

동적 IP Address를 사용하는 인터넷 서버 구축을 위한 게이트웨이

김 원 중[†] · 양 현 택^{††}

요 약

현재 대부분의 가정이나 소규모 기업에서는 인터넷 서비스를 이용하기 위하여 ADSL(Asymmetric Digital Subscriber Line)나 케이블모뎀을 사용한다. 4바이트로 구성되는 현재의 인터넷 IP 주소체계(IPv4)에서는 할당 가능한 IP 주소가 부족하여, 동적으로 공인(Public) IP 주소를 할당하는 방식을 사용하고 있다. 이러한 동적인 IP 주소를 할당받은 시스템은 모든 인터넷 서비스를 이용하는 데에는 아무런 제약이 없지만, 자신이 각종 인터넷 서버 기능을 수행하는 데는 많은 문제점이 있다. 본 논문에서는 동적으로 변하는 IP 주소를 가진 시스템에서 인터넷 서버 서비스(Telnet, FTP, HTTP, Mail 등)가 가능하도록 하는 게이트웨이(Gate-D)를 설계하였으며, 실험적으로 Gate-D를 이용한 Telnet 서버의 구축을 통하여 유용성을 확인하였다.

The Gateway for Internet Server Implementation using Dynamic IP Address

Won-Jung Kim[†] · Hyun-Teak Yang^{††}

ABSTRACT

Recently most of the home and small-size company use ADSL(Asymmetric Digital subscriber Line) or Cable Modem for using Internet Services. The number of Internet IP Address in current Internet IP Address System(IPv4) that is consisted of 4byte is almost empty, so generally the IP Address assigned dynamically is used. This way is just OK in general uses, but not OK in Internet Servers operation. This paper designed the gateway(Gate-D) system which is enable any system that get IP Address assigned dynamically to serve Internet Server Services(Telnet, FTP, HTTP, Mail, etc ...), and made sure the worth by using Telnet Server.

키워드 : 공인 인터넷 주소(Public IP Address), 게이트웨이(Gateway), 프로토콜(Protocol), 비대칭 디지털 가입자 회선(ADSL)

1. 서 론

1972년 ARPANET에서부터 시작된 인터넷에서 우리는 Gopher, Veronica, Archie, FTP(File Transfer Protocol), Telnet, IRC(Internet Relay Chat), Usenet 등의 서비스를 이용하였고, 현재는 TCP/IP 프로토콜을 사용하는 Web(World Wide Web)이 등장하여 가정에서도 아주 편리한 GUI 환경의 웹 브라우저를 사용하여 망상대화와 같은 인터넷의 사이버 공간을 효과적으로 향해하고 있다. 또한 이들 인터넷상의 각종 서버와의 물리적 연결을 위해서는 모뎀연결, LAN, ISDN, xDSL 등의 기술을 거쳐 현재는 기관이나 연구소 등의 LAN 사용자를 제외한 대부분의 사용자들은 ADSL B&A(Building

and Apartment), ADSL(Asymmetric Digital Subscriber Line), 또는 케이블 모뎀을 사용하여 인터넷 서비스를 제공하는 호스트에 접근하고 있다.

2001년 3월 인터넷 마케팅 전문기관인 베스트사이트(www.bestcote.com)와 한국갤럽의 공동조사에 의하면 국내 인터넷 인구는 전국민의 과반수를 넘어섰고, 52.9%는 가정에서 ADSL를 사용하여 인터넷에 접속하는 것으로 나타났다.

TCP/IP 프로토콜을 사용하는 인터넷 네트워크상의 모든 호스트들은 인터넷에 접근하기 위하여 전세계적으로 고유한 IP Address를 배정받아야 한다. 그러나 현행 4바이트로 되어 있는 인터넷 IP Address 체계(IPv4)는 인터넷의 기하급수적인 양적 팽창으로 할당 가능한 IP Address공간이 거의 고갈되어 가고 있는 상태이다. 2000년 12월말까지 우리나라가 보유하고 있는 인터넷 IP Address는 1,853만개로 같은 시점 국내 인터넷 인구 1,904만명에도 미치지 못하고 있는 실정이다[9].

* 본 연구는 광주·전남테크노파크 연구개발사업에 의해 수행되었음.
† 종신회원 : 순천대학교 정보통신공학부 교수
†† 정 회 원 : (주)모인벨리 대표이사
논문접수 : 2001년 8월 29일, 심사완료 : 2001년 10월 25일

따라서 이러한 IP Address의 부족문제를 해결하기 위하여, ADSL B&A 인터넷 접속 서비스에서는 사용자를 사설 IP Address(Private IP Address)에 할당한 후 NAT를 이용하여 인터넷에 연결시키고 있으며, 일반 ADSL 인터넷 접속 서비스에서는 사용자에게 공인 IP Address(Public IP Address)를 부여하기는 하지만, DHCP 기법을 사용하여 사용자에게 유동 IP Address를 부여하기 때문에 사용자가 계속 변하지 않는 고정적인 IP Address를 가지고 웹서버, FTP서버, Telnet, News 등의 인터넷 서버의 기능을 제공하는 것이 불가능하다. 즉, 이들 인터넷 접속 서비스를 사용하는 사용자가 다른 인터넷 서비스를 이용하는 것에는 제약이 없으나, 본인의 고정된 IP Address를 가지지 못하기 때문에 인터넷에 접속이 가능할지라도 각종 인터넷 서버가 되는 것은 불가능하다. 본 논문에서는 이러한 문제점을 해결하기 위한 게이트웨이를 설계 및 구현하였으며, 일차적으로 인터넷상에서 유동 IP Address를 가진 서버들에 대하여 Gate-D를 통한 Telnet 접근이 가능함을 보였다.

본 논문의 구성은 다음과 같다. 제 2장에서는 관련연구로서 인터넷 IP Address 부족문제를 해결하기 위한 방법들에 대해 살펴보고, 제 3장에서는 시스템구조와 각 모듈들의 기능에 대해서 설명한다. 제 4장에서는 시스템의 구현 및 실험 결과를 제시하고, 제 5장에서 결론과 향후 연구방향에 대해 기술한다.

2. IP Address 부족 문제 해결을 위한 방법들

부족한 IP Address문제를 해결하기 위하여 사용되는 방법으로는 사설 IP Address, Proxy 서버, NAT, DHCP 등이 있다[10].

일반적으로 ISP(Internet Service Provider)는 NAT와 DHCP 방식을 사용한다. 일부 ISP에서 가입자를 모집하기 위하여 고정된 공인 IP Address를 할당해주는 경우가 있으나, 대부분의 일반 ISP는 고정된 공식 IP Address를 사용하기 위해서는 비싼 사용료를 지불하는 경우에만 제공하며, 일반적으로는 NAT의 ADSL B&A나 DHCP의 ADSL, Cable Modem 등으로 운영하고 있다.

2.1 사설 IP Address

기업에서 보유하고 있는 컴퓨터를 인터넷과 관련하여 분류하면 인터넷에 접속하지 않아도 되는 컴퓨터, Mail, Web과 같은 인터넷 서비스를 제공하는 컴퓨터, 그리고 인터넷에 접근하여야 하는 컴퓨터 등 세 가지로 나눌 수 있다. 또, 인터넷에 접속하지 않아도 되는 컴퓨터는 네트워크에 포함이 되어 있는 것과 포함되지 않은 두 가지로 구분될 수 있다. 그러나, 대부분의 기업에서는 인터넷에 접근하지 않아도 되는 컴퓨터일지라도 업무의 효율성을 위해 자체적으로 네트워크

에 연결시켜 사용하고 있다. 자체적인 네트워크를 구성하여 동작하는 컴퓨터들에게 공인 IP Address를 할당하는 것은 자원의 낭비이다. 이러한 문제점을 해결하기 위하여 제안된 것인 사설 IP Address이다. 사설 IP Address는 인터넷의 접속은 필요 없지만 TCP/IP Protocol을 이용한 자체 네트워크 망을 구축하기 위해서 필요한 것으로서 Internic에서 권고하는 사설 IP Address 범위는 A Class인 경우 10.0.x.x, B Class인 경우 172.16.x.x, C Class인 경우 192.168.x.x이다[1, 8, 10].

사설 IP Address를 가지는 컴퓨터가 인터넷에 접속하기 위해서는 Proxy 서버나 NAT와 같은 별도의 게이트웨이가 필요하다.

2.2 Proxy 서버

사설 IP Address를 이용해 기업 네트워크를 구축한 경우 정상적인 방법으로는 인터넷에 접근할 수 없다. 이런 문제점을 해결하기 위해서 사용될 수 있는 방법이 Proxy 서버를 이용하는 것이다. Proxy 서버는 사설 IP Address와 인터넷 간의 Gateway 역할을 한다.

Proxy 서버를 도입함으로써 얻을 수 있는 이점은 크게 세 가지이다.

첫째, 공인 IP Address는 Proxy 서버에만 할당하고 나머지 컴퓨터들은 사설 IP Address를 사용함으로써 IP Address를 효율적으로 사용할 수 있다.

둘째, 캐쉬 기능 제공으로 클라이언트가 동일한 정보를 요청할 때, Proxy 서버는 로컬 캐쉬의 정보를 제공함으로써 네트워크 사용의 효율성을 높일 수 있다.

셋째, 외부의 침입으로부터 내부 네트워크를 보호할 수 있다는 것이다. 사설 IP Address를 가진 컴퓨터에 접근하고자 할 때는 Proxy 서버를 경유하여야만 가능하므로, Proxy 서버에서 필터링 기능을 제공하여 외부 침입을 막을 수 있다. 대부분의 Proxy 서버 시스템은 Firewall로 작동되며, 네트워크 구조는 NAT와 비슷하다[6-8, 10].

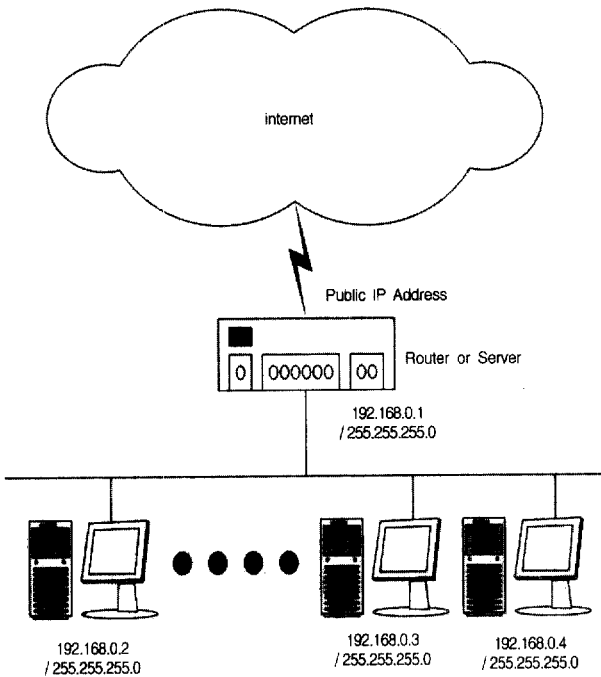
2.3 NAT(Network Address Translation)

Proxy 서버와 비슷한 개념을 가진 것으로서 Proxy 서버가 사용자 브라우저에 등록이 된 사용자만 가능하다면 NAT는 전혀 사용자가 알지 못하게 인터넷과 연결작업을 수행한다. NAT에서 인터넷망과의 연결을 위해서는 내부 네트워크의 사설 IP Address를 하나 또는 여러 개의 공인 IP Address로 사상시키는 과정이 필요하다. 그리고 들어오는 연결에 대해서도 공인 IP Address를 다시 사설 IP Address로 변환한다. 이러한 Address 변환 과정을 거치지 않고는 사설 IP Address의 컴퓨터에서 인터넷 접근을 사용할 수 없다. 사설 IP Address와 공인 IP Address 간의 변환과정은 변환 테이블(Translation Table)을 참조해서 이루어지게 된다. 따

라서, 한 개 이상의 공인 IP Address를 가지는 Exit point가 있다면, 각 NAT들은 같은 변환테이블을 참조하여야 한다.

또한 서비스 방식에 따라 정적 변환 방식과 동적 변환 방식이 있다. 정적 변환 방식은 인터넷에서 고정적인 Address가 필요로 하는 서버에 적용할 수 있으며, 동적 변환 방식은 서버가 아닌 Workstation에서 사용할 수 있다.

NAT가 가지고 있는 장점은 부족한 공인 IP Address를 효율적으로 공유할 수 있고, 내부 네트워크를 보호할 수 있다. 또한, 굳이 외부 인터넷이 연결되지 않더라도 사설 IP Address로 구성된 네트워크의 노드끼리는 서로 연결이 가능하다는 장점이 있다. 단점으로는 공인 IP Address에서 사설 IP Address로 접근을 할 수 없다는 점이다. (그림 1)은 NAT의 기본 동작 방식을 나타낸 것이다. 가운데의 Router or 서버는 1개 혹은 여러 개의 공인 IP Address를 할당받아서 밑단의 사설 IP Address를 가지는 컴퓨터와 외부 인터넷간의 연결을 위한 사상 작업을 수행한다[2, 8, 10].



(그림 1) NAT 기본 구성

2.4 DHCP(Dynamic Host Configuration Protocol)

DHCP는 네트워크 관리자들이 조직내의 네트워크 상에서 IP Address를 중앙에서 관리하고 할당해 줄 수 있도록 해주는 프로토콜이다. 인터넷의 TCP/IP 프로토콜에서는 고유한 IP Address를 가져야만 인터넷에 접속할 수 있도록 한다. 앞의 NAT방식이 사설 IP Address를 적은 개수의 공인 IP Address에 사상하는 방식을 사용한다면 DHCP는 여러 대의 컴퓨터가 서버에 접속하여 동적으로 공인 IP Address를 할당받는 방식이다. 즉, 연결되어 있는 전체 컴퓨터가 동시에 인터넷을 사용하지 않는다는 가정 아래 인터넷을 사용할 때

만 공인 IP Address를 가지고 있는 DHCP서버에서 IP Address를 할당 받은 후, 인터넷 사용이 끝나면 다시 반납하는 형태를 사용하는 방법이다. 고정 IP Address가 필요한 경우에는 고정적인 IP Address의 부여가 가능하다.

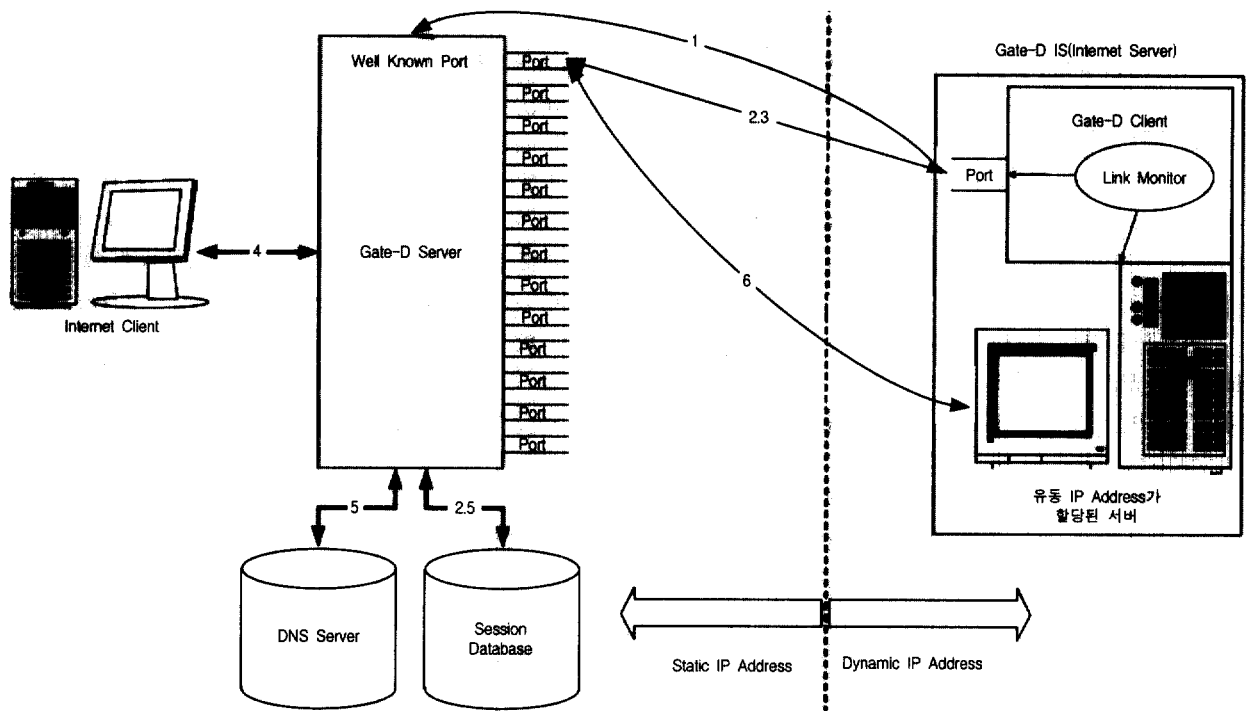
장점으로는 사용 가능한 IP Address 보다 사용할 호스트가 많이 존재하는 경우에 사용할 수 있으며, IP Address의 관리를 수작업으로 하지 않고 자동으로 부여할 수 있다는 것이다. 단점으로는 IP Address가 동시에 사용하는 시스템의 개수보다 적은 경우에는 동시에 모든 컴퓨터가 인터넷을 사용할 수 없다.

이 방식은 ADSL, Cable Modem, 위성 인터넷, ISDN, PPP (모뎀) 접속 등에서 응용하여 사용한다[4, 8, 10].

3. 시스템 구조

본 논문에서 설계 구현한 Gate-D(Gateway for Dynamic IP Address) 시스템은 크게 Gate-D Server, Gate-D IS (Internet 서버), Internet Client로 구성되어 있다. Gate-D Server는 유동 IP를 이용하여 인터넷 서버 기능을 제공할 수 있도록 시스템들을 서로 연결시키는 기능을 수행하며, 고유한 IP Address를 가지고 인터넷에 연결되어 있어야 한다. Internet Client는 인터넷의 일반 사용자들이다. Gate-D IS는 유동 IP Address를 사용하여 인터넷 서비스를 제공하고 자 하는 시스템으로 본 연구에서 개발된 Gate-D IS용 Client 프로그램을 인스톨하여 Gate-D Server에 연결을 설정한 후, 고유한 IP Address를 가진 인터넷 서버와 동일한 서비스를 제공할 수 있다. 구현 시스템은 자체 개발한 Gate-D Protocol을 사용하여 데이터를 교환하며, 메시지 기법을 사용하여 데이터의 안정성과 보안성을 높였다. Gate-D 시스템을 사용하게 되면 사용자는 Gate-D Server에 접근하지만 사용자가 원하는 정보는 Gate-D Server가 아닌 Gate-D Server와 연결되어 있는 유동 IP Address를 할당받은 서버에서 사용자가 원하는 정보를 가져오기 때문에 OSI(Open Systems Connection)의 7 계층 중 네트워크층(Network Layer)과 트랜스포트층(Transport Layer)의 역할을 가상적으로 수행한다고 할 수 있다. (그림 2)는 시스템 구조이며, 시스템의 전체 흐름은 다음과 같다.

- ① Gate-D IS에서 Gate-D Server의 Well Known Port로 접속한다. 접속이 설정이 되면 사용자 아이디와 패스워드의 Session 정보를 Gate-D Server에 제출한다
- ② Gate-D Server는 넘어온 세션 정보를 분석하여 Gate-D IS의 IP Address와 Port 번호를 Session Database에 저장한 후, 접속할 Gate-D Server의 IP Address와 Port 번호를 Gate-D IS로 리턴한다.
- ③ Gate-D IS는 자신과 Gate-D Server와의 연결을 설정한다.



(그림 2) 시스템 구조

③번까지의 과정이 끝나면, 인터넷 상의 모든 컴퓨터는 고유한 IP Address를 가지는 인터넷 서버의 접속과 동일한 방법으로 Gate-D Server에 등록된 유동 IP Address를 사용하는 인터넷 서버에 접근할 수 있다.

Gate-D Server와 Gate-D IS와의 연결 설정이 끝난 후, 인터넷 사용자가 Gate-D IS에 접근하는 과정은 다음과 같다.

- ④ 인터넷상의 클라이언트 컴퓨터에서 Gate-D IS의 도메인 이름을 가지고 접근한다.
- ⑤ Gate-D Server는 DNS(Domain Name Service) 서버와 Session Database의 정보를 이용하여 Gate-D IS에 연결된 Port와 연결시킨다.
- ⑥ 인터넷상의 클라이언트는 인터넷 서비스를 이용한다. 물론, 위의 과정은 사용자에게는 전혀 보이지 않는다.

기존의 nameIP, winIP, xDNS, eyeP.net, YiPost, Dhs.org, DynDNS.org 등의 서비스사가 제공하는 서비스는 단지 동적 DNS 서버 개념을 이용한 동적 Public IP Address를 가진 컴퓨터와의 동일한 연결을 보장한다. 일반적으로 DNS 서버는 서비스했던 DNS 정보를 로컬 DNS 서버의 캐쉬에 저장한다. 동일한 DNS 정보를 Internet Client가 요청한 경우에는 원본 DNS 서버의 DNS 정보가 변경이 되더라도 로컬 DNS 서버의 캐쉬에 저장되어 있는 DNS 정보를 제공한다. DNS 서버는 지정된 시간이 경과한 후 캐쉬의 정보를 업데이트 하기 때문이다 이러한 DNS 서버 운영 방식때문에 일부 Internet Client는 유동 IP 서버에 접근하지 못하는 문제점이 발생한다. 그러나, 본 논문에서 구현한 Gate-D 시스템은 동적으로 IP Address가 변경되더라도 유동 IP Address가 할

당된 서버에 대한 지속적인 연결을 보장한다. 개발도구는 SUN사의 JDK 1.3을 이용하여 어떤 플랫폼에서도 사용할 수 있도록 하였다.

4. 시스템의 구현 및 결과

4.1 Gate-D Server

Gate-D Server는 고정된 공인 IP Address가 할당된 서버에 작동중이며 동적 IP Address가 할당된 인터넷 서버에 설치된 클라이언트 모듈의 접근을 기다리고 있는 소켓 서버이다. 클라이언트가 접근할 때까지 idle 상태로 대기한다. 클라이언트에서 접근하게 되면 네트워크를 연결시켜서 인터넷 환경의 호스트에서 동적 IP Address 상의 내부 인터넷 서버로 접근할 수 있도록 한다. 즉, 사용자는 Gate-D Server를 통하여 동적 IP Address내의 인터넷(Telnet, FTP, Mail, HTTP 등) 서버에 접근할 수 있다. 사용자 측면에서 보면 Gate-D Server가 접근하는 서버처럼 보인다. Gate-D Server는 크게 Session 정보를 저장하고 있는 Session 정보 테이블과 도메인 이름을 IP Address로 변환하는 DNS 서버, Port Redirector가 있다.

하나의 Gate-D Server에서 다수의 유동 IP 인터넷 서버로의 접속을 지원하기 위하여 멀티 스레드 기법을 이용하여 개발하였다.

4.1.1 Session 정보 테이블

Session 정보 테이블은 인터넷 클라이언트와 Gate-D IS

(유동 IP를 사용하는 인터넷 서버)를 연결하기 위한 정보를 저장하고 있다. <표 1>은 Session 정보 테이블의 일부를 나타낸 것이다.

<표 1> Session 정보 테이블

User ID	서버 IP	Port Number	Domain Name	Protocol Name
htyang	210.110.85.213	4001	htyang.sunchon.ac.kr	HTTP
selab	210.110.85.202	4003	selab.sunchon.ac.kr	TELNET
...

User ID는 유동 IP Address가 할당된 인터넷 서버의 Gate-D Server와의 연결을 위한 세션 정보를 저장하고 있는 Primary Key라고 할 수 있다. 서버 IP와 Port Number는 서비스를 제공하고 있는 Gate-D Server의 IP Address와 Port Number를 의미한다. Domain Name은 인터넷 클라이언트가 Gate-D IS로 접근할 때 사용되는 도메인 이름을 의미하며, Protocol Name은 유동 IP Address가 할당된 서버에서 제공하는 서비스의 프로토콜을 저장함으로써 한 대의 인터넷 서버에서 다양한 인터넷(HTTP, Telnet, FTP, etc.) 서비스를 제공할 수 있도록 한다.

4.1.2 DNS(Domain Name Service) 서버

일반적인 DNS서버와 동일한 서비스를 제공한다. nameIP와 같은 기존의 서비스에서 사용하고 있는 동적 DNS 서버 뿐만 아니라 정적인 DNS 서버 역할도 수행한다.

4.1.3 Port Redirector

유동 IP Address가 할당된 인터넷 서버에 사용자가 도메인 이름을 가지고 접속하는 경우, Port Redirector는 접속 패킷에서 도메인 이름과 접근하고자 하는 프로토콜을 추출하여, 도메인 이름에 해당하는 유동 IP Address가 할당된 서버에 지정된 프로토콜에 해당되는 포트로 데이터를 Redirector하는 역할을 수행한다. 예를 들어 Client가 “http://www.htyang.co.kr”이라는 URL로 접근했을 때는 “http://htyang.sunchon.ac.kr:4001”로 “ftp://www.htyang.co.kr”일 때는 “ftp://htyang.sunchon.ac.kr:4002”로 변경해 준다. 일반적으로 Well Known Port로 구성된 URL인 경우 Session 정보 테이블에서 프로토콜을 가져오지 않는다.

<표 2>는 Gate-D 시스템을 위한 서버 프로그램의 설계 명세서이다.

<표 2> Gate-D Server의 설계 명세서

```
public class tnServer {
    public final static int inPortNo = 8944;
    public final static int outPortNo = 8955;
    public static void main() {
        // 클라이언트쪽을 위한 서버 소켓 생성;
        inServer = new ServerSocket(inPortNo);
        // 클라이언트 접속 대기;
        theConnection = inServer.accept();
```

```
// G/W 클라이언트가 접속 하면 msg_proc호출
msg_proc(theConnection);
}
static void msg_proc(Socket AinConnection) {
    // 외부 클라이언트를 위한 가상 텔넷 서버 소켓 생성
    outServer = new ServerSocket(outPortNo)
    // 외부 클라이언트의 사용자의 접속 대기
    theConnection = outServer.accept();
    // 외부 클라이언트가 접근하게 되면 link_proc 호출
    link_proc(AinConnection, theConnection);
}
static void link_proc
(Socket AinConnection, Socket AoutConnection)
{
    pin =
    new
    PrintStream(AinConnection.getOutputStream());
    // 서버와 클라이언트가 계속 통신하기 위한 쓰레드 생성
    ThreadReadWrite inner =
    new
    ThreadReadWrite(AinConnection, AoutConnection);
    ThreadReadWrite outer =
    new
    ThreadReadWrite(AoutConnection, AinConnection);
    // 쓰레드 시작
    inner.start();
    outer.start();
}
}
```

4.2 Gate-D Client

Gate-D Client는 Gate-D 시스템을 위한 클라이언트 프로그램으로 유동 IP Address가 할당된 인터넷 서버에 존재한다. Gate-D Client는 고정된 공인 IP Address상의 특정 컴퓨터에 대기하고 있는 Gate-D Server에 접근하여 네트워크 연결을 설정한 후 외부 클라이언트에서 동적 IP Address를 사용하는 특정 인터넷 서버로 접근할 수 있도록 데이터 패킷을 중계하는 기능을 수행한다. 유동 IP Address가 할당된 서버의 관점에서는 Gate-D Client가 클라이언트처럼 보인다.

<표 3>은 Gate-D Client의 설계 명세서이다.

<표 3> Gate-D Client 설계명세서

```
public class tnClient {
    public final static int inPortNo = 8944;
    public final static int outPortNo = 8955
    public static void main() {
        tnServerName = args[0]; // tnServer가 대기하는 서버
        inServerName = args[1]; // 실제로 연결해줄 서버
        // tnServer에 접속
        theSocket = new Socket(tnServerName, inPortNo);
        msg_proc(theSocket);
    }
    static void msg_proc(Socket AoutConnection) {
        // 내부 Telnet 서버에 접속
        theSocket = new Socket(inServerName, 23);
        link_proc(theSocket, AoutConnection);
    }
}
```

```

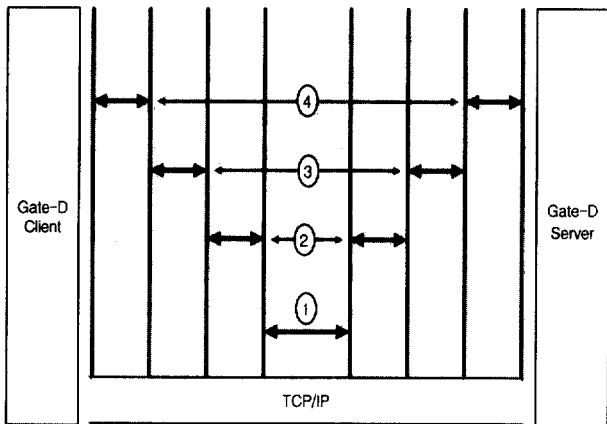
static void link_proc(Socket AinConnection, Socket
AoutConnection) {
    pin =
    new PrintStream(AinConnection.getOutputStream());
    //서버와 클라이언트가 계속 통신하기 위한 쓰레드 생성
    ThreadReadWrite inner =
    new ThreadReadWrite(AinConnection, AoutConnection);
    ThreadReadWrite outter =
    new ThreadReadWrite(AoutConnection, AinConnection);
    //쓰레드 시작
    inner.start();
    outter.start();
}
}
    
```

4.2.1 Link Monitor

Gate-D Client의 Link Monitor는 Gate-D Server와 유동 IP 서버와의 네트워크 설정에 문제가 발생한 경우 자동적으로 재연결한다. 즉, Gate-D Client가 있는 컴퓨터의 IP Address가 변경된 경우에는 사용자가 설정한 시간만큼 대기한 후 IP Address 재설정 작업을 수행하도록 한다. 네트워크에 문제가 발생한 경우 네트워크가 복구될 때까지 대기한다.

4.3 Gate-D Protocol

Gate-D 시스템에서 사용하는 전용 프로토콜로서 Gate-R 메시지의 스트림 형태의 교환 및 중계 처리를 위하여 고안되었으며 4개의 Layer로 구성되어 있다. (그림 3)은 Gate-D Protocol이다.



(그림 3) Gate-D Protocol

①의 첫 번째 계층에서는 실제로 데이터를 전송하는 계층으로서 스트림(Stream)을 사용한다. InputStream과 OutputStream에서 상속받은 DataInputStream과 DataOutputStream 메소드를 사용했다.

②의 두 번째 계층은 Message로 Encapsulation 또는 De-capsulation 하는 것으로서 전송하고자 하는 데이터를 메시지 형태로 변환한다. 메시지에 대한 내용은 다음 절에서 설명하고 있다.

③의 세 번째 계층은 메시지의 데이터 부분을 암호화하는 부분이다. 자바에서 암호화 기법을 적용하게 되면 Native Method를 사용하더라도 C나 어셈블리로 구현된 것보다도 성능이 많이 떨어지기 때문에 DES 기법을 사용하지 않고, 일반적인 해쉬 기법을 이용한 암호화 기법을 사용했다. 또한, 암호화된 데이터와 암호화되지 않은 데이터의 크기를 동일하게 만들어 무결성 검사를 할 수 있도록 하였다.

④의 네 번째 계층은 메시지의 데이터를 블록화하는 계층이다. 실제 전송되어지는 패킷을 TCP/IP의 MSS(Maximum Segment Size)의 크기에 비례하여 블록화 하였다.

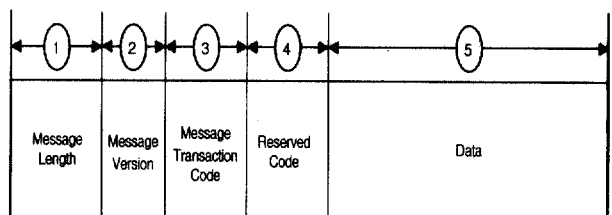
4.4 Gate-D Message

일반적으로 네트워크 상에서 데이터를 교환할 때 연속적인 스트림 형태로 전송하는 경우보다 메시지를 사용하여 보내는 것이 많은 장점이 있다.

첫 번째는 하나의 연결만이라도 여러 가지 종류의 메시지를 보낼 수 있다. 캡슐화를 하지 않을 경우 양쪽에서는 모든 메시지의 형태를 알고 있어야 한다. 즉, 서버측이나 클라이언트측 한 곳에서 업그레이드나 변경이 된 경우 양쪽 다 수정을 해 주어야 하는 번거로움이 있다.

두 번째는 특정 메시지에 생긴 에러가 에러 이후 전송되는 나머지 메시지 전체에 영향을 미치지 않는다는 것이다. 데이터가 연속적인 스트림에 오는 경우 중간에 데이터가 손상된 경우 처음부터 다시 전송을 받아야 하지만 메시지 형태로 데이터를 교환하는 경우 손상된 메시지가 중간에 있더라도 그 메시지를 다시 전송받거나 수정하면 문제가 없다는 것이다.

세 번째는 완전한 메시지만을 받는다는 것이다. 메시지를 신뢰할 수 있다면 어플리케이션 개발시 메시지 에러를 위한 예외 상황 루틴 처리를 하지 않아도 된다.



(그림 4) Gate-D Message9

(그림 4)는 본 논문에서 구현한 Gate-D에서 사용되는 메시지를 나타낸 것이다. 메시지는 최소 8Byte에서 1460Byte 까지이다. 헤더 부분은 8Byte로서 고정적이며 데이터 필드는 존재하지 않을 수 있다.

①번부터 ④번까지는 헤더 부분이며 ⑤번 부분이 실제 데이터가 있는 부분이다.

①번은 헤더 부분과 데이터 부분으로 구성된 메시지의 총 길이를 나타내며 길이는 3Byte이다.

②번은 메시지의 버전을 나타낸다. 시스템이 변경되더라도 하위 버전 시스템과의 호환성을 유지할 수 있으며, 길이는 2Byte이다.

③번은 메시지의 특성을 나타내는 것으로서 <표 4>에 정의되어 있다. 헤더를 이용하여 메시지의 특성을 알 수 있으며, 길이는 1Byte이다.

④번은 예약 공간으로서 추후 확장을 위한 예비 공간으로 길이는 2Byte이다.

⑤번은 실제 데이터 부분으로 최소 0Byte에서 1452Byte 까지 가능하다.

<표 4> Message Transaction Code

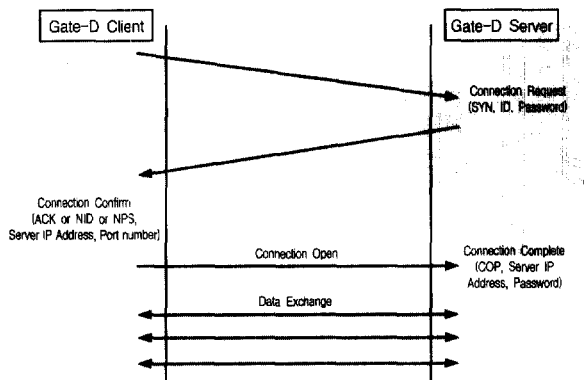
Physical	Logical	내 용
00000001	SYN	Connection 요구(ID, 패스워드 포함)
00000002	ACK	Connection 허락, Server IP/Port Number 전송
00000003	COP	Connection 설정
00000003	NID	ID없음
00000004	NPS	패스워드 틀림
00000005	DAT	데이터 전송
00000006	FIN	Connection 종료 요청

4.5 연결 설정 및 연결 종료

Gate-D 시스템은 TCP 기반으로 설계, 구현하였다. 연결 및 연결 종료는 TCP에서 사용되는 방법을 적용시켰으며, 데이터 전송의 초기 응답 시간을 단축시키기 위하여 Message Transaction Code의 상태코드에 몇 가지 기능을 추가하였다. 추가된 기능 및 기본 메시지 처리 코드는 <표 4>와 같다. 연결 설정과 연결 종료에는 TCP에서 사용하는 Three-way Handshaking 기법을 사용하였다.

4.5.1 연결 설정

Gate-D 시스템의 연결 설정은 (그림 5)와 같이 Gate-D Client에서 접속을 요구하는 "SYN" 이라는 메시지 코드와 User ID, Password를 Gate-D Server의 Well Known Port 로 전송하는 것으로 시작한다.

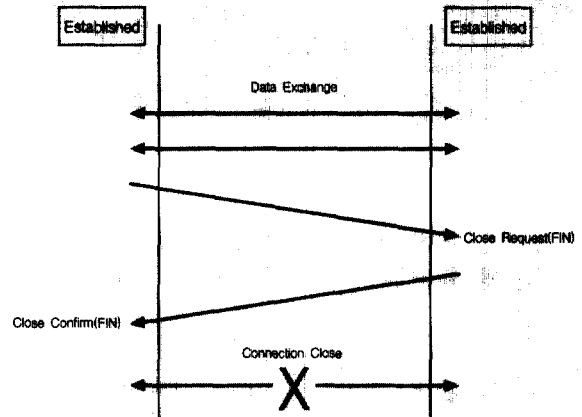


(그림 5) 연결 설정

접속 요구를 받은 Gate-D Server에서는 Session 정보 테이블에서 ID에 해당하는 해당 세션 정보를 찾아서 비교한다. 세션 정보가 일치하는 경우에는 "ACK" 메시지 코드와 Gate-D Client에서 접속할 Gate-D Server가 있는 서버의 IP Address 및 Port Number를 Gate-D Client로 넘겨준다. 일치하지 않는 경우에는 해당되는 에러 코드를 넘겨준다. 정상적인 접속 정보를 얻은 Gate-D Client는 해당 IP Address와 Port Number를 이용해서 Gate-D Server와 연결 설정을 완료한다.

4.5.2 연결 종료

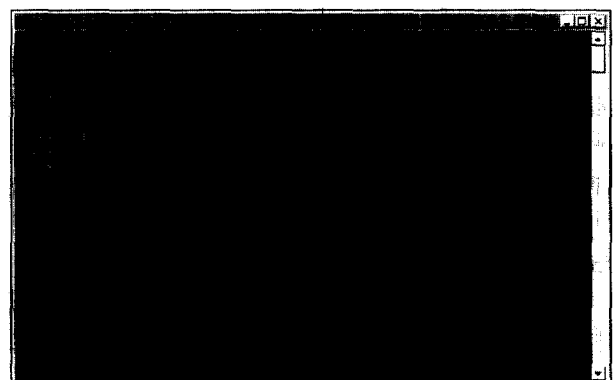
연결 종료 과정은 (그림 6)과 같이 연결 설정이 성립되어 있는 두 노드간에서 한쪽이 접속 종료 요구를 나타내는 "FIN" 메시지 코드를 전송하면, 그 종료 요구를 받은 반대쪽 노드에서 Close Confirm를 나타내는 "FIN"을 접속한 후 접속을 종료하는 과정을 거친다.



(그림 6) 연결 종료

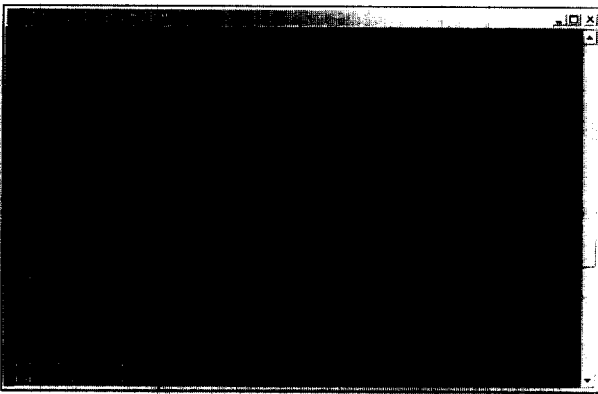
4.6 구현결과

본 연구에서 설계한 Gate-D 시스템의 적용을 위하여 Gate-D Server를 S대학에 할당된 고정된 공인 IP Address 210.110.xxx.xxx를 가진 시스템에 설치하였고, Gate-D Client는 ADSL이 설치된 S아파트의 컴퓨터에 설치하였다. S아파트

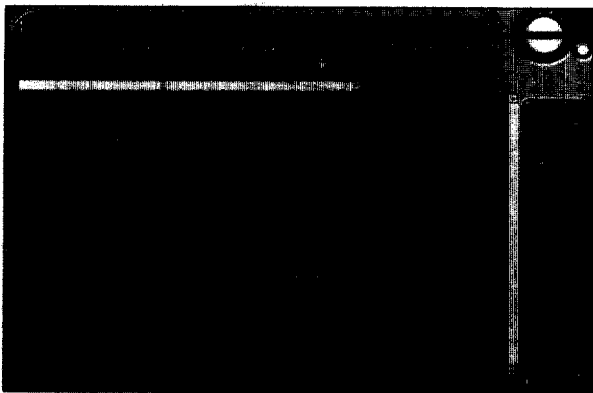


(그림 7) Gate-D 서버 작동 화면

에 설치된 컴퓨터는 운영체제로 리눅스를 사용하였으며, 제공하는 인터넷 서비스는 텔넷이다. (그림 7)은 Gate-D Server가 작동된 화면을 보여주며, (그림 8)은 Gate-D Client가 작동된 화면, (그림 9)는 일반 인터넷 사용자가 Gate-D Server를 통하여 S아파트에 설치된 동적 IP Address를 가진 Telnet 서버에 접속하고 있는 화면이다. (그림 9)는 해당 서버의 IP Address를 DNS서버를 통해서 구한 후 인터넷 서버와 인터넷 클라이언트를 직접 연결한 것이 아니라 Gate-D 시스템을 통하여 내부 서버와 연결하고 있음을 나타낸다.



(그림 8) Gate-D Client 작동 화면



(그림 9) 내부 서버 연결 화면

5. 결론 및 향후 연구 방향

기존의 NameIP, xDNS 등에서 동적으로 작동되는 DNS서버를 이용해서 접근할 수 있도록 하고 있으나, IP Address가 동적이기 때문에 IP Address가 변경된 경우 DNS 서버의 동기화가 지연되어 연결 보장을 할 수 없는 경우가 빈번하다. 그러나, 본 논문에서 설계 구현한 Gate-D 시스템은 IP Address가 동적으로 변경되더라도 동일한 접근을 보장해 주었다.

TCP 기반 응용 프로그램으로서 전송 데이터의 보안을 위한 암호화 기법 적용 및 HTTP, FTP와 같은 다양한 프로토

콜을 지원할 수 있도록 시스템을 확장하는 것이 앞으로의 과제이다.

참 고 문 헌

- [1] B. Moskowitz, D. Karrenberg, G. J. de Groot, and E. Lear, "Address Allocation for Private Internets," RFC 1918, February, 1996.
- [2] K. Egevang and P. Francis, "The IP Network Address Translator(NAT)," RFC 1631, May, 1994.
- [3] Merlin Hughes, Michael Shoffner, Derek Hamner, Umesh Bellur, Java Network Programming, 1999.
- [4] R. Droms, "Dynamic Host Configuration Protocol(DHCP)," RFC 1531, October, 1993.
- [5] W. Richard Stevens, TCP/IP Illustrated Volume-I The Protocol, pp.223-358, 1994.
- [6] http://searchsecurity.techtarget.com/sDefinition/0_sid14_gci212840,00.html
- [7] http://www.linuxdoc.org/HOWTO/Firewall-HO_WTO.html
- [8] <http://www.terms.co.kr>
- [9] <http://korea.internet.com>
- [10] 한국전산원, 사설 IP 주소를 이용한 인터넷 접속 사례 연구, 2000.



김 원 중

e-mail : kwj@sunchon.ac.kr

1987년 전남대학교 계산통계학과(이학사)

1989년 전남대학교 대학원 전산통계학과 (이학석사)

1991년 전남대학교 대학원 전산통계학과 (이학박사)

1999년~2000년 Iowa State University 교환교수

1992년~현재 순천대학교 정보통신공학부 부교수

관심분야 : 소프트웨어공학, 시스템 모델링, 객체지향 시스템, 인터넷 정보검색



양 현 택

e-mail : htyang@moinvalley.co.kr

1985년 전남대학교 계산통계학(이학사)

1999년 순천대학교 대학원 컴퓨터학과 (이학석사)

1999년~현재 순천대학교 대학원 컴퓨터과 (이학박사)

1994년~현재 (주)모인밸리 대표이사

관심분야 : 소프트웨어공학, 인터넷 정보검색, 의료정보시스템