

---

# 결함내성을 가진 도메인네임 서버의 구축 및 연동시험

최재원\*

Foundation Techniques and Cooperation Test of Fault-tolerant Domain  
Name Servers for Internet Name Resolution

Jae-Won Choi\*

---

이 논문은 2010학년도 경성대학교 학술연구비지원에 의하여 연구되었음.

---

## 요 약

DNS(Domain Name System)는 인터넷상의 호스트의 도메인주소를 IP주소로 변환하거나 IP주소를 도메인주소로 변환하는 이름해결 메카니즘을 총칭한다. 본 논문에서는 1차 DNS 서버가 오류로 인해 정지하더라도 2차 DNS 서버가 대신하여 서비스를 지속할 수 있도록 하는 결함내성을 갖는 DNS 시스템 구축에 관해 연구하였다.

## ABSTRACT

DNS(Domain Name System) is the Name Resolution Mechanism that makes conversion from a Domain Name of a computer on the Internet to an IP Address or the reverse conversion. In this paper we researched on the Foundation techniques of Fault-tolerant DNS Servers that the secondary DNS can take over and provide continuous services even though primary DNS stops due to some critical errors.

## 키워드

네임서버, 결함내성, 이중화

## Key word

DNS, Domain Name System, Fault Tolerance, Duplication

---

\* 종신회원 : 경성대학교 컴퓨터학부 정교수 (choejw@ks.ac.kr)

접수일자 : 2010. 08. 03

심사완료일자 : 2010. 09. 27

## I. 서 론

DNS(Domain Name System)는 네트워크에서 도메인이나 호스트 이름을 숫자로 된 IP 주소로 해석해 주는 TCP/IP 네트워크 서비스이다.[1] 인터넷 상에서 IP 주소 대신에 도메인 주소를 사용하는 것은 222.122.195.6과 같은 IP 주소보다는 www.naver.com과 같은 문자 주소를 사용하는 것이 편리하기 때문이다.

DNS 서버는 접근하고자 하는 호스트의 도메인 주소를 IP주소로 변환하거나 IP주소를 도메인주소로 변환하는 서비스를 제공하여 사용자로 하여금 인터넷상의 컴퓨터의 접근을 용이하게 한다. 또한 시스템 이진으로 호스트의 IP 주소의 변경이 발생하더라도 DNS 서버에 이의 IP 주소를 갱신하면 사용자는 동일한 도메인주소로 서비스를 계속해서 받을 수 있다.

DNS 서버는 도메인주소를 IP주소로 변환하는 서비스를 제공하는 서버로서 인터넷에서 최소한의 기본 서버이다. 그러므로 이의 신뢰성 및 무결성 향상을 위한 연구는 중요하다. 시스템의 신뢰성 향상을 위한 방법은 여러 가지가 있지만 오류발생을 사전에 완전히 배제하기란 실재론 거의 불가능하므로 일반적으로 오류가 발생할 때 이에 대응하는 결함내성(fault tolerance)의 방법으로 시스템의 신뢰성을 향상시킨다.[2]

본 논문에서는 1차 DNS 서버가 수행 중에 시스템 오류가 발생하여 서비스 제공이 어렵게 되면 2차 DNS 서버가 서비스를 유지하여 이용자가 시스템의 이상을 전혀 인식하지 못하도록 계속 서비스 받을 수 있게 하는 DNS 시스템의 결함내성에 대해 연구하였다. 구축한 1-2차 DNS 서버 간에는 주기적 갱신에 의해 동일 상태가 유지되고, 1차 DNS 서버가 오류로 인해 정지하더라도 2차 DNS 서버가 대신하여 서비스를 지속할 수 있도록 하였다.

본 연구에서는 결함내성을 갖는 DNS 구축을 위해 도메인(infonet.ac.kr)을 신규 등록하고 해당 도메인을 관장하는 1차 네임서버(121.174.8.52)와 2차 네임서버(121.174.8.53)를 실제 예를 통해 리눅스 기반의 1-2차 DNS 서버의 환경설정 및 구축절차를 상세히 기술하였다.

## II. DNS 시스템 개요

DNS는 인터넷에서 컴퓨터의 도메인주소와 IP주소 간의 주소변환을 담당하여 사용자의 접근을 용이하게 한다. 본 장에서는 DNS 시스템의 개요와 도메인 등록절차에 대해 연구하였다.

### 2.1 DNS 시스템

#### 1) DNS의 이해

DNS는 도메인이나 호스트 이름을 IP 주소로 변환해 주는 거대 규모의 분산 네이밍 시스템이다.[3] 도메인 이름(domain name)은 인터넷상에서 웹사이트나 원격서버에 접속하고자 할 때 IP 주소 대신에 사용되는 문자화된 인터넷 주소이다. TCP/IP 기반의 인터넷은 호스트 지정을 위해 숫자로 된 IP 주소를 필요로 하지만 문자로 된 도메인 이름이 사람들이 기억하기 더 쉬운 표기 수단이 된다.

도메인(Domain)은 infonet.ac.kr과 같이 호스트 컴퓨터들이 묶여져 있는 그룹을 의미하는 이름 공간(name space) 혹은 영역(name zone)을 의미하고, 도메인 이름(Domain Name)은 호스트명(hostname)과 도메인(domain)으로 구성된 도메인주소를 말한다. 그리고 네임서버(Name Server 혹은 DNS Server)란 그림 1과 같이 해당 도메인 내에서 호스트의 도메인 주소와 IP 주소의 변환을 책임지는 DNS 서버를 말한다.

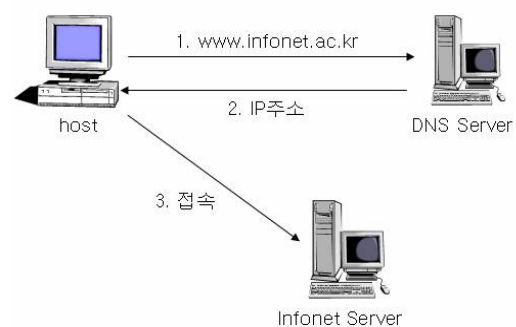


그림 1. 도메인주소의 IP주소 변환  
Fig 1. Conversion of Domain Name to IP Address

2) DNS의 동작

DNS는 계층구조의 도메인 트리로 이루어져 있다. 특정 컴퓨터로부터 이름해결 요구가 발생하면 우선 자신의 DNS 서버에 질의를 하고, 자사 내에서 해결이 안 될 때는 자동적으로 상위의 DNS 서버에 질의하는 구조가 확립되어 있다. DNS에 의한 이름해결은 그림 2와 같이 www.xxx.co.kr 웹서버를 접근하고자 하는 경우 사내의 DNS 서버가 모르면 상위 Root DNS 서버로 질의하여 하위 DNS에 대한 정보를 구하고, 이를 통해 반복적으로 하위 DNS 정보를 제공받아 최종적으로 xxx.co.kr DNS 서버로부터 www.xxx.co.kr 웹서버의 IP 주소를 얻게 된다.

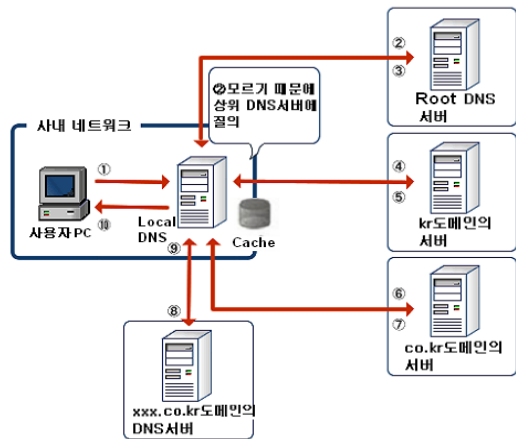


그림 2. 도메인주소의 IP주소 변환 절차  
Fig 2. Conversion from Domain Name to IP Address

2.2 도메인 등록

도메인 등록은 사용하고자 하는 도메인을 신청하고 해당 도메인을 관장하는 네임서버의 IP주소를 상위 네임서버에 등록하는 과정이다. 도메인 등록기관인 www.inames.co.kr에 접속하여 등록하고자 하는 도메인의 등록 유무를 확인한 후 해당 도메인을 관장하는 1차 네임서버와 2차 네임서버를 등록한다.

① 도메인 관리

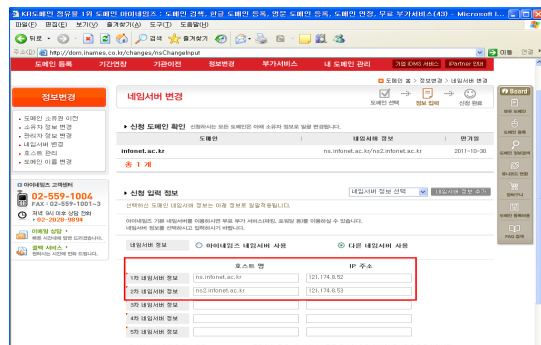
신규 도메인의 등록절차는 참고문헌 [4]에 자세히 기술되어 있으며, 본 논문에서는 도메인(infonet.ac.kr) 등록의 가정 하에 해당 도메인의 네임서버 입력을 위해선 www.inames.co.kr 사이트에 접속하여 [도메인관리] 메뉴를 선택한 후 [네임서버변경] 버튼을 누른다.

뉴를 선택한 후 [네임서버변경] 버튼을 누른다.



② 네임서버 변경

1차 네임서버의 이름(ns.infonet.ac.kr)과 IP (121.174.8.52) 그리고 2차 네임서버의 이름(ns2.infonet.ac.kr)과 IP(121.174.8.53)를 입력한 후 [네임서버정보변경] 버튼을 누른다.



III. 결함내성의 DNS 서버 구축

결함내성을 갖는 DNS 시스템 구축을 위해선 시스템을 1차 DNS와 2차 DNS로 이중화(duplication) 한다. 1차 DNS는 해당 도메인의 주소변환 서비스를 제공하는 메인 DNS이고, 2차 DNS는 메인 DNS의 이상을 대비하여 동작하는 보조 DNS이다. 본 장에서는 결함내성을 갖는 DNS 서버 구축을 위한 1-2차 DNS 프로그램의 설치, DNS의 환경설정, DNS 서버의 구동 및 동작시험에 관해 연구하였다.

### 3.1.1차 DNS 서버 구축

BIND(Berkeley Internet Name Domain)는 DNS 서버 구축을 위한 패키지이고, 이는 소스 파일, rpm 패키지, yum 명령의 3가지 설치방법이 있다.[4] 본 논문에서는 프로그램의 설치가 가장 용이한 YUM 명령을 이용하여 설치하였고, 1차 DNS 서버의 IP주소는 121.174.8.52 이다.

#### 3.1.1 DNS 프로그램 설치

YUM(Yellowdog Updater Modified)은 redhat 기반의 플랫폼에서 인터넷을 통한 편리한 설치와 업데이트를 가능케 해주는 프로그램으로 프로그램 의존성을 자동으로 확인하여 DNS 프로그램을 안전하게 설치, 삭제, 업데이트해 준다.

##### ① BIND 설치 여부 확인 및 제거

기 설치된 bind 프로그램이 존재하면 yum을 통해 쉽게 제거할 수 있다. 아래 명령을 실행 후 확인에 y만 입력하면 자동으로 제거된다.

# rpm -qa | grep bind

```

[root@ns1 ~]# rpm -qa | grep bind
rcbind-0.11.4-6.fc7
ypbind-1.19-9.fc7
bind-utils-9.4.0-6.fc7
bind-libs-9.4.0-6.fc7
[root@ns1 ~]#
    
```

# yum remove bind\*

```

[root@ns1 ~]# yum remove bind*
Loading "installonlyn" plugin
Setting up Remove Process
Resolving Dependencies
--> Running transaction check
--> Package bind-libs, 1386 31:9.4.0-6.fc7 set to be erased
--> Package bind-utils, 1386 31:9.4.0-6.fc7 set to be erased
Dependencies Resolved

Package Arch Version Repository Size
-----
Removing:
bind-libs 1386 31:9.4.0-6.fc7 installed 2.4 M
bind-utils 1386 31:9.4.0-6.fc7 installed 316 k

Transaction Summary
-----
Install 0 Package(s)
Update 0 Package(s)
Remove 2 Package(s)

Is this ok [y/N]:
    
```

##### ② BIND 설치

아래의 명령을 실행 후 확인에 y만 입력하면 자동으로 설치한다.

# yum install bind\*

```

[root@ns1 ~]# yum install bind*
Loaded plugins: refresh-packagekit
Setting up Install Process
Resolving Dependencies
--> Running transaction check
--> Package bind, 1386 32:9.5.2-1.fc10 set to be updated
--> Package bind-chron, 1386 32:9.5.2-1.fc10 set to be updated
--> Package bind-devel, 1386 32:9.5.2-1.fc10 set to be updated
--> Package bind-libs, 1386 32:9.5.2-1.fc10 set to be updated
--> Package bind-sdb, 1386 32:9.5.2-1.fc10 set to be updated
--> Package bind-to-tinydns, 1386 0:0.4.3-4.fc10 set to be updated
--> Package bind-utils, 1386 32:9.5.2-1.fc10 set to be updated
--> Package bindfs, 1386 0:1.0.2-1.fc10.1 set to be updated
--> Finished Dependency Resolution

Dependencies Resolved

Package Arch Version Repository Size
-----
Installing:
bind 1386 32:9.5.2-1.fc10 updates 3.3 M
bind-chron 1386 32:9.5.2-1.fc10 updates 59 k
bind-devel 1386 32:9.5.2-1.fc10 updates 426 k
bind-libs 1386 32:9.5.2-1.fc10 updates 981 k
bind-sdb 1386 32:9.5.2-1.fc10 updates 281 k
bind-to-tinydns 1386 0:4.3-4.fc10 updates 21 k
bind-utils 1386 32:9.5.2-1.fc10 updates 195 k
bindfs 1386 1.8.2-1.fc10.1 updates 27 k

Transaction Summary
-----
Install 8 Package(s)
Upgrade 0 Package(s)
Total download size: 3.3 M
Is this ok [y/N]:
    
```

#### 3.1.2 DNS 환경 설정

BIND 프로그램을 설치한 후 DNS 환경 설정을 위해 선 주변환경과 환경설정 파일을 설정해주어야 한다.

##### 1) 주변 환경 설정

###### ① /etc/host.conf

order 옵션은 주소해석을 위한 정보 참조순서를 나타낸다. 기본 설정은 먼저 /etc/hosts 파일을 참조한 후 DNS 서버에 질의하도록 한다.

###### ② /etc/hosts

IP주소와 호스트이름의 매핑이 수록된 파일이다. DNS 서버로의 질의에 앞서 참조하도록 하면 빠른 주소 해석에 도움이 된다.

###### ③ /etc/resolv.conf

해당 도메인(예:infont.ac.kr)을 관장하는 지역 네임 서버의 IP(예:121.174.8.52)가 수록된 파일이다. 네트워크 환경 [데스크탑→관리→네트워크] 설정에서 DNS를 지정하면 이 파일에 수록된다.

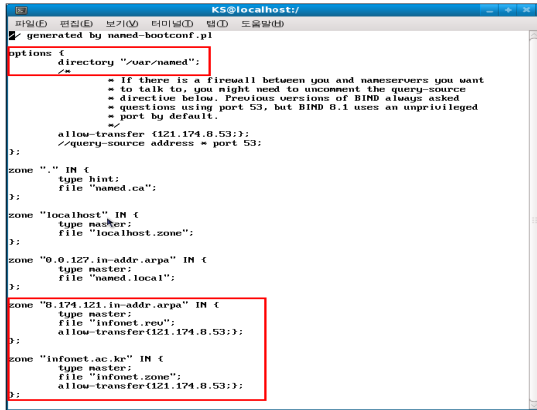
```

[root@ns1 ~]# cat /etc/host.conf
order hosts,bind
[root@ns1 ~]# cat /etc/hosts
# Do not remove the following line, or various programs
# that require network functionality will fail.
127.0.0.1 localhost.localdomain localhost
121.174.8.52 ns.infont.ac.kr
[root@ns1 ~]# cat /etc/resolv.conf
search infont.ac.kr
nameserver 121.174.8.52
[root@ns1 ~]#
    
```

2) 기본 환경 설정

(1) /etc/named.conf

BIND 시동 시 참조되는 핵심 설정 파일로 운영할 도메인에 대한 환경정보를 담고 있다.



① directory "/var/named";

DNS 환경설정 파일들이 위치하는 경로를 지정한다. 이 디렉토리에 named.ca, named.local, named.zone, infonet.zone, infonet.rev 파일들이 존재한다. 보안상 chroot를 적용하면 /var/name/ chroot/var/named 디렉토리에 환경설정 파일들이 위치하게 된다.

② zone "infonet.ac.kr" IN {

```
type master;
file "infonet.zone";
allow-transfer { 121.174.8.53; };
```

infonet.ac.kr 도메인 내에 있는 모든 컴퓨터들의 IP가 infonet.zone 파일에 정의되고, 1차 네임서버는 type을 master로 하고 2차 네임서버는 type을 slave로 한다.

allow-transfer { }는 zone-transfer를 허용할 IP를 지정해 준다. 보통 2차 네임서버를 지정하여 해당 zone 파일을 지정한 2차 네임서버가 복사해 갈 수 있도록 허용해 주는 옵션이다.

③ zone "180.110.210.in-addr.arpa" IN {

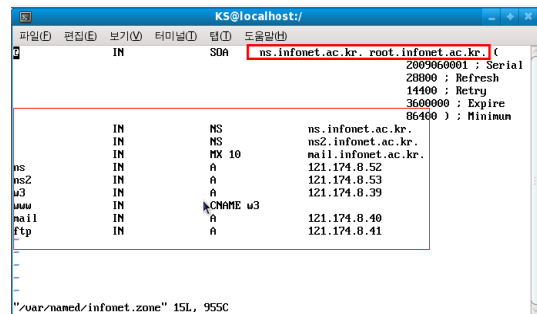
```
type master;
file "infonet.rev";
allow-transfer { 121.174.8.53; };
```

};

망주소가 210.110.180인 네트워크에 속해있는 컴퓨터들의 IP주소를 도메인주소로 변환하기 위한 정보가 infonet.rev 파일에 수록된다. 네트워크 주소부분은 거꾸로(예:121.174.8 → 8.174.121) 적어 주어야 한다. allow-transfer는 전과 같다.

(2) /var/named/infonet.zone

infonet.ac.kr 도메인 내에 있는 컴퓨터들의 도메인주소에 대한 IP주소를 저장하는 파일이다.



① TTL (Time To Live)

zone 설정 파일의 맨 첫줄에 위치하는 값으로 다른 서버에서 자신의 정보를 가져갔을 때 그쪽 서버의 캐시에 그 정보가 얼마나 오랫동안 유효한 지(캐시 활성화 시간)를 지정한다. 만약 명시 하지 않으면 Minimum 값을 참조한다.

② @ IN SOA ns.infonet.ac.kr. root.infonet.ac.kr.

- @ : /etc/named.conf 파일의 zone 영역에 설정된 도메인 infonet.ac.kr을 가리킨다.
- ns.infonet.ac.kr. : 네임서버를 의미한다.
- root.infonet.ac.kr. : 도메인에 문제 발생시 이를 알려줄 관리자의 메일주소(root@infonet.ac.kr)로 사용된다.

③ ( ) 괄호로 둘러싸인 부분

Serial, Refresh, Retry, Expire, Minimum은 2차 네임서버의 제어를 위한 정보로 이는 2차 네임서버의 환경 설정에서 자세히 설명하였다.

④ IN NS ns.infonet.ac.kr.

NS(Name Server)는 해당 도메인을 관장하는 네임서버를 가리킨다.

⑤ IN MX 10 mail.infonet.ac.kr.

MX(Mail eXchange)는 해당 도메인의 메일서버를 가리킨다. 번호가 낮을수록 우선순위가 높으며, 메일서버가 여러 개인 경우에는 우선순위가 높은 메일서버에 먼저 전달되고 문제가 발생할 경우 차순위 메일서버에게 전달된다.

⑥ 호스트명 IN A IP주소

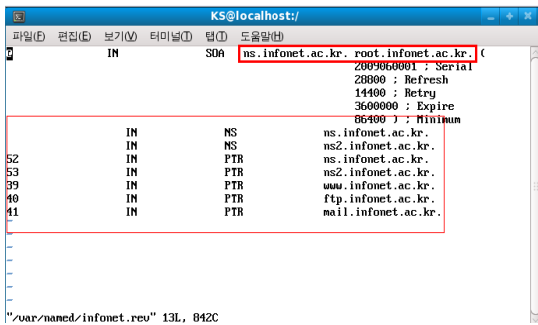
A(Address)는 해당 호스트의 IP 주소를 지정한다.

⑦ www IN CNAME w3

CNAME은 도메인에 대한 또 다른 이름이 가능할 때 사용한다. w3.infonet.ac.kr 대신 www. infonet.ac.kr을 사용할 수 있음을 의미한다.

(3) /var/named/infonet.rev

infonet.ac.kr 도메인 내에 있는 컴퓨터들의 IP주소에 대한 도메인주소를 저장하는 파일이다.



① PTR

PTR(PoinTeR)는 IP 주소에 대응하는 도메인 이름을 매핑시켜 준다.

② 도메인명 끝에 ‘.’ 추가

infonet.zone과 infonet.rev 파일 설정 시 도메인명을 표시할 때 반드시 끝부분에 ‘.’을 추가해 주어야 한다. 예를 들어, “mail.infonet.ac.kr.”에서 끝의 ‘.’을 추가해 주지 않으면 DNS는 “mail. infonet.ac.kr.infonet.ac.kr” 로 인식하게 된다.

(4) /var/named/named.ca, localhost.zone, named.local

name.ca 파일은 루트 네임서버의 정보가 들어 있는 데이터베이스 파일이다. 만약 이 파일이 없으면 “# dig . ns

> /var/named/named.ca”를 입력하면 생성된다. localhost.zone과 named.local 파일은 루프백 시험용 파일로서 기본 파일을 수정 없이 그대로 사용하면 된다.

(5) 보안을 위한 기본 디렉토리 변경 및 퍼미션 설정

named.conf 파일을 보면 infonet.zone과 infonet.rev 파일은 /var/named에 위치해야 하지만 보안상의 이유로 실제파일은 /var/named/chroot/var/named에 두고 /var/named에는 링크파일만을 둔다. 이를 위해 다음과 같이 심볼릭 링크를 생성하고 퍼미션을 설정해 준다.

실제 파일	심볼릭링크 파일
/var/named/chroot/var/named/infonet.zone	/var/named/infonet.zone
/var/named/chroot/var/named/infonet.rev	/var/named/infonet.rev
/var/named/chroot/etc/named.conf	/etc/named.conf

# ln -s [실제파일경로] [링크의위치]



3.1.3 1차 DNS 서버 구동 및 시험

DNS 환경설정이 끝나면 1차 DNS 서버를 구동하여 DNS 서버의 정상 동작여부를 확인한다.

1) DNS 서버 구동

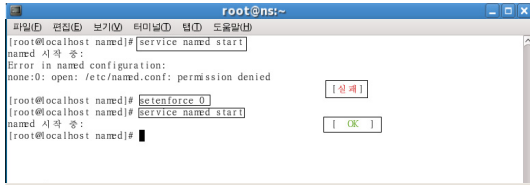
1차 DNS의 정상 동작 여부를 확인하기 위해 resolv.conf 파일에 자신의 IP를 지역 네임서버의 IP로 지정한다.



DNS 서버의 구동은 아래의 명령을 이용하여 구동한다. 방화벽의 활성화로 서버 구동이 실패하는 경우에는

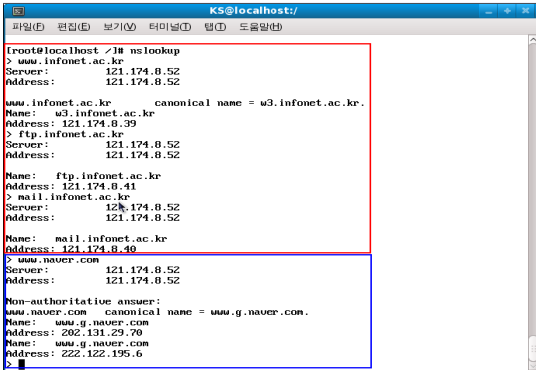
“#setenforce 0”를 입력하여 방화벽을 해제 한 후 다시 서버를 구동한다.

- ① 서버구동: #/etc/rc.d/init.d/named start
- ② 서버정지: #/etc/rc.d/init.d/named stop
- ③ 서버재구동: #/etc/rc.d/init.d/named restart



## 2) DNS 동작 시험

nslookup 명령은 구축한 DNS 서버의 zone 파일을 참조하여 도메인주소에 대응하는 IP주소를 제공할 뿐만 아니라 외부 도메인주소에 대해서도 IP주소 변환을 함으로 1차 DNS 서버 구축이 정상적으로 이루어진 것을 확인할 수 있다.



## 3.2.2차 DNS 서버 구축

일반적으로 시스템의 결합내성을 위해선 시스템의 중요 요소를 중복시켜 하나의 요소에 오류가 발생하더라도 중복된 다른 요소가 이를 대체하여 서비스를 계속 유지시키는 이중화 방식으로 구현한다.[5] 이중화(duplication) 방식은 어느 한 쪽만이 동작하다가 장애 발생시 다른 한 쪽이 동작을 이어가는 방식이다.

결합내성을 갖는 DNS 시스템을 위해선 시스템을 메인(primary, 1차) DNS와 보조(secondary, 2차) DNS로 이중화하고, 1차 DNS는 해당 도메인의 주소변환 서비스를 제공하는 메인 DNS 서버이고, 2차 DNS는 메인 DNS의 이상을 대비하여 동작하는 보조 DNS 서버이다. 이를 위

해 2차 DNS 서버는 해당 도메인의 주소변환을 위한 각종 정보를 1차 DNS 서버로부터 주기적으로 받아 1차 DNS 서버와 동일한 상태를 유지시킨다. 그리고 1차 DNS 서버가 수행 중에 시스템 오류가 발생하여 서비스 제공이 어렵게 되면 2차 DNS 서버가 서비스를 유지하여 이용자는 시스템의 이상을 전혀 인식하지 못한 상태에서 계속 서비스를 받을 수 있게 된다.

### 3.2.1 DNS 프로그램 설치

2차 DNS 서버의 프로그램 설치는 3.1.1절의 1차 DNS 서버의 프로그램 설치와 동일한 방식으로 YUM 명령을 이용하여 설치할 수 있으므로 이의 과정은 생략하기로 한다. 1차 DNS 서버의 IP주소는 121.174.8.52 이고, 2차 DNS 서버의 IP주소는 121.174.8.53 로 하였다.

### 3.2.2 DNS 환경 설정

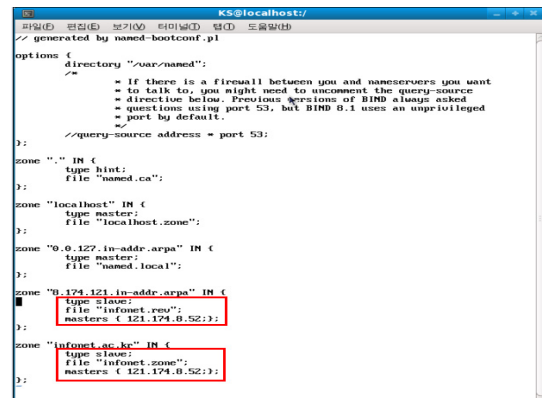
2차 DNS 서버의 환경 설정은 1차 DNS 서버의 zone 파일들을 복사하여 저장하면 되므로 2차 DNS 구축을 위해 추가된 환경 설정 부분만을 중점적으로 다루었다.

#### 1) 환경 파일 설정

##### (1) /etc/named.conf

1차 DNS 서버의 named.conf 파일에서는 type이 master 였는데 2차 DNS 서버의 named.conf 파일에서는 type을 slave로 설정해주고, “masters { IP }”를 1차 DNS 서버의 IP로 설정해 줌으로써 2차 DNS 서버가 1차 DNS 서버로부터 해당 도메인의 주소변환을 위한 정보를 주기적으로 받아올 수 있도록 한다.

즉, infonet.zone과 infonet.rev 파일을 주기적으로 복사해 줌으로써 1-2차 DNS 간에 동일 상태를 유지한다.

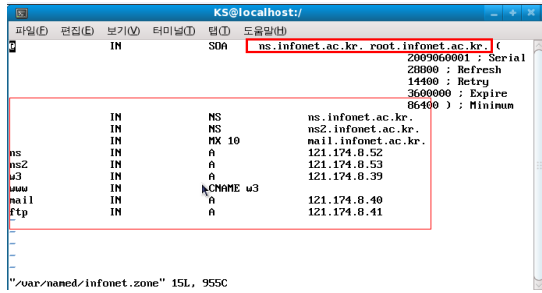




(2) /var/named/infonet.zone

scp 명령을 이용하여 1차 DNS의 zone 파일을 2차 DNS로 복사하여 초기 환경을 구축한다.

```
# scp 121.174.8.52:/var/named/chroot/var/named/infonet.zone
# vi /var/named/infonet.zone
```



(3) /var/named/infonet.rev

scp 명령을 이용하여 1차 DNS의 rev 파일을 2차 DNS로 복사하여 초기 환경을 구축한다.

```
# scp 121.174.8.52:/var/named/chroot/var/named/infonet.rev
# vi /var/named/infonet.rev
```

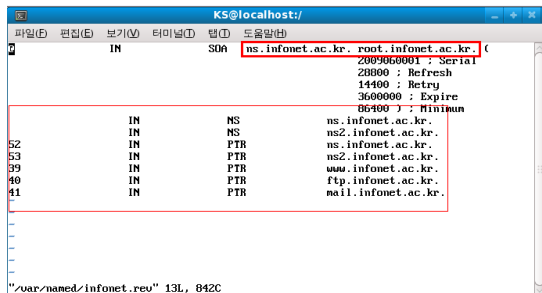


표 1. 1-2차 DNS 연동을 위한 환경변수의 기능  
Table 1. Function of Environment Variables for DNS Cooperation

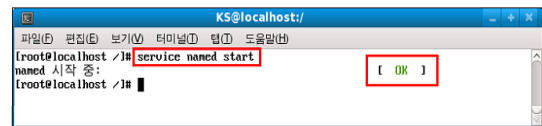
환경변수	설명
TTL (TIME TO LIVE) (\$TTL 3600)	zone 파일의 맨 처음에 위치하는 값으로 다른 서버에게 자신의 정보를 가져왔을 때 그쪽 서버의 캐시에 그 정보가 얼마나 오랫동안 머물 것인지에서 활성화 시간을 결정한다. 만약 명시 하지 않으면 Minimum 값 을 참조한다. (1시간)
Serial (2009043000)	Serial은 zone 파일이 최종 갱신된 날짜를 나타낸다. 2차 네임서버는 자신의 serial 값이 1차 네임서버의 serial 보다 작으면 새로 갱신된 것으로 보고 1차 네임서버의 zone 파일을 재전송 받는다. (2009. 04. 30. 00) 보물 YYYYMMDDHH의 형식으로 입력한다. 만약 zone 파일의 정보가 변경이 되었다면 꼭 증가시켜 줘야 한다. 그래야 2차 네임서버에서 변경 여부를 확인할 수 있다.
Refresh ( 28800 )	Refresh는 2차 네임서버가 1차 네임서버의 zone 파일 변경 여부를 검사 하는 업데이트 주기이다. (8시간)
Retry ( 14400 )	Retry는 2차 네임서버에서 1차 네임서버로 연결이 되지 않을 때 접속을 재시도 하는 주기이다. refresh 주기보다 작아야 한다. (4시간)
Expire ( 3600000 )	Expire로 지정한 시간동안 1차 네임서버에 연결하지 못할 경우 현재 가지고 있는 정보가 유효하지 않다고 판단하고 주소변환 서비스 요청에 응답하지 않는다. (1000시간 즉 41일)
Minimum ( 86400 )	Minimum은 다른 네임서버가 자신의 zone 파일에 저장된 자료를 가져왔을 경우 그 자료에 대한 유효기간을 나타낸다. 이 값은 zone 파일 처음에 표시된 TTL 값을 대체할 수 있으며, TTL 값이 설정되어 있으면 이 설정은 유효하지 않다. 캐시된 도메인 정보의 유효시간으로 TTL과 동일한 기능을 한다. (24시간)

2) 환경변수의 기능

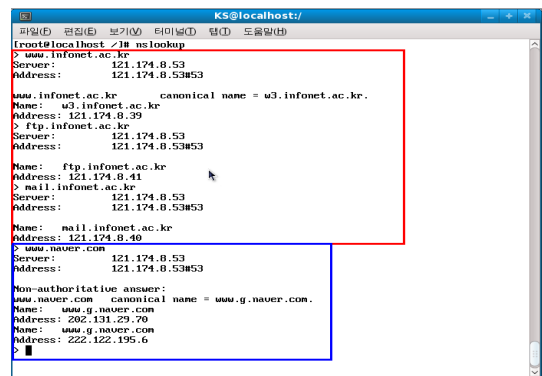
zone과 rev 환경설정 파일의 Serial, Refresh, Retry, Expire, Minimum은 1-2차 네임서버가 주기적 갱신을 통해 동일한 상태를 유지시키기 위한 환경변수들이고, 이에 대한 설명은 표 1에 나타내었다.[3] 환경변수의 기본 단위는 ‘초’ 이다

3.2.3 2차 DNS 서버 구동 및 시험

2차 DNS의 정상 동작 여부를 확인하기 위해 resolv.conf 파일에 자신의 IP를 지역 네임서버의 IP로 지정한 후, 2차 DNS 서버를 구동하여 정상 동작여부를 확인한다.



nslookup 명령을 통해 2차 DNS 서버 구축이 정상적으로 이루어진 것을 확인할 수 있다.



IV. DNS 연동 및 결함내성 시험

이중화의 경우에는 어떻게 양측의 데이터를 동일하게 유지시킬 것인지, 또 장애시 얼마나 빨리 계열을 변경하여 서비스의 연속성을 보장할 것인지가 중요하다. 본 장에서는 구축한 1-2차 DNS 서버 간 연동으로 주기적 갱신에 의해 동일한 상태가 유지되는지, 1차 DNS 서버가



오류로 인해 정지하더라도 2차 DNS 서버가 대신하여 서비스를 지속할 수 있는지를 시험하였다.

#### 4.1 DNS 연동 시험

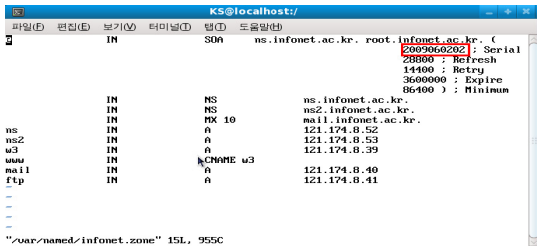
2차 DNS 서버는 해당 도메인의 주소변환을 위한 각종 정보를 1차 DNS 서버로부터 주기적으로 받아 1차 DNS 서버와 동일한 상태를 유지한다. 2차 네임서버는 환경변수 Refresh의 업데이트 주기를 따라 주기적으로 1차 네임서버의 zone 파일과 rev 파일을 접근하여 Serial 번호가 증가되었으면 주소정보가 갱신된 것으로 보고 1차 네임서버의 zone 파일과 rev 파일을 2차 네임서버로 복사해 온다.

##### 1) 1차 DNS의 Serial 번호 증가

1, 2차 DNS 서버를 재 시작시킨 후, 1차 DNS의 infonet.zone 파일과 infonet.rev 파일의 Serial 번호를 증가시킨다.

```
# /etc/rc.d/init.d/named restart
```

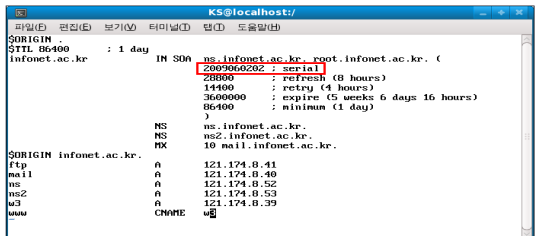
```
# vi /var/named/infonet.zone
```



##### 2) 2차 DNS의 갱신 확인

2차 DNS는 Refresh 주기가 경과되면 1차 DNS의 Serial 번호 증가를 확인하고 1차 DNS의 zone 파일과 rev 파일을 2차 DNS로 복사해와 갱신함으로 1, 2차 DNS 서버 간의 연동이 원활하게 일어나는 것을 확인할 수 있었다.

```
# vi /var/named/infonet.zone
```

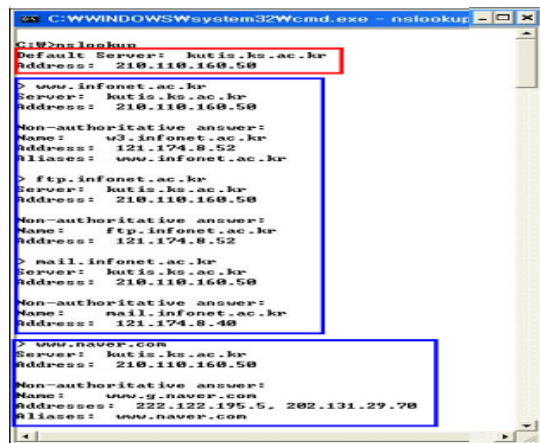
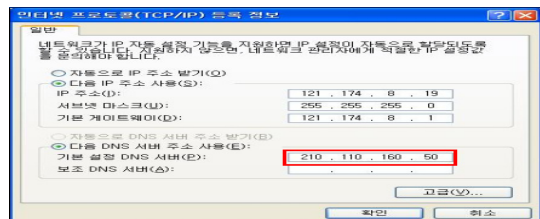


#### 4.2 DNS 결합내성 시험

1차 DNS 서버에 시스템 오류가 발생하여 서비스 제공이 어렵게 되면 2차 DNS 서버가 서비스를 유지하여 이용자는 시스템의 이상을 전혀 인식하지 못한 상태에서 계속 서비스를 받을 수 있는지를 시험하였다.

##### 1) 도메인 외부에서 접근 시 주소변환

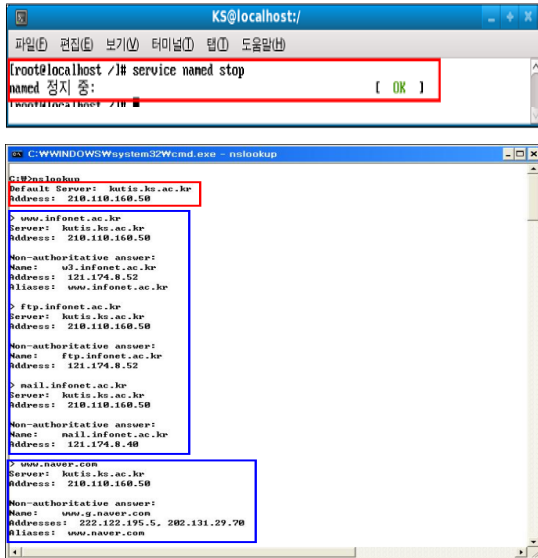
도메인 외부의 임의의 클라이언트 컴퓨터에서 nslookup 명령으로 도메인 내 각종 서버의 도메인 이름을 입력하면 DNS 환경설정 시에 지정한 서버의 IP주소로 변환이 정상적으로 이루어짐을 확인할 수 있었다. 외부 동작 시험을 위해 경성대학교(ks.ac.kr 도메인)의 DNS서버(210.110.160. 50)를 기본 DNS로 설정하였고, 외부 DNS 서버(210.110.160.50)를 통하여 실제로 inames 사이트에 등록된 1,2차 DNS 서버로부터 DNS 정보를 가져오는 것을 확인할 수 있었다.



##### 2) 도메인 외부에서 접근 시 결합내성

1차 DNS 서버(121.174.8.52)를 강제로 정지시킨 후, 도메인 외부의 임의의 클라이언트 컴퓨터에서 nslookup 명령으로 도메인 내 각종 서버의 도메인 이름을 입력하면, 외부 DNS 서버(210. 110.160.50)를 통하여 실제로

inames 사이트에 등록된 2차 DNS 서버(121.174.8.53)로부터 DNS 정보를 가져오는 것을 확인할 수 있었다.



### V. 결론

본 논문에서는 도메인주소와 IP주소의 변환 서비스를 제공하는 DNS 서버의 구축과 이의 신뢰성 및 무결성 향상을 위한 연구를 하였다. 1차 DNS 서버가 시스템 오류로 서비스 제공이 어렵게 되면 2차 DNS 서버가 서비스를 유지하여 이용자는 시스템 이상을 전혀 인식하지 못한 상태에서 계속 서비스를 받을 수 있도록 하는 DNS 시스템의 결합내성에 관해 연구하였다.

결합내성을 갖는 DNS 시스템 구축을 위해 평상시에는 1차 메인 DNS가 해당 도메인의 주소변환 서비스를 제공하고, 2차 보조 DNS는 메인 DNS의 이상 시에 동작하도록 DNS 서버를 이중화(duplication)하였다. 이를 위해 1-2차 DNS의 등록, YUM을 이용한 DNS 프로그램의 설치, DNS 동작 시 참조되는 named.conf 파일과 해당 도메인 내 컴퓨터들의 도메인주소와 IP주소를 가지는 zone 파일과 rev 파일 등 1-2차 DNS의 환경설정과 주변 환경 설정에 관해 연구하였다.

구축한 1-2차 DNS의 연동시험을 통해 2차 DNS는 Refresh 주기가 경과되면 1차 DNS의 Serial 번호 증가를 확인하고 1차 DNS의 zone 파일과 rev 파일을 2차 DNS로

복사해와 갱신함으로써 1,2차 DNS 서버 간의 연동이 원활하게 일어나는 것을 확인할 수 있었다. 그리고 결합내성 시험을 통해 1차 DNS 서버가 오류로 인해 정지하더라도 2차 DNS 서버가 대신하여 서비스를 계속 유지해 나가는 것을 확인함으로써 결합내성을 갖는 DNS 시스템 구축이 이루어졌음을 검증하였다.

본 연구에서는 실제 예를 통해 도메인 등록절차와 결합내성을 갖는 DNS 구축을 위한 1-2 DNS 서버 구축 절차를 상세히 기술함으로써 DNS를 처음 접하는 사용자나 관리자에게 서버 구축과 활용에 많은 도움이 될 것이다.

### 참고문헌

- [1] 정진욱, 김현철, “TCP/IP와 인터넷”, pp.250-296, 생능출판사, 2004.
- [2] Behrouz A. Forouzan, “TCP/IP Protocol Suite”, pp.247-270, McGraw-Hill.
- [3] 서자룡, “페도라 리눅스 그대로 따라하기”, pp.533-560, 헤지원, 2008.
- [4] 최재원, “인터넷 이름해결 메카니즘 도메인네임서버 구축에 관한 연구”, pp.43-53, 경성대학교 공학기술연구지 제15권, 2008년 12월.
- [5] Andrew S. Tanenbaum, “Computer Network”, pp.665-673, Prentice Hall

### 저자소개



최재원(Jae-Won Choi)

1988년 2월 고려대학교  
컴퓨터공학과 (공학사)  
1990년 8월 미시간주립대학교  
컴퓨터공학과 (공학석사)  
1995년 8월 건국대학교 전자공학과 (공학박사)  
1990년 10월~1997년 8월 삼성전자 정보통신연구소  
선임연구원  
1997년 9월~2010년 현재 경성대학교 컴퓨터학부  
정교수  
※관심분야: 정보통신, 이동통신, 인터넷응용, 운영체제