia-Industrial Cooperation Society, pp. 1053-1056, July 2020.
- [9] "The Network Simulator - ns-2", <https://www.isi.edu/nsnam/ns/>

김 태 준(Tae-Joon Kim)

[종신회원]



- 1982년 3월 ~ 1996년 2월 : 한국 전자통신연구원 근무
- 1996년 3월 ~ 현재 : 공주대학교 정보통신공학부 교수

〈관심분야〉
정보통신, 네트워크