안녕 하세요. 윈도우 2000 DNS 에 대한 자료는 NTFAQ 회원이신 송원석님이 제공해 주신 자료입니다.

자료제공자 : 송원석 (blue@secure.pe.kr / http://www.secure.pe.kr)

제목 : 윈도우 2000 에서 DNS 개요 및 설정 작업 방법

목차 :

1. DNS 의 이해
2. DNS 의 구조 및 Process
3. DNS 설치 및 구성
4. DNS Server 관리
5. DNS 문제해결

등으로 계속 추가 하겠습니다.

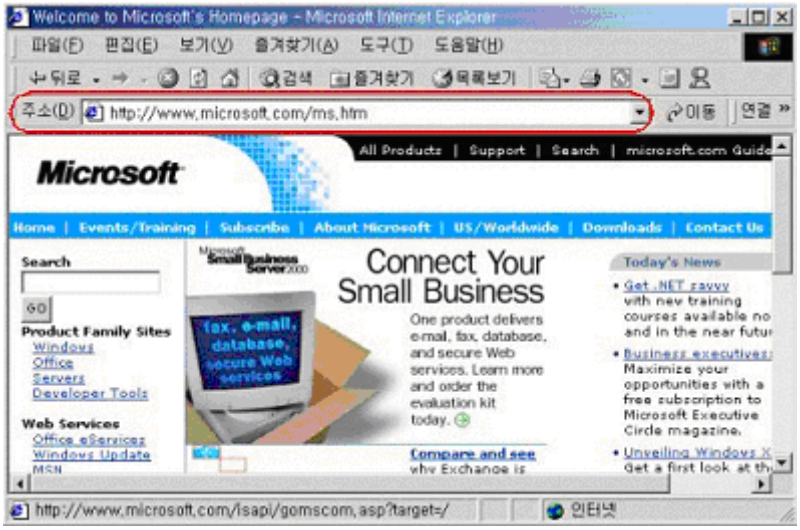
오늘은 첫번째인 윈도우 2000에서 DNS 이해에 대해서 제공 하겠습니다.

1.윈도우 2000에서 DNS 의 이해

굳이 기업네트워크의 전문가가 아니더라도 여러분들이 컴퓨터 앞에 앉아 있다면, 늘 DNS의 도움을 받고 있다. DNS는 마이크로소프트의 전유물은 결코 아니다. 수많은 사람들의 노력의 결과이며, 당분간 DNS의 영향력은 점점 더 증가하게 될 것이라고 감히 말하고 싶다. 마이크로소프트는 Windows 제품군으로 기업용 네트워크 운영체제인 NT에 이어서 Windows 2000이 나오면서 DNS의 쓰임새가 대폭 확장되었고, 이제 Windows2000을 제대로 이해하기 위해서 DNS를 알아야 하는 것은 필수적인 조건이 되었다.

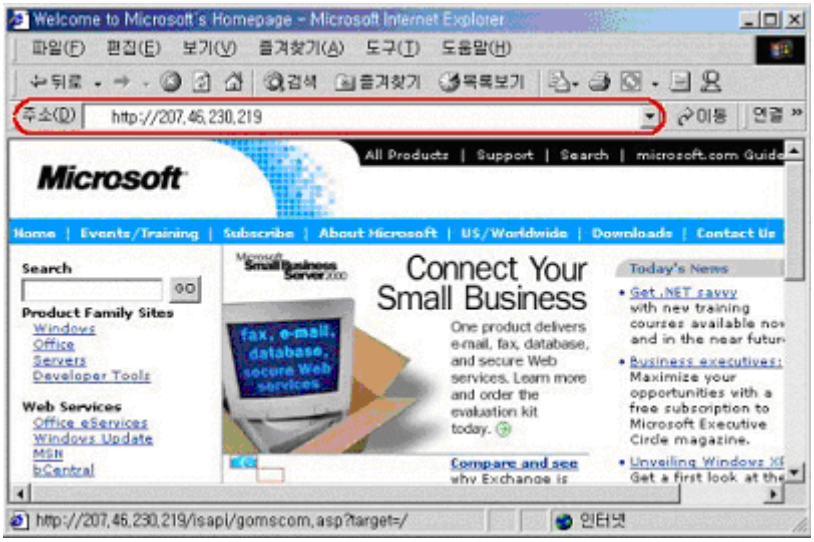
필자는 Windows2000의 DNS를 위주로 하여 전반적인 DNS의 기능과, 개념 등에 대해서 설명하고자 한다. 어려운 개념은 아니지만, 필자의 경험상으로 봤을 때 많은 사람들이 가장 어려워하는 내용이기도 하다. 어느 정도 지식이 있는 사람이 보기에는 "나를 뭘로 알고...??"라는 생각이 들 정도로 아주 기초적인 내용부터 접근해 보도록 할 것이다. 차근차근 익히다 보면 이 내용이 끝났을 때 여러분은 많은 내용을 얻을 수가 있을 것이다.

일단 단적으로 이해를 돕기 위해 DNS의 한가지 주된 용도에 대해서 알아보자. 우리는 인터넷에 접근하기 위하여 인터넷 익스플로러, 넷스케이프 같은 웹 브라우저를 이용한다. 아래의 그림을 보자.



[그림1. www.microsoft.com homepage]

간선으로 구분해 놓은 곳을 보면 찾아가길 원하는 서버의 주소를 입력하는 창을 볼 수 있다. 지금 예제에서는 www.microsoft.com 이라는 웹서버를 찾아서 홈페이지를 보고자 한다. www.microsoft.com 이라는 것은 결국 Microsoft라는 회사의 홈페이지를 담고 있는 컴퓨터의 이름인 것이다. 보통 유저라면 "흠.. 당연한거 아닌가?" 라고 생각할 수도 있다. 그렇다면 아래의 그림도 살펴보자.



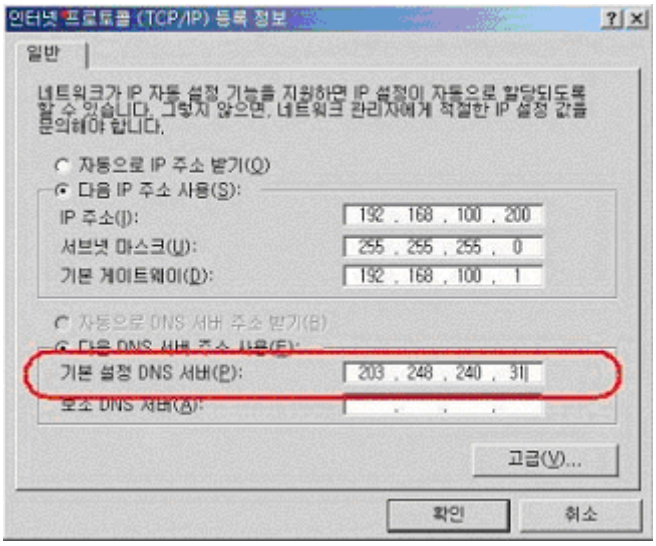
[그림2. 207.46.230.219]

출력되는 화면에는 차이가 없지만, 주소란은 분명히 차이가 보인다. 그림1.에서의 주소와 비교해 보면, 그림1.에서는 우리에게 친숙한 문자가 쓰인 반면에, 그림2.에서는 외우기엔 조금 어려워 보이는 몇자리의 숫자가 적혀 있다. 바로 서버의 주소를 가리키는 IP Address 이다. IP Address라는 것에 대해서 용어가 생소하다면, DNS는 나중 일이다. 먼저 IP Address라는 것에 대한 이해를 구하고 DNS를 배우도록 하는 것이 바람직하다.

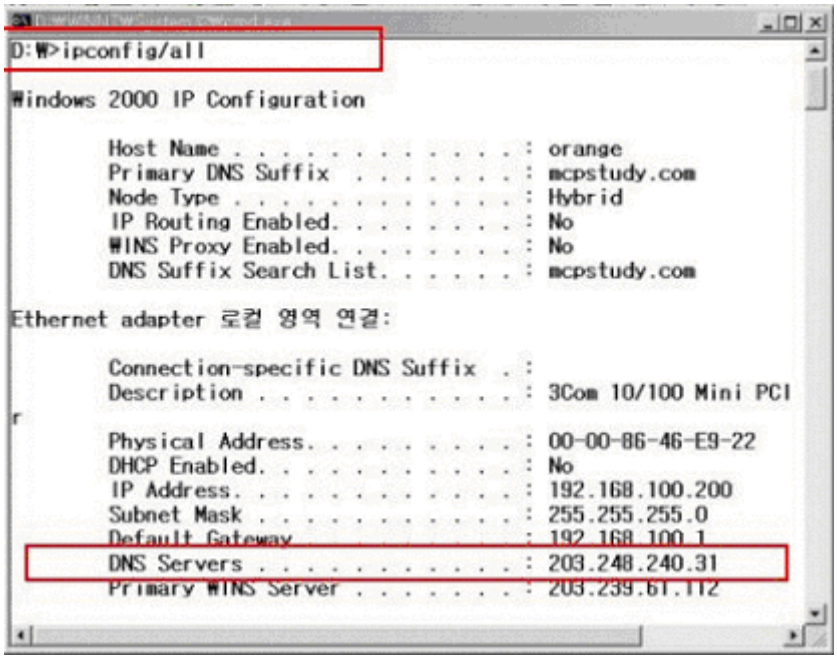
위에서 확인해 본 바로는 문자로서 입력하든, 숫자로서 입력하든 하나의 서버로 접근해서 동일한 내용을 얻을수 있다는 사실을 알 게 되었다. 그렇다면 사용자 입장에서 볼 때 어느 것이 보다 편하게 쓸 수 있는 주소일까? 여러분이 남들로부터 "쟤는 많이 특이해.."라는 소리를 듣지 않는 이상, [그림1]의 경우가 보다 편하다는 것을 느낄 수 있을 것이다.

TCP/IP환경에서 IP Address라는 것을 가지고 각각의 시스템을 구분하게 해 주고 있다. 웹 서버도 마찬가지고, 여러분이 집에서 사용하는 컴퓨터도 마찬가지이다. IP Address라는 것을 가지고 서로를 구분하고 네트워크 상에서 통신이 가능하다는 것이다. 문제는 이러한 IP Address라는 것이 쉽게 외워서 쓰기에는 다소 부담스러운 숫자형태로 되어 있기 때문에 이 숫자를 대신할 수 있는 문자를 할당해서 www.microsoft.com과 같은 문자를 사용하고 있는 것이다. 하지만, 숫자를 대신할 수 있는 문자를 할당했다고 해서 IP Address라는 숫자는 쓰이지 않아도 된다는 것은 결코 아니다. 우리가 컴퓨터이름을 사용해서 통신을 시도했다고 할 지라도 실제 통신을 위해서는 반드시 해당 컴퓨터 이름을 가지고 있는 컴퓨터의 IP Address를 알아야만 하는 것이고, 특별한 서버를 둬서 그러한 일만 전문적으로 처리하도록 역할을 맡기게 된 것이다. 그 서버가 바로 DNS서버이다.
그렇다면 [그림1]에서 여러분이 www.microsoft.com이라는 컴퓨터이름을 입력했는데 어떻

게 '207.46.230.219'이라는 IP Address를 알아내서 통신이 이루어 진 것일까? 여러분은 단지 이름하나 치고 [Enter]키 한번 눌렀을 뿐인데. 여러분의 컴퓨터의 TCP/IP설정을 확인해 보라. 아래의 그림을 볼 수 있다. 화면은 Windows 2000 Professional컴퓨터의 등록정보이다. OS마다 다르다. 다른 인터페이스는 무시하고 넘어가자. 중요한 것은 DNS서버라고 하는 항목에 역시 IP Address가 적혀 있다는 것이다.



[그림3. TCP/IP 등록정보]



[그림4. IPCONFIG]

이러한 DNS서버의 IP정보를 담고 있는 컴퓨터를 가리켜서 DNS Client라고 부를 수 있다. DNS를 사용하는 클라이언트라는 의미이다. DNS클라이언트는 DNS서버의 도움이 필요한 어떤 작업을 해야 할 경우가 있을 때마다, 설정된 IP Address의 컴퓨터를 찾아가서 문의를 하게 된다. 위에서 확인했던 웹 브라우저를 통해서 웹서버에 접근하는 작업이 대표적인 예이다. DNS클라이언트가 있다면 반대편엔 분명히 이러한 응답을 처리해야 할 DNS서버가 있어야 한다. <그림5>에서 확인할 수 있다. DNS서버로 설정된 서버를 살펴보면 그림처럼 DNS Server라고 하는 서비스(Application)이 시작되고, 동작하고 있음을 확인할 수 있을 것이다.

DHCP Client	IP 주소와 ...	시작됨	자동	LocalSystem
DHCP Server	DHCP(Dyn...	시작됨	자동	LocalSystem
Distributed File S...	LAN 또는 ...	시작됨	자동	LocalSystem
Distributed Link ...	네트워크 ...	시작됨	자동	LocalSystem
Distributed Link ...	볼륨 간의 ...	시작됨	자동	LocalSystem
Distributed Trans...	둘 이상의 ...	시작됨	자동	LocalSystem
DNS Client	DNS(도메...	시작됨	자동	LocalSystem
DNS Server	DNS(도메...	시작됨	자동	LocalSystem
Event Log	프로그램 ...	시작됨	자동	LocalSystem
Fax Service	팩스를 보...		수동	LocalSystem
File Replication ...	여러 서버 ...	시작됨	자동	LocalSystem
FTP Publishing ...	인터넷 정...	시작됨	자동	LocalSystem
IIS Admin Service	인터넷 정...	시작됨	자동	LocalSystem
Indexing Service	로컬 및 원...		수동	LocalSystem
Internet Authenti...	전화 접속 ...	시작됨	자동	LocalSystem

[그림5. DNS Server Service status]

정리하면, 사용자가 www.mcpstudy.com이라는 컴퓨터이름을 주소창에 입력하면, 사용자의 컴퓨터가 DNS클라이언트로 설정되어 있기 때문에 설정된 DNS서버로 www.mcpstudy.com라는 컴퓨터이름에 해당하는 서버가 무슨 IP Address를 사용하고 있는지 문의를 하게 되고, 설정된 DNS서버는 사용자의 요청에 대해 '207.46.230.219'라는 응답을 하게 된다. 이 과정을 통해서 사용자는 IP Address를 알아냈기 때문에 이 IP Address를 가지고 통신을 시도하고, 해당 서버에 접속하여 홈페이지를 받아올 수 있었던 것이다.

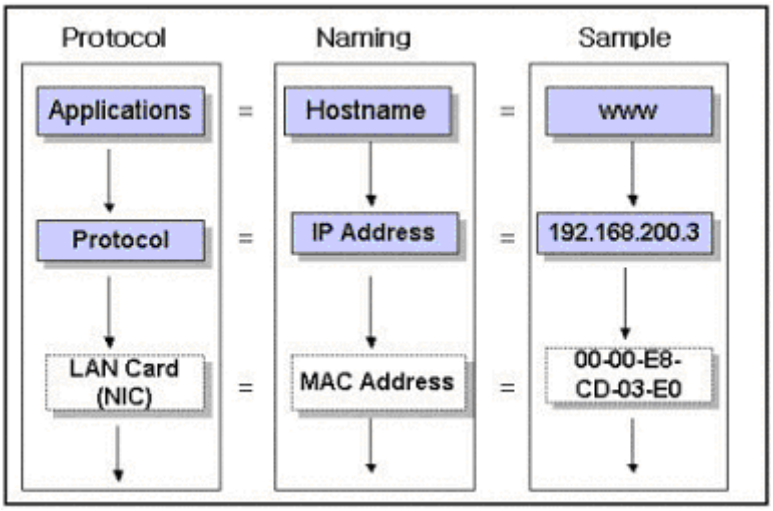
개념상으로는 단순해 보이지만, 이러한 DNS가 동작할 수 있도록 하기까지 수많은 사람들의 노력이 있었다. 이제부터는 조금 더 구체적으로 접근해 보도록 하자. DNS는 왜 생겨났을까? 탄생배경에 대해서 아는 것은 DNS를 공부하는데 상당히 도움이 된다. DNS의 필요성을 이해해보도록 하자.

먼저 말했던 것처럼 TCP/IP프로토콜을 사용하는 모든 컴퓨터(보통 host라고 부른다)들은 네트워크상에 각각 컴퓨터들을 구분하는 방법으로 고유하게 할당된 IP Address를 사용한다. 우리는 분명히 컴퓨터들을 지칭할 때 그들에게 할당한 컴퓨터이름을 이야기하지만, TCP/IP 세계에서는 이름보다는 IP Address가 보다 절대적이다. 학교의 한 학급을 생각해 보자. 한

반에 동명이인은 있을 수도 있지만, 번호가 같은 학생은 있을 수 없다. 더 생각해 보면, 1학년1반의 23번과, 1학년2반의 23번은 분명히 다른 학생이다. 23번이라는 번호는 같지만, 반이 틀려서 서로를 유일하게 구분해주고 있는 것이다. 이것과 IP Address는 개념이 유사하다. 한 학생이 학교에 입학해서 번호를 부여받게 되면 결국 그 학생은 학교내에서 고유하게 구분될 수 있는 "학번"이라는 것이 할당된다. 이렇듯 IP Address는 네트워크에서 유일한 주소인 것이다.

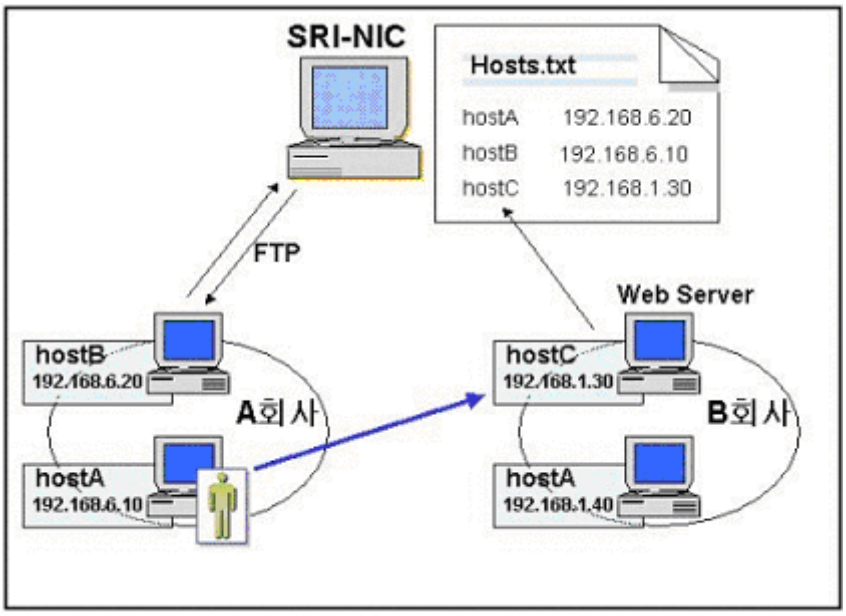
IP Address는 각각 8bit단위마다 .(dot)로 구분된 32bit체계로 이루어져 있다. 실제로 표기할 때는 2진수보다는 보기가 용이한 10진수로서 기록하고 있다. 예를 들어서 192.168.200.100 과 같은.. 그래서 컴퓨터들은 IP주소를 보고 서로를 구분하고 통신을 해야만 하는데, 문제는 이러한 IP주소가 사용자가 기억해서 쉽게 입력하고 접근하기에는 조금 까다로운 숫자 형태라는데 있다.(숫자 좋아하는 사람이 얼마나 있겠는가?) 그래서 이러한 주소를 조금 더 쉽게 표현할 수 있는 방법이 없을까 하고 연구한 결과, IP address마다 별도의 이름을 할당을 하게 되었다. 예를 들면 www, ftp, smtp, mail, korea 등이 그것이다. 당연히 사용자 입장에서 보면 숫자보다는 보기 편한 문자로 되어 있기 때문에 접근하기가 용이할 것이다. 이 이름을 가리켜 "HOST NAME"이라고 부르며, 별명(Alias)라는 표현을 한다. IP address대신 붙이는 별명이라는 뜻이다.

TCP/IP통신의 기본흐름을 도식화해 보았다. 거추장스러운 추가 프로토콜을 모두 걷어내고, dns를 이해하는데 필요한 핵심내용만 정리한 그림이다. 우리가 웹 브라우저를 가지고 웹서버에게 데이터를 보내는 과정을 생각해 보자. 웹 브라우저는 Application이다. 실제로 네트워크에 데이터를 날려보내는 것은 물론 네트워크 어댑터카드(NIC)이다. 그 둘을 연결시켜주는 것이 무엇일까? 그것이 바로 Protocol인 것이다. 결국 어플리케이션을 가지고 통신을 하기 위해서는 반드시 프로토콜을 도움을 받아야 하고, 마지막으로 네트워크 어댑터라는 물리적인 장비를 통해서 상대방 컴퓨터와 통신을 할 수 있게 되는 것이다.



[그림6. TCP/IP통신의 기본흐름]

앞에서 보았던 IP Address와 Hostname의 관계를 생각하고, [그림6]을 보면 이해를 도울 수 있다. 결국 어플리케이션이 hostname을 사용한다고 하더라도 실제 통신을 위해서는 거쳐야 할 단계가 IP Address라는 사실을 알수가 있게 된다. MAC Address라는 것은 그 다음 단계이다. 위의 단계에서 작업이 더 이상 진행되지 않는다면더 이상 통신을 수행할 수가 없게 된다. 아래의 [그림7]을 예제로 설명한다.



[그림7. hosts file을 이용한 Name resolution]

[그림7]의 예제는 A회사의 hostA컴퓨터를 사용하는 사용자가 B회사의 hostC라는 이름을 가진 Web server를 찾아가서 서비스를 제공받고자 한다. A회사와 B회사는 물론 인터넷이라

는 네트워크 환경으로 연결되어 있겠지만, hostA의 사용자로서는 자신이 원하는 데이터를 가지고 있는 서버가 이 지구상에 도대체 어디에 존재하는지 알 턱이 없다. 다만 알고 있는 정보라고는 오직 하나, 서버이름을 알고 있을 뿐이다. 앞에서 설명할 때 서버이름을 알고 있더라도 통신을 위한 다음단계는 분명히 IP Address라는 것을 알아야 한다고 했다. 그렇다면 어떤 방법으로 서버이름을 통해서 IP Address를 얻어 낼 수가 있을 것인가? 이것을 가능하도록 하기 위해서 인터넷초기에는 다음과 같은 방법을 사용하게 되었다.
인터넷에 연결되는 모든 컴퓨터들은 각각 SRI-NIC이라는 컴퓨터의 hosts.txt라는 파일에 자신의 hostname과 IP Address정보를 등록하고, 각 회사에서는 FTP를 이용해서 이 파일을 다운로드 받아서 각각 외부에 통신할 때 이 파일에서 원하는 hostname에 대한 IP Address 정보를 찾아서 통신을 하는 방법이다. 충분히 가능한 일일 것이다. 전세계 인터넷에 연결된 모든 호스트가 하나의 서버에 등록될 것이기 때문에 당연히 사용자는 원하는 정보를 찾는데 큰 어려움이 없게 될 것이기 때문이다. 하지만, 여기에는 몇가지 문제점이 내포되어 있다.

첫 번째 문제점은 업데이트가 늦다는 것이다. B회사에 새로운 호스트가 추가되었다고 가정하면, 그 정보를 A회사에서 알기 위해서는 몇가지 절차가 필요하다. B회사에서는 새로운 호스트의 정보를 역시 Hosts.txt파일에 등록해야 하고, A회사는 새롭게 바뀐 파일을 다운로드 받아야만 비로소 사용할 수가 있게 되는 것이다. 처음엔 가능했을지 모르지만 요즘같은 폭발적인 인터넷 발전속도를 따라가기엔 당연히 역부족일 수밖에 없다.

두 번째 문제점은 네트워크 트래픽이다. 파일에 등록된 호스트이름이 늘어갈수록 파일크기는 커질 수밖에 없고 인터넷이 FTP관련된 불필요한 트래픽으로 가득 차는 우스운 일이 생길 수밖에 없다. 이에 데이터베이스를 한곳에 집중시키는 문제점을 해결하기 위한 방안이 필요해졌다. 가장 좋은 방법은 데이터를 분산시키는 방법일 것이다.

세 번째 문제점은 IP Address를 대신할 컴퓨터의 호스트이름 짓기가 너무 힘들다는 것이다. 모든 컴퓨터의 이름이 flat한 구조로(www, ftp, A회사 등) hosts.txt파일에 등록이 되었기 때문에 당연히 사용하고자 하는 이름은 hosts.txt파일내에서 유일해야만 정상적인 통신을 할 수가 있게 된다. 만일 어떤 회사가 자신의 회사의 웹서버의 이름을 "www"로 지어서 등록을 했다면 전세계의 어떤 컴퓨터도 더 이상 www라는 이름을 사용할 수 없을 것이다. 당연한 결과이다. 이러한 문제점도 역시 해결을 해야할 필요성이 생겼다.

Internet을 관리하는 사람들은 이러한 문제점을 해결할 필요성을 느꼈고, 위의 몇가지 문제점을 해결할 수 있는 "분산된 데이터베이스 형태의 새로운 방법"이 필요해진 것이다. 그 결과로 나온 것이 바로 "DNS (Domain Name System)"이다.

'빨리 DNS사용법이나 이야기하지 무슨 사설이 이렇게 긴가'라고 생각할 수도 있겠다. 하지

만, DNS가 단순히 사용법만 알아서 사용할 만큼 단순한 서비스가 아니다. 여러분들이 한번만 DNS에 대해서 체계적인 이해를 하여 둔다면 적어도 향후 몇 년간은 두고두고 도움을 받을 수 있는 중요한 서비스인 만큼 이런 내용을 소홀히 할 수가 없기 때문임을 이해해 주기 바란다. 길었던 사설은 이제 마무리 하겠다. 이제 DNS라고 하는 녀석의 구조를 알아보고, 접근방법, 설치, 구성 등에 대해서 차례대로 알아보도록 하자.

2.윈도우 2000 DNS 의 구조 및 Process

앞에서 우리는 Hostname을 IP Address로 분해(Name resolution)하는 방법으로 hosts.txt파일을 이용했을 때 생기는 몇가지 문제점을 살펴 보았다. 이름중복이라는 문제점을 해결하고, 현재상태의 컴퓨터의 호스트이름들을 반영할 수 있어야 하고, 관리적인 측면에서도 가장 손쉽게 접근할 수 있는 방법이라면 어떤 방법을 이용해야 할까? 이 점이 해결해야 할 중점과제이다.

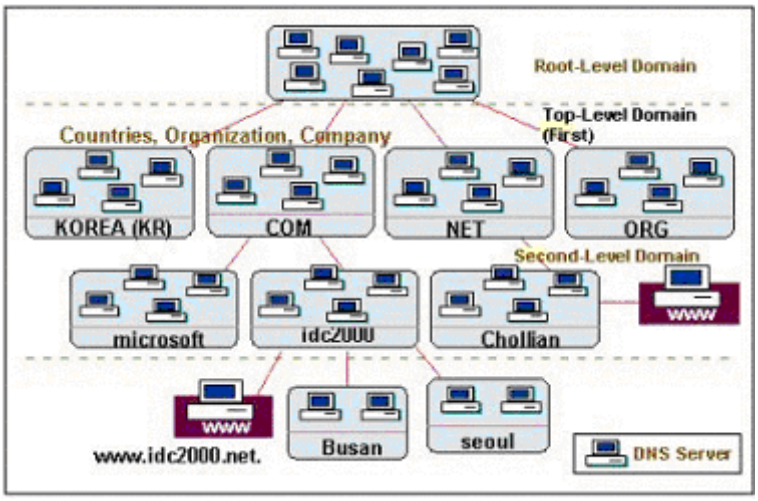
한가지 예를 들어보자. 회사에는 수많은 부서가 존재한다. 그렇듯 부서를 나누는 이유는 무엇일까? 물론 여러 가지 이유가 있겠지만 관리적인 측면을 무시할 수 없을 것이다. 모든 사원들을 사장이 혼자 관리하는 것보다는 당연히 부서를 나누고 부서장이 관리를 한다면, 보다 제대로 관리를 할 수가 있을 것이다. 또 부서가 없는경우에 한 개인은 "000"라는 형식으로 개개인의 이름을 가지지만, 부서로서 개인을 조직하게 되면 "00부서의 000"로서 불리우게 될 것이다. 한 회사에 "송원석"이라는 이름을 가진 사람이 둘이 있다고 하자. 부서의 개념이 없을 때는 단순히 "송원석"하고 부르면 누구를 부르는 건지 헷갈릴 수밖에 없다. 하지만 그들을 구별해 주는 부서가 있다면 "00부서의 송원석"이라는 식으로 어긋나지 않게 원하는 사람을 구별할 수 있게 된다.

앞에서 문제가 되었던 이름중복은 조금만 생각을 바꾸면 해결이 가능하다. 이름을 평면적인(flat) 이름으로 사용하는 것 대신에 부서처럼 다른 이름을 하나쯤 더 붙이면 되지 않을까? 예를 들면 "www"라는 이름 대신에 "A회사의 www"로서 등록을 한다는 것이다. 그렇다면 모든 회사의 웹서버가 "www"라는 이름을 사용하더라도 더 이상 이름의 충돌은 염려하지 않아도 될 것이다.

hosts파일크기의 증가로 인한 네트워크 트래픽을 줄이고, 빠른 업데이트를 가능하게 하기 위해서는 관리를 분산시킬 수 있는 구조가 최선이라는 해결책을 찾고 결국 호스트들을 그룹화하는 작업을 하게 되었다. 그 그룹을 바로 "도메인(domain)"이라고 부른다. 여기서의 도메인은 Microsoft가 사용하는 도메인이라는 용어와는 구별을 해야 한다. Microsoft의 도메인은 어디까지나 마이크로소프트의 NT라는 운영체제가 제공하는 회사의 작업그룹에 한정되는 반면에, DNS에서의 도메인은 전 세계에서 구분할 수 있는 작업그룹이다.

DNS의 도메인은 수없이 많은 종류가 있을 수 있다. 대표적으로는 여러분들이 속해 있는 회사를 생각할 수 있고, 국가를 구분할 수도 있고, 특별한 단체를 구분지을 수도 있다. 인터넷에 연결되고 자신들의 집단을 구분할 수 있는 이름을 가지고자 한다면 어느 누구라도 Domain이라는 하나의 작업그룹을 할당받을 수 있다는 이야기다. 추가로 이러한 그룹 역시 또 다른 그룹에 소속되어 있고, 최상위에 모든 그룹은 하나의 그룹으로 묶이게 된다. 회사의 조직구조와 비슷하다. 회사안에는 몇 개의 사업본부가 있고, 사업본부 안에는 몇 개의 부서가 있고, 부서 아래에 팀이 있고, 팀 안에 팀원들이 소속되어 있는 구조와 같다. 실제 구조를 살펴보면 우리가 윈도우 탐색기를 통해서 볼 수 있는 폴더구조와 흡사하다. 최상위에 root(c:)가 있고, 그 밑에 폴더가 있으며 폴더밑에는 또 다른 폴더가 있고, 최종적으로 폴더 밑에는 파일이 들어있다. 이러한 형태의 구조를 우리는 계층구조라고 부른다.

결국 최종적으로 만들어진 DNS의 구조를 살펴보면 아래의 [그림8]과 같다.



[그림8. Domain Name Space의 계층구조]

위의 그림에서 볼 수 있는 것처럼 DNS는 뚜렷한 계층구조를 가지고 있다. 최상위에는 모든 도메인의 근본이 되는 root domain을 두고, null라벨(공백의 문자, " ")로서 이름을 할당하였다. 이것을 가리켜서 root-level domain이라고 부른다. root-level domain아래에는 top-level domain을 두는데, 우리에게 친숙한 이름이 보일 것이다. kr, com, net, org등이 그것이다. 각각의 top-level domain은 특별한 원칙을 가지고 있다. 어느 집단이 사용을 하는 이름인지를 알아볼 수 있게 만든 것이다. 현재 약 250여개 정도의 top level domain이 사용되고 있다. [표1참고]

<Top-Level-Domain Organization>

Com : 영리를 목적으로 하는 기관 (commercial Organization)
Net : 네트워크 기관 (chollian이나, hitel등이 사용하고 있다.)
Org : 비영리 기관
Gov : 정부기관
Mil : 군사기관
Edu : 교육기관
Int : 국제적인 기관
Kr (Korea), jp (Japan), Au (Australia) 등 : 국가이름

[표1. top level domain 이름구조]

이들 root domain과 top level domain은 우리가 등록해서 사용할 수 있는 이름이 아니다. InterNIC에서는 이미 전세계의 도메인들의 밑바탕이 될 이름을 등록하였고, 실제로 우리가 사용하게 될 이름은 이들 top level domain아래의 도메인부터라고 할 수 있다. [그림8]에서 볼 수 있는 것과 같이 top level domain아래에는 second level domain이 있고, 거기에서 실제로 우리의 회사의 이름을 반영하는 이름들을 찾아볼 수 있게 된다. 우리가 도메인 등록을 한다고 이야기를 하면 바로 그러한 이름을 이야기하는 것이다. 물론 COM이나 Net과 같은 도메인 아래에 여러분의 회사를 등록한다면 바로 second level에서 등록하는 것이 가능하겠지만, 우리나라를 가리키는 Kr도메인 아래에 회사를 등록하고자 한다면, 우리는 한단계 더 아래로 내려갈 수밖에 없다. Kr아래에 또 다시 영리회사를 가리키는 "co"라는 이름이 second level로서 등록되고, 여러분이 원하는 회사이름은 그 아래에 등록되어야 하기 때문이다. 왜 그래야 할까? 답은 뻔하지 않은가! 미국에서 만들어서 사용하고 그네들이 발전시켜 오고 있는 것이기 때문에 우리는 그들이 할당해준 kr이라는 top level domain을 우리나라 입장에서는 root로 삼고, 다시 그 안에서 분리작업을 해야 하기 때문인 것이다. 아쉬운 일이지만 힘을 키우는 도리밖에 어쩔 수가 없는 노릇이다.

이러한 계층구조를 가진 분산데이타베이스 구조를 가리켜서 DNS (Domain Name Space혹은 Domain Name System)라고 부른다. 그렇다면 이들 DNS는 실제로 어디에 존재하는 것일까? 질문이 너무 막연해 보인다. 여러분이 즐겨찾는 인터넷이라는 공간을 가리켜 우리는 사이버공간이라고 부른다. 실체는 없는 가상공간이라는 뜻이다. 실제로 존재하는 것은 클라이언트 시스템으로서 사용하는 PC가 있고, 그들이 접속해서 사용하는 Web server라는 것이 있다. 전세계에 네트워크에 연결된 이들 모든 컴퓨터들이 어우러져서 InterNet이라고 부르는 하나의 커다란 가상공간이 존재하게 된 것이다. DNS도 마찬가지이다. 어느 한곳에 DNS라는 이름이 기록되고 관리되고 있는 형태는 아닌 것이다. 실체로 본다면 역시 물리적인 시스

템인 서버가 존재하겠지만, 이들 모든 서버들이 만들어 내고 있는 하나의 가상공간인 셈이고, 이러한 가상공간을 제공하는 서비스를 DNS를 관리하는 서버, 바로 DNS Server가 이루어놓고 있는 것이다. 결국 <그림8>에서 각각의 그룹을 가리키는 도메인이라는 녀석이 존재하고 있었는데, 그 도메인의 중심에는 바로 DNS Server라고 하는 컴퓨터가 차지하고 있게 된다.

DNS를 정리해보면 "각각 회사를 구분해주는 도메인이름을 관리하는 DNS Server들이 모여서 만들어낸 가상이름공간"이라고 요약을 해 볼 수 있겠다.

다음은 DNS의 표기법에 대해서 알아보자. 이름충돌의 문제를 해결한다는 예제를 가지고 생각해 보자. [그림8]에서 second level domain중에 "idc2000"이라는 이름이 있고, 그 회사가 가지고 있는 서버중에 "www"라는 웹서버가 있다. 또한 Chollian이라는 회사도 역시 "www"라는 웹서버를 가지고 있다. 이들은 어떻게 똑같은 이름을 가질 수가 있게 되는가? 바로 DNS에서 바라보는 서버의 이름체계가 이 문제점을 해결해 주고 있다. 이제 더 이상 인터넷에 연결된 서버들은 서로를 구분하는데 있어서 평면적인 www라는 형태의 단순한 이름체계로 사용되지는 않는다. 인터넷상에서 서버들의 이름은 아래와 [표2]의 표기법을 사용하기 때문이다.

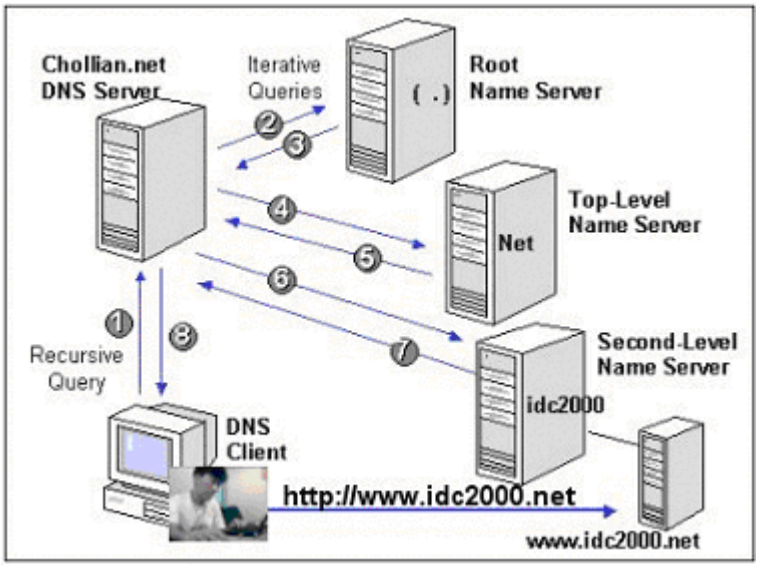
Fully Qualified Domain Name (FQDN)
DNS에서의 서버의 이름 = Host name + Domain Name
예) www + idc2000 + net + (공백)
표기법) www.idc2000.net. www.microsoft.co.kr

[표2. Fully qualified Domain Name]

결과적으로 idc2000의 웹서버인 www서버는 www.idc2000.net과 같이 표기되고, 천리안의 웹서버는 www.chollian.net과 같이 표기되고 구별되기 때문에 전혀 다른 이름을 가지게 되는 것이다. 예전에 사용했던 평면적인 이름체계에서 벗어나 계층구조의 이름을 가짐으로써 이름충돌에 대한 문제점을 해결하게 된 것이다.
한가지 고려사항은 여전히 남아있다. 같은 계층에 2개의 같은 도메인이 있을 수가 있을까? 라는 것이다. 예를 들면 net이라고 하는 도메인에 chollian이라는 도메인이 등록되어 있다. 또 다른 회사에서 역시 그들도 chollian이라는 이름을 사용하고자 한다면 당연히 허용해 주어선 안될 일이다. 이러한 문제 때문에 새로운 신기술이 나오면 서로 먼저 그 이름을 이용한 도메인을 등록하기 위해서 도메인전쟁이라고까지 표현되는 등록싸움이 발생하고 있는 것

이다. 그렇다면 왜 도메인 등록을 하여야 할까. 그냥 우리가 사용하고 싶은 이름을 마음대로 쓰면 되는 것 아닌가? 라고 생각한다면 다음에 설명할 DNS의 이름분해과정을 이해하면 생각을 바꿀 수가 있을 것이다. 지금부터는 DNS Server가 어떤 과정을 통해서 이름분해 과정을 진행하는 지에 대해서 설명하도록 하겠다.

[그림9]의 예제에서 DNS Client로 표기된 PC에 앉은 사용자는 www.idc2000.net이라는 이름을 가진 IDC2000회사의 웹서버에 접근해서 웹서비스를 받고자 한다. 사용자가 www.idc2000.net의 IP Address를 알고 웹 브라우저를 통해서 이름이 아닌 IP주소로 접근을 했다면 간단하겠지만, 사용자가 그런 주소를 알 필요는 없다. 이 사용자의 PC는 천리안의 DNS서버를 사용하는 DNS클라이언트로 셋팅이 되어 있다. DNS Client에 대한 부분은 1장에서 확인한 바 있다.



[그림9.DNS Name Resolution Process]

사용자는 웹 브라우저의 주소창에 http://www.idc2000.net 이라고 입력하고 [Enter]버튼을 누른다. 이 요청은http라는 프로토콜을 가지고 www.idc2000.net이라는 이름의 서버에 요청을 보내겠다는 뜻이다. 결국 웹서비스를 요청하는 메시지이다. 이 요청을 처리해 주기 위해서 사용자의 시스템은 먼저 www.idc2000.net의 IP Address를 알아야 한다. 하지만, 스스로 알 수 있는 방법이 없다. 그런데 DNS Client로 셋팅이 되어 있기 때문에 DNS Server로 셋팅된 서버에게 질문을 던질 수가 있다.

사용자의 시스템은 자신의 TCP/IP등록정보에 DNS Server로 셋팅된 IP Address의 서버에게 www.idc2000.net의 IP Address를 요청하게 된다.(그림9-①) 이 요청을 가리켜서 resursive

query(재귀적질의)라고 한다. resursive query라고 함은, 이 요청을 받은 서버가 '모 아니면 도' 둘중의 하나의 응답을 해야만 하는 Query를 말한다. IP Address를 가르쳐주거나, 에러 메시지를 반납하거나.

①의 요청을 받은 DNS Server는 응답을 해야 하는데, 자신 역시 자신이 관리하는 이름을 알 뿐이지 어디 있는지도 모르는 idc2000이라는 회사의 웹서버의 IP Address를 알 턱이 없다. 그래서 먼저 DNS Server는 Cache라고 부르는 데이터베이스에서 클라이언트가 요청하는 정보가 있는지를 확인해 본 다음, 없다면 다른 DNS에게 질문을 던져서 클라이언트의 요청을 해결하려는 시도를 하게 된다. Windows2000 시스템에 DNS Service를 추가하고, DNS Server로 동작하게 되면 그 서버에는 한가지 정보가 기본적으로 추가된다. 바로 Root Domain을 관리하고 있는 DNS Server들의 목록이다. 서버는 그 정보를 이용해서 root domain의 DNS Server에게 클라이언트가 요청한 www.idc2000.net 에 대한 이름분해 요청을 보내게 된다. (그림9-②) 이 때의 Query를 가리켜서 Iterative Query (반복적 질의)라고 부른다. Recursive Query와는 조금 성격이 틀린 것을 알 수 있을 것이다.

Root name server는 요청을 받았으니 응답을 해야 할텐데, 자신 역시 모든 정보를 알고 있는 것은 아니다. 자신이 알고 있는 것은 "net"도메인을 관리하는 DNS Server의 정보를 알고 있을 뿐이다. 그래서 천리안의 DNS Server가 원하는 모든 정보에 대한 응답이 아닌, "net" 도메인을 관리하는 DNS Server의 정보를 제공해 주게 된다. (그림9-③)

천리안의 Name Server는 한번의 query를 통해서 net도메인의 Name server의 정보를 알 게 되었다. 아직 완전한 정보는 얻지 못했기 때문에 임무에 충실할 책임을 다한다. "net"도메인의 DNS Server에게 www.idc2000.net. 의 IP정보를 요청하게 된다. (그림9-④)

하지만 'net'도메인의 DNS Server역시 천리안의 Name server가 요청한 호스트이름에 대한 IP정보를 알지는 못한다. 그렇지만 idc2000을 관리하고 있는 DNS서버의 정보는 알고 있다. 자신이 관리하고 있는 도메인이기 때문이다. 그래서 idc2000.net 의 DNS서버의 정보로서 요청에 응답해 주게 된다. (그림9-⑤)

idc2000.net.도메인의 Name server정보를 얻은 천리안의 name server는 idc2000.net의 name server에게 www.idc2000.net.이라는 자신의 클라이언트가 요청했던 호스트에 대한 IP정보를 요청하게 된다. (그림9-⑥)

이 요청을 받은 idc2000.net.의 DNS Server는 요청한 FQDN이 자신이 관리하고 있는 도메인에 소속된 호스트이기 때문에 IP Address정보를 응답해 줄 수가 있게 된다.(그림9-⑦)

몇 번의 query를 통해서 DNS Client가 요청한 정보를 알게 된 천리안의 DNS Server는 www.idc2000.net의 IP Address를 제공해 줄 수가 있게 되었다.(그림9-⑧)

그렇다면 이제 사용자는 DNS Server를 통해서 원하는 웹서버에 대한 IP Address를 알 게 되었기 때문에 TCP/IP통신을 할 수가 있게 되었다. http프로토콜로써 웹서버에게 서비스를 요청하고, 웹서버는 홈페이지를 서비스해 주게 되는 것이다.

글로써 기록하니 상당히 복잡해 보인다. 하지만, 실제로 사용자 입장에서 보자면, 사용자는 웹 브라우저의 주소창에 친숙한 이름표기법에 따르는 이름을 입력했을 뿐이고, 내부사정은 알바가 아니지만 홈페이지를 볼 수가 있게 되었다. 이러한 작업이 가능하게 만들어지게 하기 위해서 DNS Server는 부지런히 발로 뛰며 서비스를해 주고 있었던 것이다. DNS가 없었다면 이런 일이 가능했을까? 실로 DNS의 위력은 엄청나게 보인다. 필자는 감히 인터넷을 이끌어가고 있는 것은 바로 DNS라고 말하고 싶다. 그만큼 인터넷에 있어서 DNS의 기능은 대단하다고 할 수 있기 때문이다. 만일 DNS가 없다면 지금 우리가 사용하는 인터넷이라는 환경이 상당히 많이 다를 거라고 생각한다. TV에서 하고 있는 광고중의 상당수는 도메인이름을 알리는 광고를 하고 있다. DNS가 없다면 업체에서는 도메인이름을 알리는 광고가 아니라 IP Address를 알리는 광고를 해야만 할 것이다. 채림이가 나와서 hihome.com이라는 점(.)을 내세운 광고를 하고 있지만, 그것도 안될 말이고, 수많은 dot com으로 대변되는 벤쳐기업들도 이름자체가 애매해졌을 것이다.

그렇다면 앞서서 가졌던 한가지 의문중에 "왜 반드시 원하는 도메인이름을 등록해야만 사용할 수 있을까"라는 것을 생각해 보자. [그림9]의 예제에서 웹서버 입장을 생각해 보자. idc2000이라는 회사는 자신의 웹서버를 통해서 회사를 홍보하고, 몇가지 서비스를 판매하는 일을 하고 있다고 가정해 보자. 사용자가 자신의 웹서버의 IP Address를 알고 찾아오게 하는 일은 가장 중요한 일이라고도 할 수 있다. 그런데 앞에서 보았듯이 사용자가 자신의 웹서버의 IP주소를 알기 위해서는 어떤 과정이 있었는가? idc2000의 DNS Server까지 요청이 와야 하는데, 그것보다 선행되었던 작업이 net도메인의 DNS Server를 찾는 과정이었다. 만일 net도메인의 DNS서버에 idc2000의 도메인의 정보가 등록되어 있지 않았다면 당연히 net도메인의 DNS Server는 천리안의 name서버의 요청에 그런 도메인은 없다라는 에러메시지를 전송해야만 했을 것이다. 당연히 천리안의 DNS Server는 이름분해를 처리하지 못했기 때문에 사용자에게 에러메시지를 반송할 수밖에 없게 되는 것이다.

그래서 도메인등록이라는 용어를 풀어본다면, "회사에서 사용하고 싶은 도메인 이름을 관리하는 DNS Server의 정보를 상위 도메인을 관리하는 DNS Server에 등록하는 작업"이라고 생각할 수 있겠다.

처음 볼 때는 상당히 번거로운 작업이라는 생각이 들 수도 있다. 하지만, 이러한 DNS의 진행과정은 2장에서 보았던 hosts.txt파일을 이용한 이름분해의 과정이 가졌던 모든 문제점을 해결할 수 있는 방법이 되고 있다. 문제점과 비교해서 DNS의 이름분해과정을 연관시켜서 정리해 보길 바란다.

지금까지 DNS의 구조와 이름분해과정(name resolution process)에 대해 설명하였다.

3.윈도우 2000 DNS 설치 및 구성

1. Windows 2000 DNS Service 설치

이번 장에서는 한글 Windows 2000 Server 를 기준으로 DNS 를 설치하고 구성하는 방법에 대해서 설명한다. 앞에서 살펴보았듯이, DNS 를 사용하기 위해서는 먼저 우리가 원하는 도메인을 등록해야 한다. 도메인 관리는 InterNIC 에서 담당하고 있다. 국외 도메인의 경우(.com .net .org 등)는 ICANN (The Internet Corporation for Assigned Names and Numbers / http://www.icann.org)에서 등록을 할 수 있고, 국내의 도메인의 경우(.co.kr .pe.kr 등)는 KRNIC (Korea Network Information Center / http://www.nic.or.kr)에서 등록할 수 있다. 하지만, 일반적으로는 직접 등록을 하지 않고, 등록대행기관을 통해서 하고 있다. 특히 국외 도메인의 경우는 기타 제반서류가 영문으로 처리가 되어야 하므로국내도메인 등록대행기관을 통해서 하는 것이 훨씬 수월하다고 하겠다. 국내 도메인등록 대행기관이라면 WHOIS ,Doregi, Netpia 등 많은 기관이 있다. 일반적으로 도메인을 등록할 때는 아래의 몇가지 정보를 제공해 주어야 한다.

① 회사 혹은 개인의 이름
② 주소
③ 관리자
④ Name Server (= DNS Server)

이 장에서 DNS 설치의 예제로 사용할 도메인은 필자가 등록한 mcpstudy.com 이라는 도메인이다. 먼저 아래에 캡쳐해 놓은 화면을 살펴보자.

```
WHOIS information for mcpstudy.com:

The Data in Network Solutions' WHOIS database is provided by Network
Solutions for information purposes, and to assist persons in obtaining
information about or related to a domain name registration record.
Network Solutions does not guarantee its accuracy.  By submitting a
WHOIS query, you agree that you will use this Data only for lawful
purposes and that, under no circumstances will you use this Data to:
(1) allow, enable, or otherwise support the transmission of mass
unsolicited, commercial advertising or solicitations via e-mail
(spam); or  (2) enable high volume, automated, electronic processes
that apply to Network Solutions (or its systems).  Network Solutions
reserves the right to modify these terms at any time.  By submitting
this query, you agree to abide by this policy.

Welcome to DOREGI.COM's WHOIS data service.

Registrant:
Song won suk

   Domain Name: MCPSTUDY.COM
   Administrative, Technical, Billing Contact:
      Song won suk  runner01@chollian.net

   Domain name servers in listed order:

   blueapple.mcpstudy.com          203.239.61.112


The previous information has been obtained either directly from the
registrant or a registrar of the domain name other than Network Solutions.
Network Solutions, therefore, does not guarantee its accuracy or
completeness.
```

[화면 1.WHOIS 의 "mcpstudy.com" Domain 등록정보]

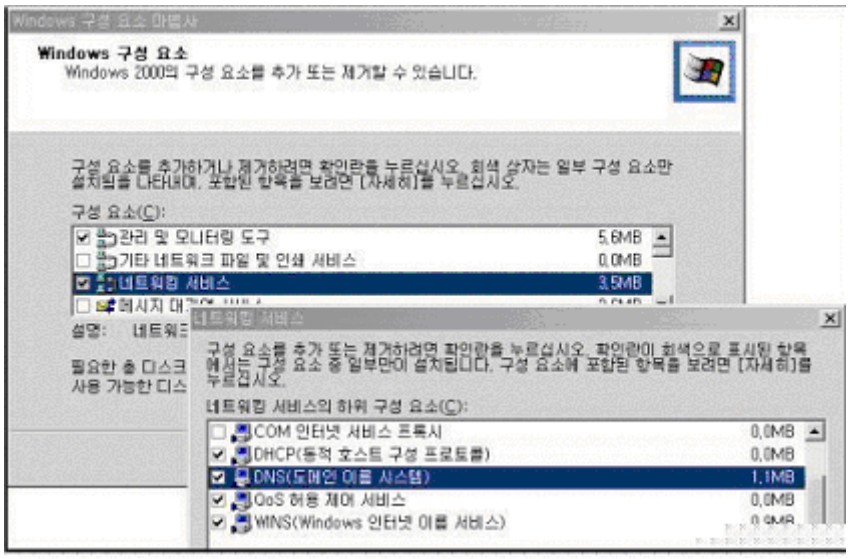
WHOIS (http://www.whois.org)를 통해서 도메인등록 정보를 알아볼 수 있다. [화면 1]에서 중요한 정보는 mcpstudy.com 을 관리하는 DNS Server 의 정보이다. 화면에서 볼 수 있듯이 mcpstudy.com 도메인은 blueapple.mcpstudy.com 이라는 FQDN 을 가진 Name Server 에 의해 관리되고 있으며 IP Address 는 203.239.61.112 이다. 이제 DNS 서버를 설치할 수 있는 기본준비가 되었다. 일반적인 DNS 설치과정은 아래의 순서를 따른다. 자세한 사항은 설치하면서 하나씩 알아가도록 하자.

< DNS Server설치순서 >

① DNS Domain 등록
② DNS Server Service설치
③ Forward Lookup Zone추가 (Primary DNS)
④ ③의 Zone에 record type별 record추가
⑤ Reverse Lookup Zone추가 (Primary DNS)
⑥ ⑤의 Zone에 record type별 record추가
⑦ Secondary Name Server설치 (if you need)
⑧ Dynamic Update구성 (option)
⑨ DNS Client 구성

[표 3. DNS Server 설치순서]

위에서 mcpstudy.com 의 Name server 로 등록한 서버의 IP Address 는 203.239.61.112 였다. 이 서버가 DNS Service 를 할 수 있도록 하기 위해서 먼저 DNS Service 를 추가해 보자. Windows2000 에서 service 라고 하는 것은 백그라운드에서 동작하도록 만들어진 네트워크 어플리케이션을 말한다. 오피스와 같은 프로그램을 포그라운드 어플리케이션이라고 하는 것과 비교하면 이해를 할 수 있을 것이다. [시작]-[제어판]-[프로그램추가/제거]-[Windows 구성요소]를 통해서 아래의 화면으로 접근할 수 있다. Windows2000 원본 CD 가 필요하다.[화면.2]



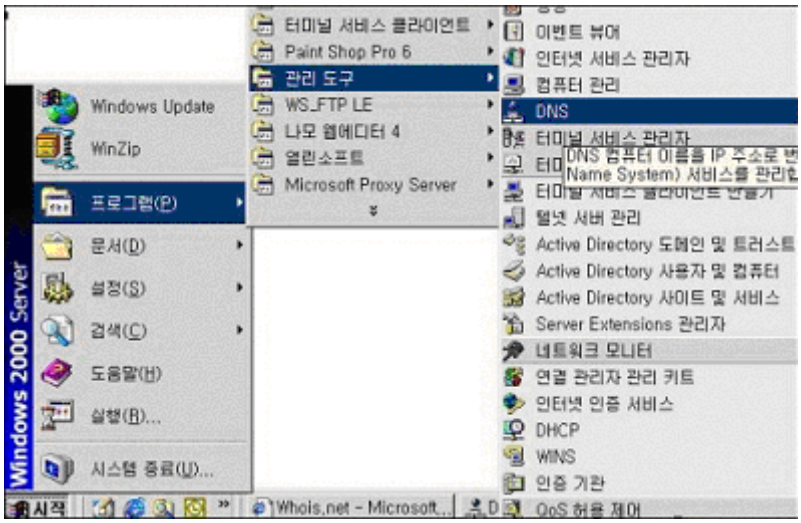
[화면 2. DNS Service 설치]

설치가 끝나고 [시작]-[프로그램]-[관리도구]-[서비스]항목을 보면 DNS Server Service 가 동작하고 있음을 확인할 수 있다. [화면 3]

DHCP Client	IP 주소와 ...	시작됨	자동	LocalSystem
DHCP Server	DHCP(Dyn...	시작됨	자동	LocalSystem
Distributed File S...	LAN 또는 ...	시작됨	자동	LocalSystem
Distributed Link ...	네트워크 ...	시작됨	자동	LocalSystem
Distributed Link ...	볼륨 간의 ...	시작됨	자동	LocalSystem
Distributed Trans...	둘 이상의 ...	시작됨	자동	LocalSystem
DNS Client	DNS(도메...	시작됨	자동	LocalSystem
DNS Server	DNS(도메...	시작됨	자동	LocalSystem
Event Log	프로그램 ...	시작됨	자동	LocalSystem
Fax Service	팩스를 보...		수동	LocalSystem
File Replication ...	여러 서버 ...	시작됨	자동	LocalSystem
FTP Publishing ...	인터넷 정...	시작됨	자동	LocalSystem
IIS Admin Service	인터넷 정...	시작됨	자동	LocalSystem
Indexing Service	로컬 및 원...		수동	LocalSystem
Internet Authenti...	전화 접속 ...	시작됨	자동	LocalSystem

[화면 3. DNS Server Service 상태]

DNS Service 를 설치하면, 관리도구에 [DNS]관리콘솔이 등록이 된다. 실행해 보자.[화면 4]

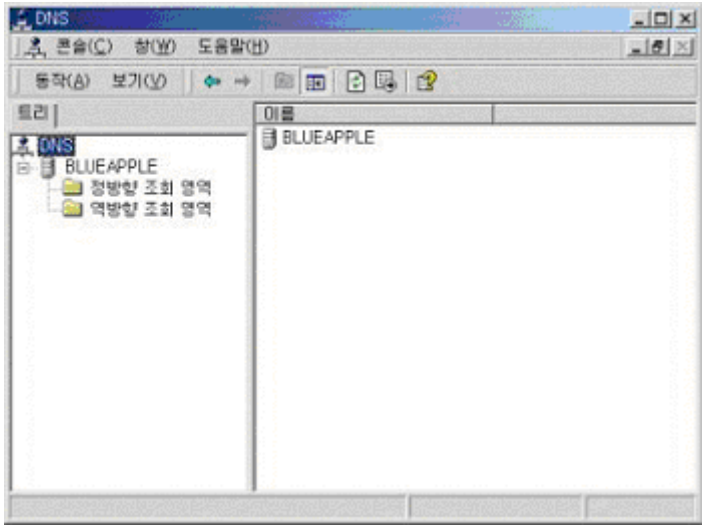


[화면 4. DNS 관리도구]

처음 만나게 되는 화면이다.[화면 5] DNS 라는 항목아래에 Blueapple 이라는 이름의 서버가 있고, 그 아래에는 정방향조회영역(Forward Lookup Zone)과 역방향조회영역(Reverse Lookup Zone)이라는 항목을 볼 수 있다.
DNS 에서의 영역(Zone)은 DNS Server 가 관리하는 하나의 관리단위이다. 하나의

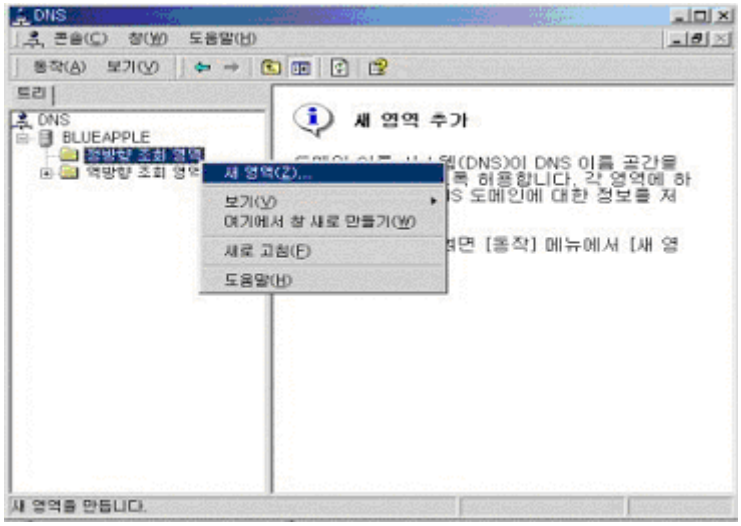
DNS 서버는 여러개의 도메인을 관리 할 수 있는데, 그 때 하나하나 도메인들을 가리켜서 Zone 이라고 부른다. 알고 넘어가야 할 것이 있다. Zone 이라는 것은 알겠는데 그렇다면 정방향조회, 역방향조회 라는 것이 무얼 말하는가? 원어로는 Forward Lookup, Reverse Lookup 이라고 되어 있다. 필자도 개인적으로는 영어를 잘하지도 못하고 또한 좋아하지도 않는다. 인터넷 관련된(물론 컴퓨터 자체가 그렇지만) 거의 모든 용어가 영어로 되어 있다. 물론 번역을 해서 표기를 하고 있지만, 가급적이면 본래 용어로 기억을 하도록 하자. 그편이 정신건강에 도움이 된다.

DNS 의 기본기능을 생각해보자. 1 장에서 설명한 대로 DNS 가 하는 일의 가장 주된 기능이라면,www.mcpstudy.com 이라는 hostname 을 IP Address 로 분해하는 일이었다. forward Lookup Zone 은 그러한 기능을 제공하기 위한 영역이다. 그렇다면 Reverse Lookup Zone 은 무엇일까? 거꾸로 IP Address 를 가지고 Hostname 을 알고 싶었다고 하면 어떻게 해야 할까. DNS Server 가 알아서 해 주겠지. 물론 그렇다. 하지만, 컴퓨터는 우리가 제공하지 않은 정보까지 스스로 알진 못한다. 누군가 IP Address 를 통해서 Hostname 을 알고자 했다면(Inverse Query) 그 정보를 제공하기 위해서는 역시 DNS Server 에 등록을 해 주어야만 한다. 즉 Reverse Lookup Zone 은 거꾸로 물어보기(Inverse Query)에 응답하기 위한 영역이 되는 것이다.

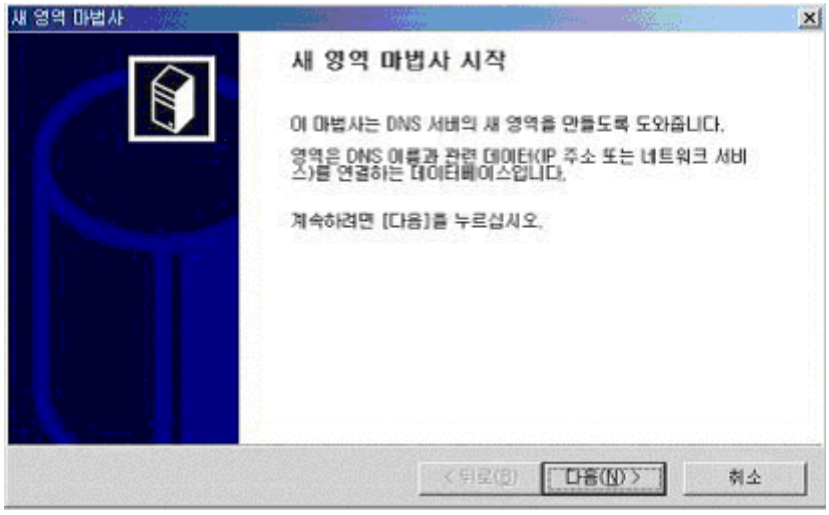


[화면 5. Windows 2000 DNS Console]

이 화면에서 관리하고자 하는 도메인을 추가한다. 예제에서는 위에서 등록했던 mcpstudy.com 이라는 도메인을 등록해 보도록 하겠다. [화면 6]

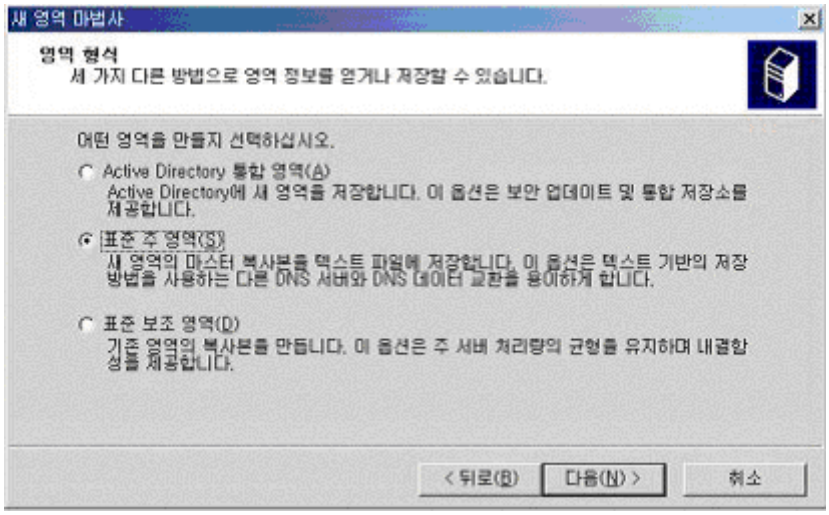


[화면 6. Zone 추가]



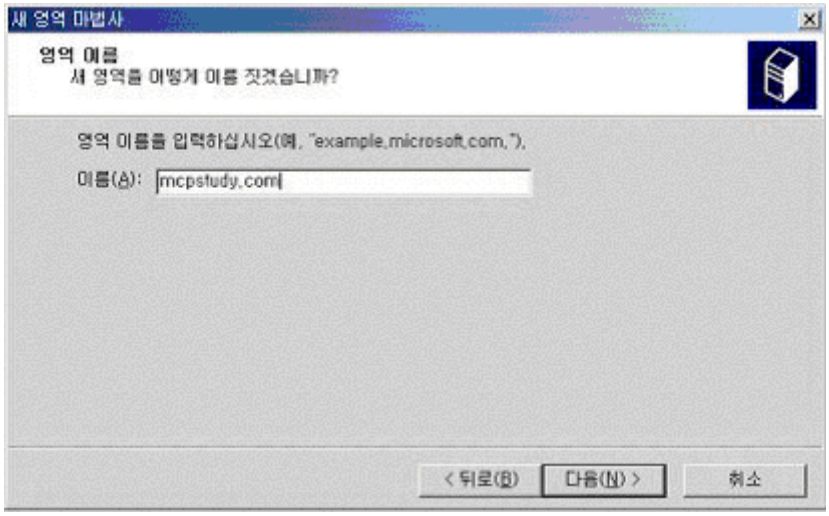
[화면 7. 새 영역 마법사]

영역을 선택하는 화면이 나온다.[화면 8] 기본설정은 표준주영역으로 되어 있다. 첫 번째의 'Active Directory 통합영역'은 Windows2000 에서 추가된 영역이다. 오로지 Domain Controll 로 설정된 (Active Directory 를 가진) 서버에서 DNS 를 설치할 때 사용할 수 있는 옵션이다. 나중에 바꿀 수도 있다. 표준보조영역은 처음 설치되는 DNS Server 에서는 사용할 수 없다. 회사에서 기존에 도메인을 관리하는 DNS Server 가 있는 상태에서 추가로 DNS Server 를 더 두려고 했을 경우에 사용할 수 있는 옵션이다. 뒤에서 자세히 다루어 보겠다. 여기서는 처음설치를 다루는 것이기 때문에 기본설정으로 그대로 두도록 하겠다.



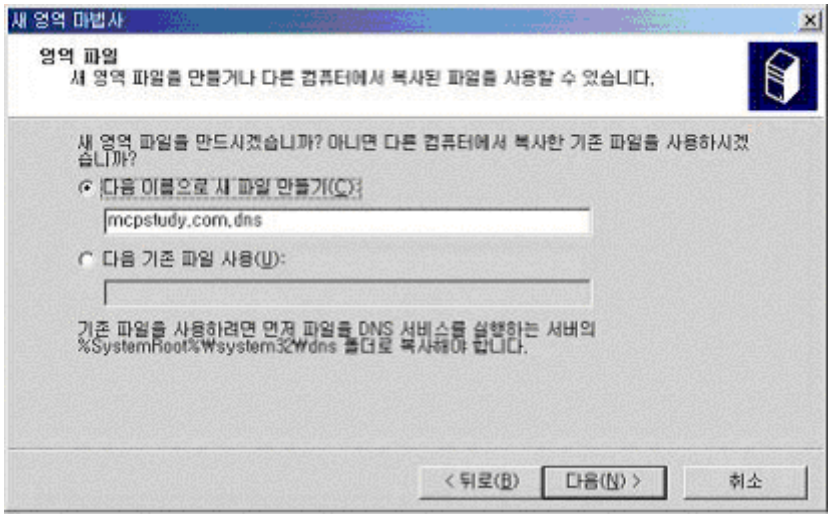
[화면 8. DNS 영역형식 설정]

다음에 영역이름을 입력해 준다[화면 9]. mcpstudy.com 이라고 입력하였다. 간혹 도메인이름과 호스트이름을 구별하지 못하는 경우를 보았다. 도메인이름은 논리적인 그룹을 표기하는 것이고, 호스트이름은 실제 서버에 주어진 컴퓨터의 이름이다. 예제에서 mcpstudy.com 도메인을 관리하는 서버의 이름은 blueapple.mcpstudy.com 이지만, 도메인의 이름은 'mcpstudy.com"이 되는 것이다. 만일 이 항목에 blueapple.mcpstudy.com 이라고 입력했다면, 그것은 mcpstudy.com 의 하위도메인으로서 인식되게 된다.



[화면 9. 영역 이름 입력]

DNS 데이터베이스는 파일형태로 저장된다. [화면 10]에서는 파일이름을 결정하는 항목이다. 기본적으로 도메인이름.dns 라는 파일로서 생성이 된다. 좋은 설정이다. 바꾸어도 되겠지만, 굳이 그럴 이유는 없다. 설치가
끝난다음, %systemroot%₩system32₩dns 폴더를 검색해 보자.
"도메인이름.dns"라는 파일을 찾을 수 있을 것이다.메모장을 통해서 열어볼 수 있는 텍스트파일이다

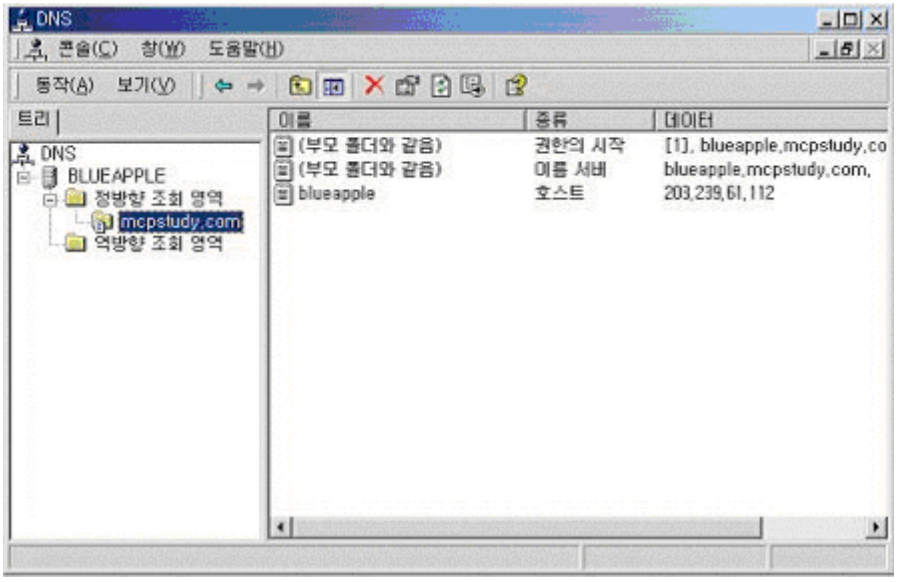


[화면 10. 영역 파일 이름 결정]

[화면 11. 영역 마법사 완료]

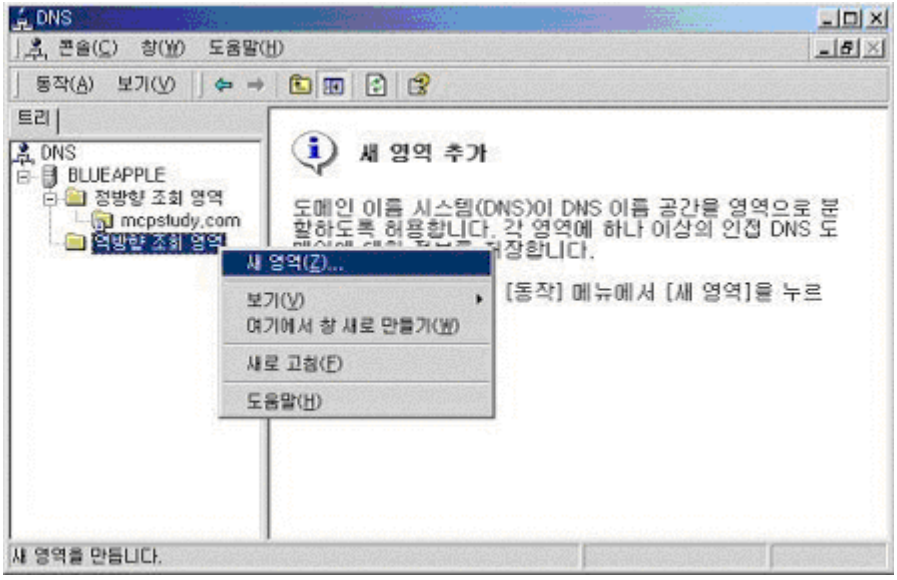
자. 이제 영역이 만들어졌다. [화면 12]에서 만들어진 mcpstudy.com Zone 을 확인할 수 있다. 콘솔 오른쪽에는 3 개의 레코드가 등록되어 있는 것이 보인다.

blueapple 이라는 호스트가 203.239.61.112 라는 정보를 보여주고 있다. 결국 blueapple 이라는 호스트의 FQDN 은 blueapple.mcpstudy.com 이 되는 것이고, 외부의 다른 DNS Server 가 이 정보를 요청했다면 Blueapple DNS Server 는 요청에 응답해 줄 수 있게 되었다.



[화면 12. mcpstudy.com Zone 의 생성확인]

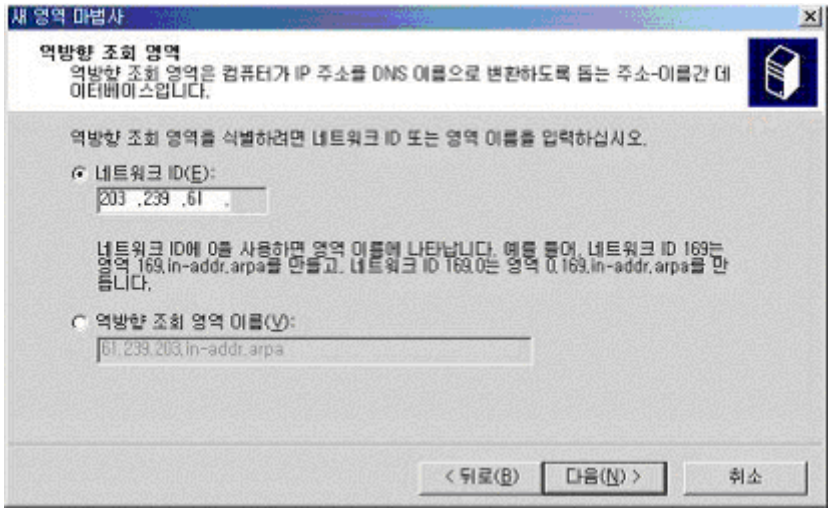
이어서 Reverse Lookup Zone 에 대해서도 구성을 해 보자. [화면 13]



[화면 13. Reverse Lookup Zone 만들기]

위에서 설명한 바와 같이 이번에는 문자도메인이 아니다. IP Address 로 구성된

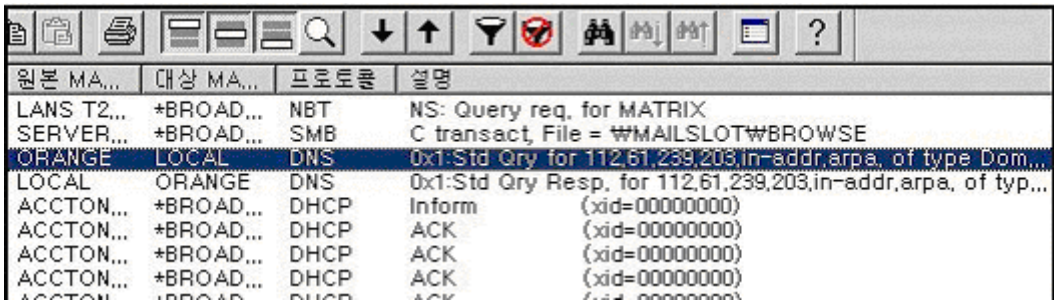
도메인을 만들어야 한다. 203.239.61.112 의 IP Address 는 203.239.61.0 이라는 "C"class 네트워크 네트워크 ID 에서 만들어진 호스트 ID 이기 때문에 네트워크 ID 부분을 입력한다. 입력하는 대로 아래부분에 있는 "역방향조회영역이름"을 살펴보면, "61.239.61.in-addr.arpa"라고 표기되는 것을 확인할 수 있다. 그것이 바로 생성될 Zone 의 이름이다.



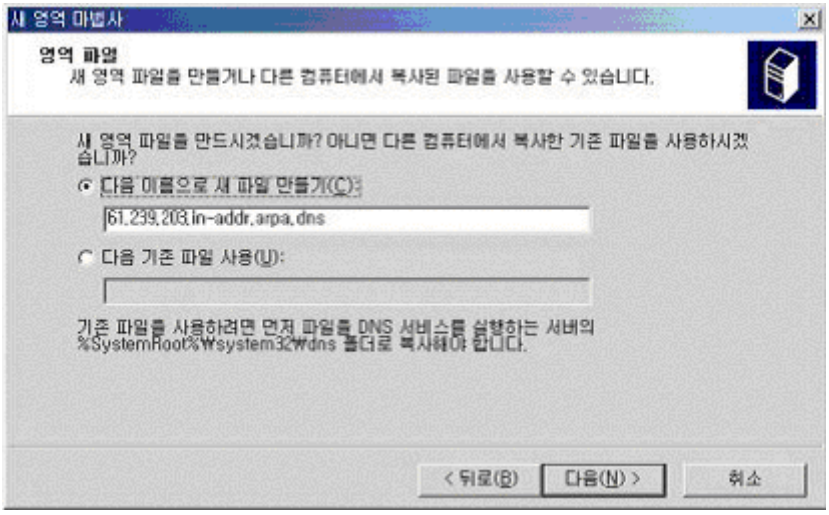
[화면 14. Reverse Lookup Zone 등록]

- 주요 팁 : Forward Lookup Zone / Inverse Query

  이 영역에 대해서는 이미 약속이 되어 있다. DNS 라는 것이 마이크로소프트만이 사용하는 것이 아닌 Internet 표준이기 때문에 모든 DNS 간에는 호환성이 유지되어야만 한다. 그래서 IP 주소를 이용해서 호스트이름을 확인하는 inverse query 를 발송할 때, "in-addr.arpa"도메인명을 이용하도록 약속된 것이다. 만일 외부의 DNS Server 가 203.239.61.112 를 사용하는 서버의 hostname 을 요청했다면 실제 요청은 "112.61.239.203.in-addr.arpa" 로서 날아오게 된다. 아래의 화면은 DNS Server 에서 Inverse Query 를 Windows2000 이 제공하는 Network Monitor Tool 을 사용하여 캡쳐한 화면이다. 진한부분을 보면 요청하는 이름을 확인할 수 있다. 도메인명은 "61.239.203.in-addr.arpa"가 되고, "112"는 forward lookup zone 의 레코드 형태로 하나의 호스트이름에 해당되는 내용이다.

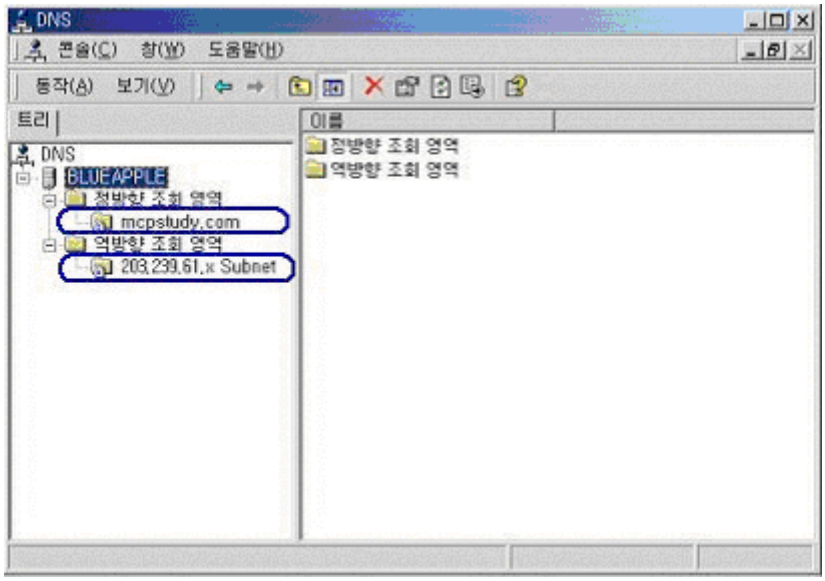


다음 과정은 역시 Reverse Lookup Zone 에 대한 데이터베이스를 저장할 파일이름을 결정해 준다. [화면 15]



[화면 15. Reverse Lookup Zone Database File 결정]

Reverse Lookup Zone 설치를 하고 나면 [화면 16]에서 보는 바와 같이 blueapple 이라는 DNS Server 는 2 개의 Zone 을 관리하는 DNS Server 로서 설치가 되었다.



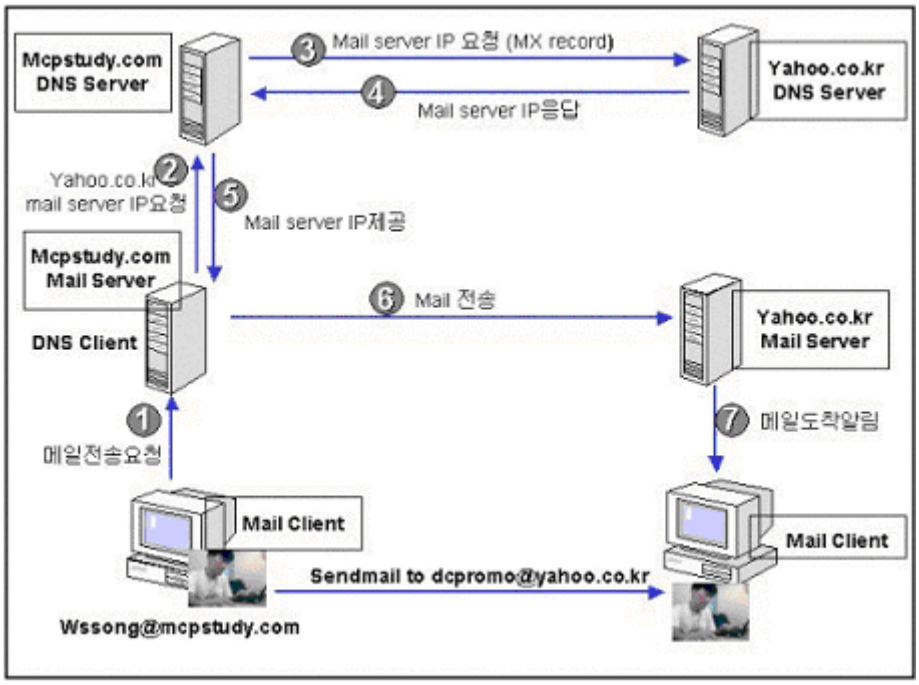
[화면 16. Reverse Lookup Zone Database File 결정]

지금까지 DNS Service 를 설치하고, DNS Server 가 관리할 Domain 을 구성하는 작업을 해 보았다. 막상 하고 나면 뭐가 이렇게 쉽나? 하는 생각이 든다.(돌 날아오는 소리가 들리는 것 같다 ^^;) 마이크로소프트의 Windows 기반의 OS 의 가장 큰 장점은 user interface 가 쉽다는데 있다. 관리자 입장에서도 마찬가지이다. 개념을 이해하는 과정이 조금 까다로울 뿐이지 막상 관리작업은 너무나 쉽다.

2. DNS 레코드 등록

DNS 의 기본구성을 했는데 그 다음엔 무엇을 해야 할까? 한가지 생각을 해 보자. 지금 DNS Server 를 구성하고 도메인을 추가하는 작업을 왜 한 것일까? 누구를 위해서?? 지금까지의 용도로 보자면 원격지의 클라이언트가 다른 DNS 서버를 통해서 mcpstudy.com 도메인에 속해있는 서버들의 IP Address 를 알 수 있도록 하기 위한 구성이었음을 알 수 있을 것이다. 그렇다면 당연히 추가작업을 생각할 수 있다. 바로, 웹서버, 메일서버, FTP 서버 등, 실제 외부에서 접근해야할 서버들을 도메인에 추가하는 작업을 해야 할 것이다. 도메인에 등록되는 이들 서버들의 정보를 "record"라고 부른다.

결국 해야할 작업은 관리할 도메인에서 "레코드"를 추가하는 작업을 해야 한다는 얘긴데, 여기에는 중요한 원칙이 한가지 들어있다. 아래의 그림을 보고 설명하도록 한다.



[그림 10. DNS 와 Mail Server]

[그림 10]에서는 wssong@mcpstudy.com 이라는 E-mail 을 쓰는 사용자가 dcpromo@yahoo.co.kr 에게 메일을 전송하는 과정에서 DNS Server 가 어떻게 동작하는지를 보여주고 있다. wssong 이라는 사용자로부터 메일전송요청을 받은 mcpstudy.com 의 메일서버는 yahoo.co.kr 의 메일서버의 IP 를 알아야만 메일을 전송할 수 있다. 이 때 메일서버의 IP 주소를 알아내기 위해서 DNS Client 로서 동작하게 된다. 자신의 DNS Server 로 설정된 DNS Server 에게 yahoo.co.kr 의 메일서버의 IP 를 요청하게 되고, 요청을 받은 DNS 서버는 iterative query 를 통하여 메일서버의 IP 를 알아오게 된다. 이 과정이 선행되지 않으면 메일서버들끼리 메일전송은 있을 수 없는 일이다.

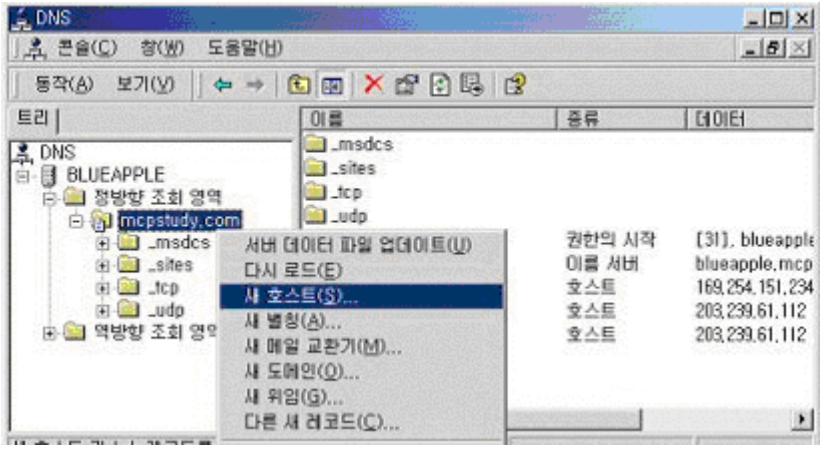
이때 DNS Server 가 상대방 mail server 의 IP 정보를 가지고 있는 DNS Server 에게 IP 요청을 할 때 어떤 형식으로 요청을 해야 할까? 무작정 "너네 회사의 메일서버가 누구인지 가르쳐줄래?"라고 우리가 쓰는 말처럼 물어볼 수는 없는 노릇이다. 역시 DNS Server 간에 약속된 어떤 표기법이 필요하다는 것이다. 이러한 이유로 DNS 는 DNS Server 에 등록되는 하나하나의 레코드 마다 별도의 "record type"을 정의해 둠으로써 호환성을 이루어 내고 있는 것이다. 그래서 메일서버는 메일서버(MX)로서, Name Server 는 Name Server(NS)로서 각각 하는 역할에맞는 record type 으로 등록이 되어 있어야 한다. [표 4.DNS Record Type 참고]

< DNS Record Type의 종류 >

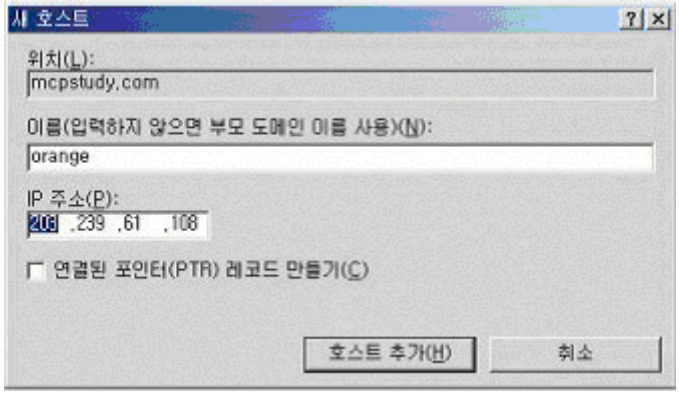
- **SOA (Start of Authority) :** Zone의 등록정보 제공
- **NS (Name Server) :** DNS Server 나열
- **A (Address) :** hostname -> IP Address
- **MX (Mail Exchange) :** Mail Server
- **CNAME (Chenonical Name) :** 별칭제공 (www,ftp등)
- **PTR (Pointer) :** IP Address -> hostname
- **SRV (Service) :** Service Resouce Record (ldap,kerberos등)
- 기타 (WINS, ISDN, WINS-R 등)

[표 4. DNS Record Type 의 종류]

DNS Server 를 통해서 이름분해가 되어야 할 호스트를 추가한다<화면 17>. 호스트를 추가할 도메인영역을 통해서 접근한다. 여러 가지 메뉴가 나온다. 어떤 레코드를 추가할 것인지에 따라 각각 다른 메뉴를 사용한다.

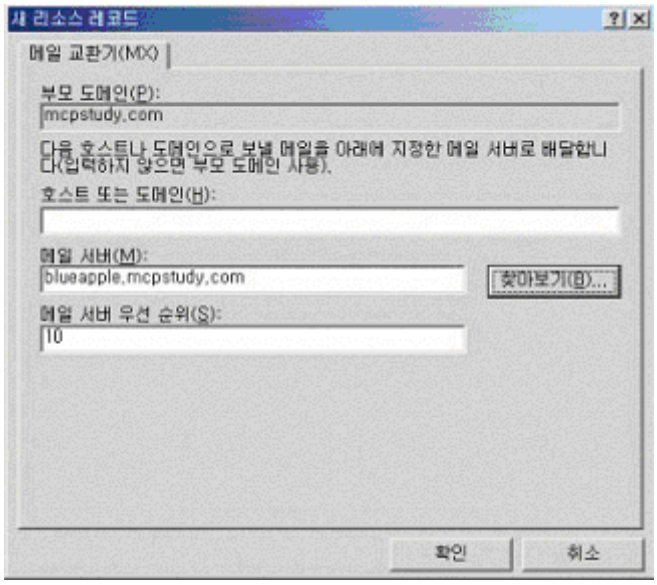


[화면 17. 새로운 레코드 추가메뉴]



[화면 18. 새호스트 추가작업]

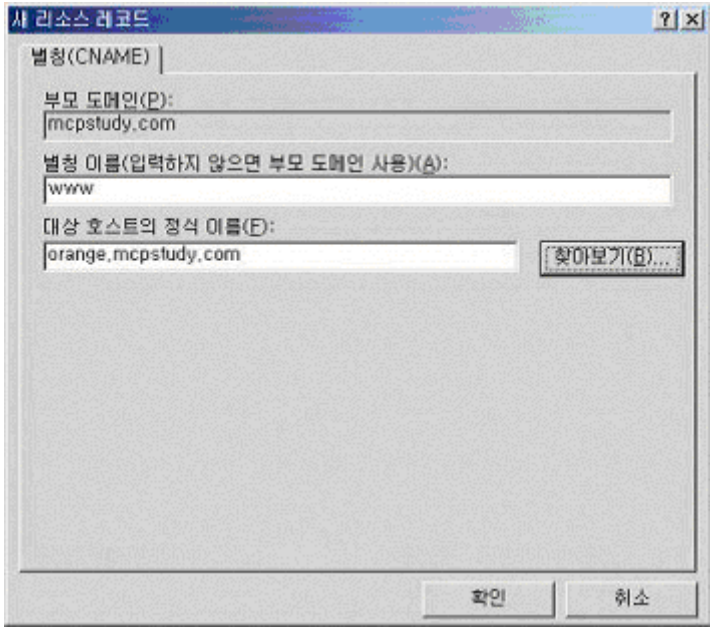
[화면 18]의 [이름]란에 호스트이름을 입력한다. 화면에서는 'orange'라고 입력하였다. 그렇다면 이 호스트의 FQDN 은 'orange.mcpstudy.com'이 된다. 아래에 있는 "연결된 포인터(PTR)레코드 만들기"체크박스가 있다. 기본적으로는 clear 된 상태이다. 체크를 하게 되면 호스트 레코드를 등록을 작업 한번만으로 동시에 Reverse Lookup Zone 에 PTR 레코드까지 생성해 주는 옵션이다. 이 작업이 제대로 되기 위해서는 먼저 reverse Lookup Zone 이 구성되어 있어야 한다. 그렇지 않다면 나중에 추가해 주면 된다



[화면 19. Mail Server Record 추가]

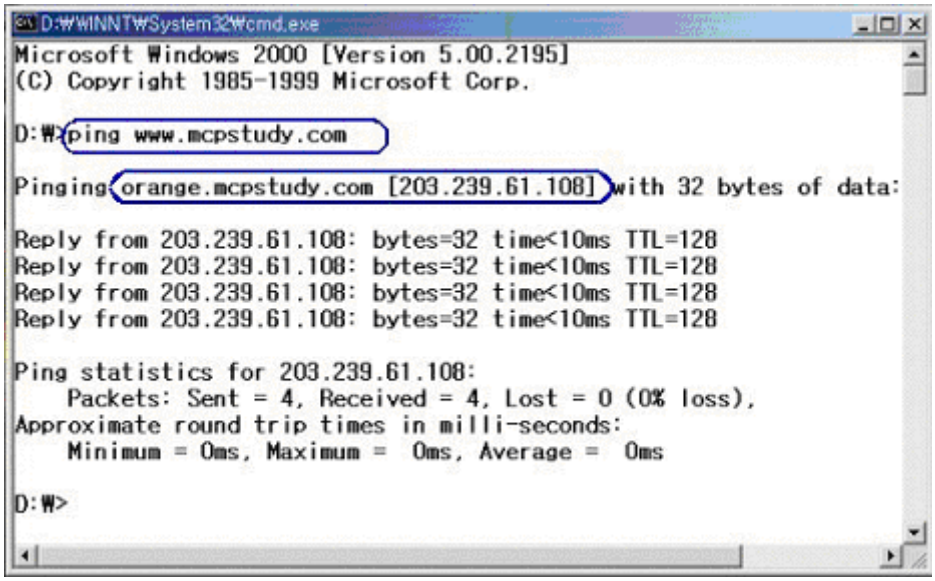
[화면 19]에서는 메일서버를 위한 레코드를 추가하고 있다. [찾아보기]버튼을 눌러서 메일서버의 호스트이름을 찾는다. 메일서버 우선순위는 '10'이 기본값이다.

메일서버가 여러대 있다면 우선순위를 별도로 할당해 줄 수 있다. 숫자가 낮을수록 우선순위는 높다. '호스트 또는 도메인'항목은 공백으로 두었다. 예제에서 사용하게 될 메일주소는 username@mcpstudy.com 형태이다. 만일 username@mail.mcpstudy.com 등의 형태를 사용하고 싶다면, 공백 대신에 'mail'이라고 기록해 준다.



[화면 20. CNAME 레코드 추가]

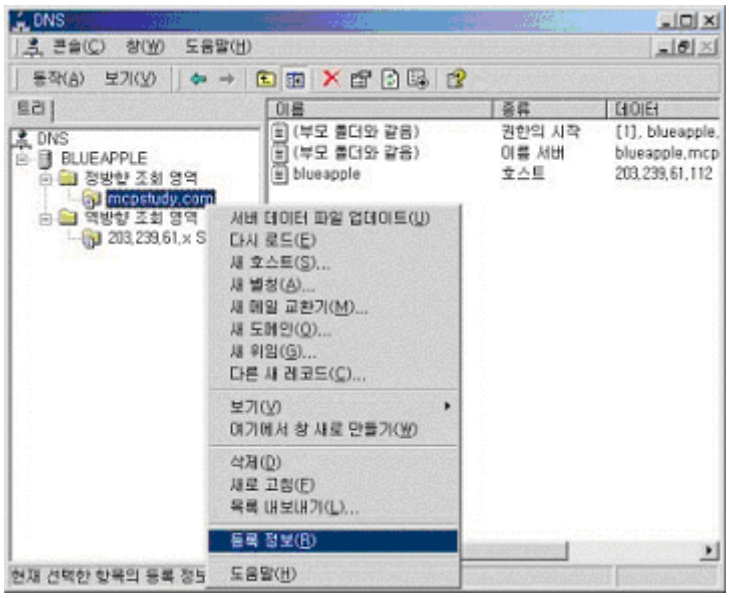
[화면 20]에서는 많이 사용하는 레코드중의 하나인 CNAME 을 등록하는 작업을 보여주고 있다. 일반적으로 흔히 볼 수 있는 www, ftp, pop3, mail, www2 등의 이름에서 사용한다. 한 컴퓨터가 orange.mcpstudy.com 이라는 호스트이름을 이미 가지고 있다면, 그 서버가 추가로 web, ftp 등 여러서비스를 하고 있을 때, 각각 호스트이름을 등록하는 것 대신에 별칭을 사용하는 것이 일반적이다. CNAME 레코드는 독립적으로는 동작할 수 없다. A 레코드와 한쌍으로 이루어져 있어야 한다. 아래의 <화면 21>에서 www.mcpstudy.com 에 ping 을 보냈지만, 실제로 응답하는 서버는 orange.mcpstudy.com 임을 주목하자.



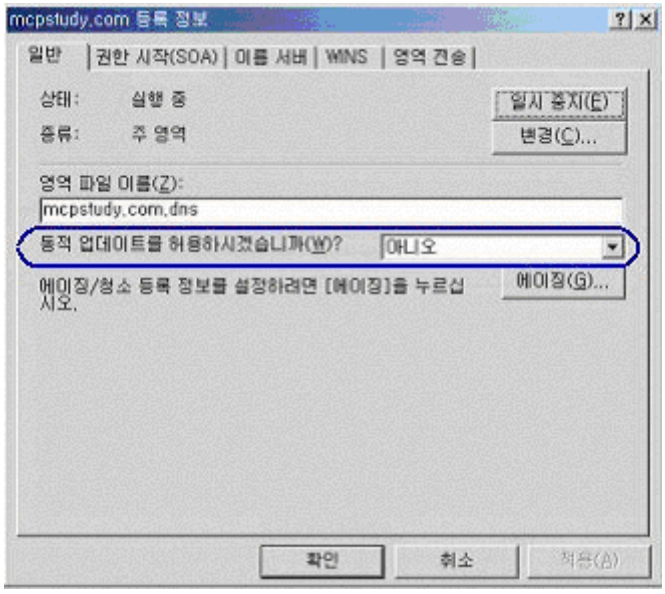
[화면 21. CNAME Record Ping test]

3. DNS Dynamic Update

작업을 하면서 답답함을 느낄 것이다. 회사에 등록해야 할 서버가 100 대라면 100 번을 반복해야 하는 것인가? 이런 번거로움을 Windows 2000 의 DNS Server 는 간단히 해결해 주고 있다. 바로 dynamic update 기능이다. Windows 2000 의 DNS 는 Dynamic Update Protocol 로써 레코드를 자동으로 업데이트하는 기능인 Dynamic DNS (DDNS)를 제공한다. 설정방법은 간단하다. 아래의 [화면 21,22]를 참고하면, 메뉴를 확인할 수 있다. "동적업데이트를 허용하시겠습니까?" 옵션을 "예"로 바꿔주면 동적업데이트가 진행된다. 관리자의 작업이 상당부분 경감되게 될 수 있다



[화면 21. DNS Zone 등록정보]



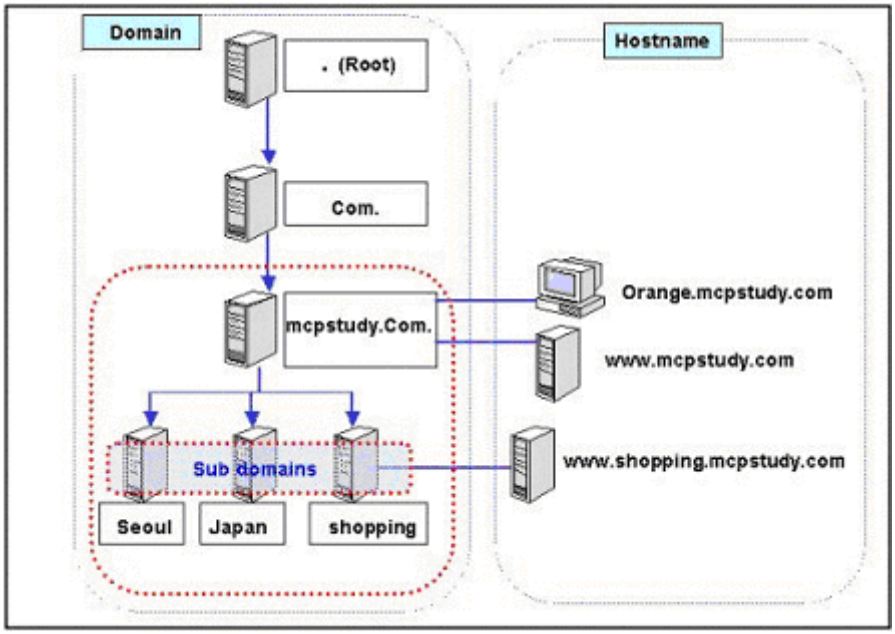
[화면 22. 동적업데이트 사용메뉴]

지금까지 도메인을 등록하고, 설치하고, Zone 을 구성하는 방법에 대해서 알아보았다. 대부분의 기업환경에서는 위와 같은 설정으로 DNS 를 마무리 할 수 있다. 하지만, 보다 복잡한 기업환경에서는 DNS 를 보다 확장시키는 방법을 사용한다. 자신의 하위도메인을 등록해서 도메인을 확장시켜서 사용할 수도 있다. 추가로 다루어보자. 지금까지의 설명이 조금 벅차게 생각되는 독자는 아래의 부분은 일단 건너뛰고 4 장으로 바로 넘어가도 좋다. DNS 에 대해서만 다룬 서적도

수백페이지 이상 되는 책이 부지기수다. 갈길은 멀다. 기초부터 튼튼히 하자는 얘기다.

4. 하위 도메인(Subdomain)

하위도메인이 무엇을 의미하는지 이해를 돕기 위해서 아래의 [그림 11]을 참고하자.



[그림 11. DNS Subdomain 의 구조]

[그림 11]에서는 mcpstudy.com 도메인을 예제로 DNS 구조를 간소화시켜 놓았다. 왼쪽편이 Domain 부분이고, 오른쪽편이 해당 도메인에 등록된 record 들이다. 그림이 다소 복잡해보이지만 차근차근 접근해 보자. mcpstudy.com 도메인은 com.도메인으로부터 mcpstudy 라는 이름을 등록받았고, 자기 자신은 seoul, japan, shopping 이라는 하위도메인들을 등록해 주었다. 이들 세 도메인들은 mcpstudy.com 의 하위도메인들이다. shopping 도메인 입장에서의 도메인이름은 "shopping.mcpstudy.com"이 된다. 그 도메인에서 'www'라는 이름의 웹서버를 가지고 있다면 오른쪽에 표기된대로 웹서버의 FQDN 은 www.shopping.mcpstudy.com 이 되야 한다.

이들 하위도메인을 등록하는데, "com."이나, "root"같은 상위도메인과 연관성이 있는가? 전혀 그럴필요가 없다. 하위도메인을 등록하는 작업은 오로지 mcpstudy.com 을 관리하는 DNS Server 에서만 생성되면 그만인 작업이다.

앞장에서 보았던 DNS Process 를 이해했다면 알 수 있는 사실이었다.

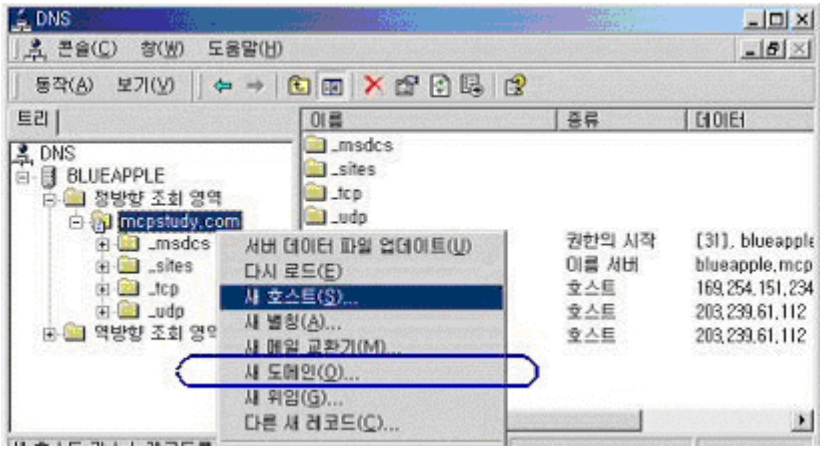
그렇다면 이러한 하위도메인은 언제 사용하게 될까? 회사가 각각 별도의 관리가 이루어지는 여러개의 관계회사를 가지고 있다고 가정하자. 하나의 도메인이름을 쓰는 것보다, 별도의 도메인이름을 사용함으로써 대외적으로 명확한 구분을 지어주고 싶을 수도 있을 것이다. 물론 경우에 따라서는 완전히 분리된 도메인을 사용할 수도 있다. 하지만 처음에 등록한 회사를 대표하는 이름이 있을 것이고 그 아래에서 분리시킴으로써 회사의 실제 조직구조에 맞게 도메인을 배치하는 작업이 가능해진다. 사실상 지금까지의 도메인 분리작업에 비추어 보자면 그다지 큰 쓰임새는 찾을 수가 없었다. 하지만, Windows 2000 의 Active Directory 도메인에서 DNS 의 도메인구조를 그대로 따르게 되면서 서브도메인은 Active Directory Multi domain 환경에서 반드시 필요해진 하나의 기능이 되었다.

삼성그룹을 예로 들어보자. 삼성은 "samsung.co.kr"이라는 도메인명을 가지고 있다. 관계회사중에는 우리가 익히 알고 있는 SDS, Unitel, 삼성전자 등 수많은 회사가 존재한다. 이들 관계회사들은 각각 독립된 도메인명을 가지고 있다. sds.co.kr, unitel.co.kr, sec.co.kr 등이 그것이다. 하지만 이것과는 별개로 또 다른 도메인명도 가지고 있는데 그것이 바로 samsung.co.kr 의 하위도메인명이다. sds.samsung.co.kr, unitel.samsung.co.kr, sec.samsung.co.kr 등을 의미한다. 왜 이렇게 하위도메인을 사용하는 것일까? 내부적인사정이겠지만, 대외적으로 samsung.co.kr 이라는 대표이름에 따른 하나의 일관성을 부여하기 위한 작업이었다는 생각이 든다. 반박하는 사람이 있을지 모르겠다. 이것은 오직 내 생각이니까 당신입장은 알아서 생각해라.

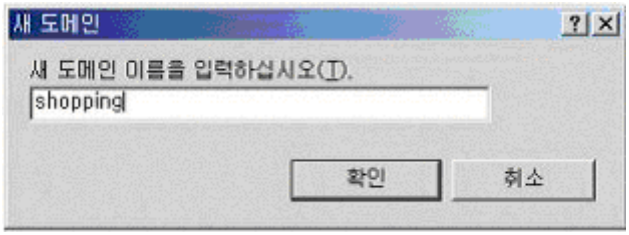
하위도메인에 필요성에 대해서 이야기했다. 그렇다면 하위도메인은 어떻게 만드는 것일까? 여기에는 관리방법에 따라 두가지 방법이 제공된다. mcpstudy.com 도메인이 본사, shopping.mcpstudy.com 이 지사라는 가정을 해 보겠다. 본사에서 지사의 도메인까지 관리를 해 주는 방법이 있겠고, 지사 역시 별도의 IT 조직이 있기 때문에도메인에 대한 독자적인 관리권한을 가지고 별도의 DNS Server 를 관리하는 경우도 있을 것이다.

5. 하위 도메인(Subdomain) 등록

전자의 경우는 아래의 [화면 23]의 메뉴를 통해서 접근한다.



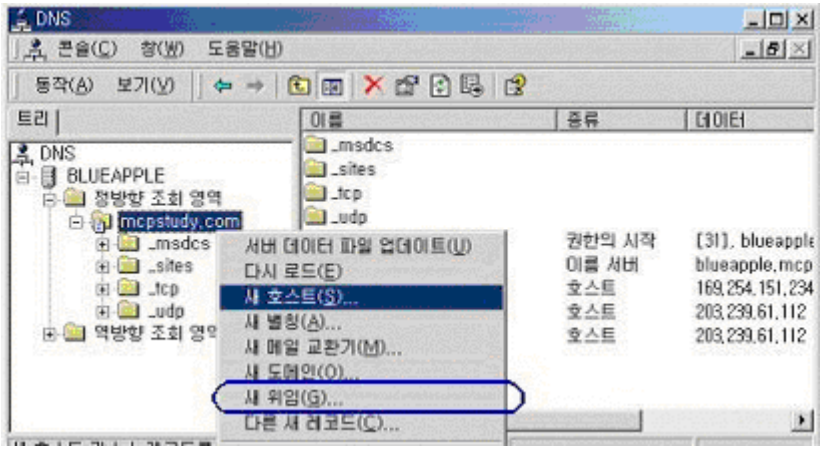
[화면 23. 새 도메인 추가]

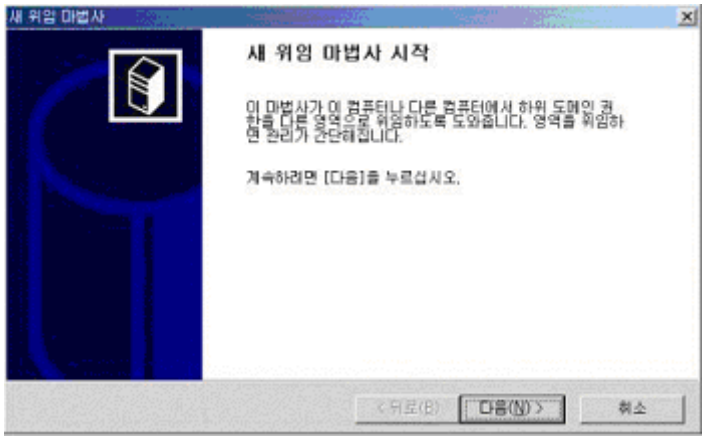


위의 화면에서, 생성하고 싶은 도메인명을 입력한다. 예제에서는 'shopping'을 입력하였다. 아래의 [화면 29. 완성된 DNS 예제]를 보면, DNS 콘솔의 MCPSTUDY.COM 밑에 "shopping"이라는 폴더를 찾아볼 수 있다. 하위도메인이 생성된 것이다. 이제 그 도메인에서 레코드를 생성한다면 생성되는 레코드들의 FQDN 은
"레코드명.shopping.mcpstudy.com"이 된다. 예를 들면,
www.shopping.mcpstudy.com 을 들 수 있다.

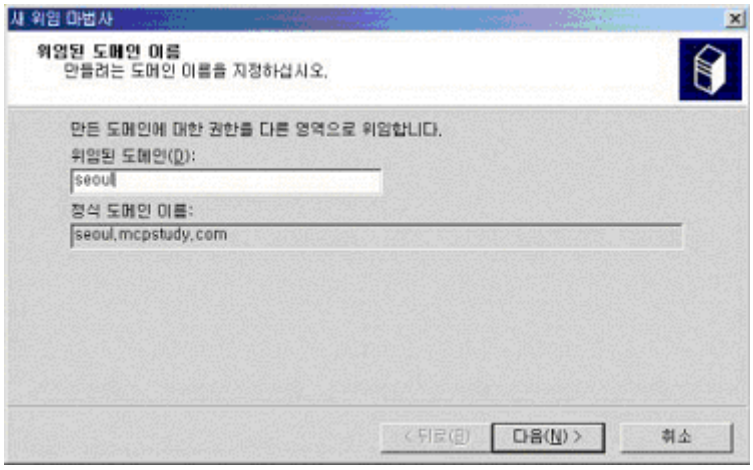
6. 하위 도메인(Subdomain) 위임(delegation)

두 번째 방법은 별도의 DNS Server 를 통해서 하위도메인을 관리하는 방법이 있다고 했다. 이때 상위도메인의 DNS Server 에서는 하위의 도메인을 생성하고, 그 도메인을 관리할 별도의 DNS Server 를 지정해 주어야 한다. 이 과정을 가리켜서 도메인위임(delegation)이라고 한다. 관리권한을 위임한다는 뜻이다.





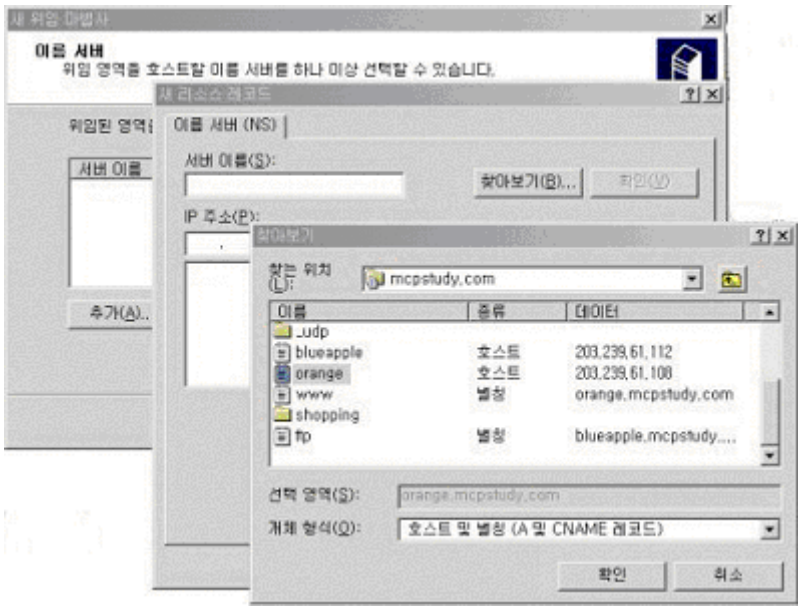
[화면 24. 새 위임 마법사 시작]



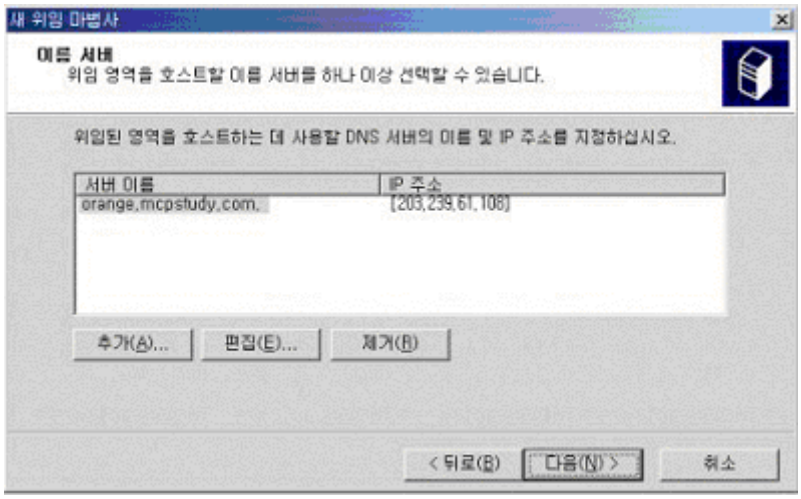
[화면 25. 위임할 도메인이름 지정]

[화면 25]에서 위임할 도메인 이름을 입력하면, 정식도메인이름이 바뀐다.

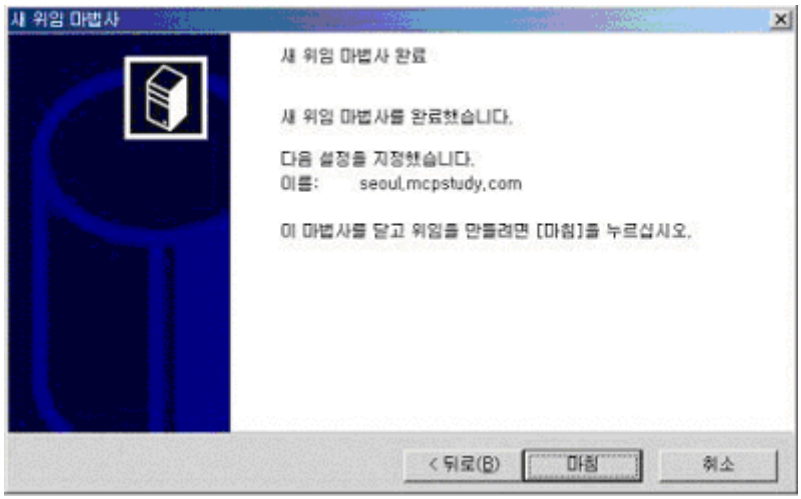
'seoul'이라고 입력했더니, [정식도메인이름]은 seoul.mcpstudy.com 이 되었다.



[화면 26. 위임해줄 DNS Server 지정]

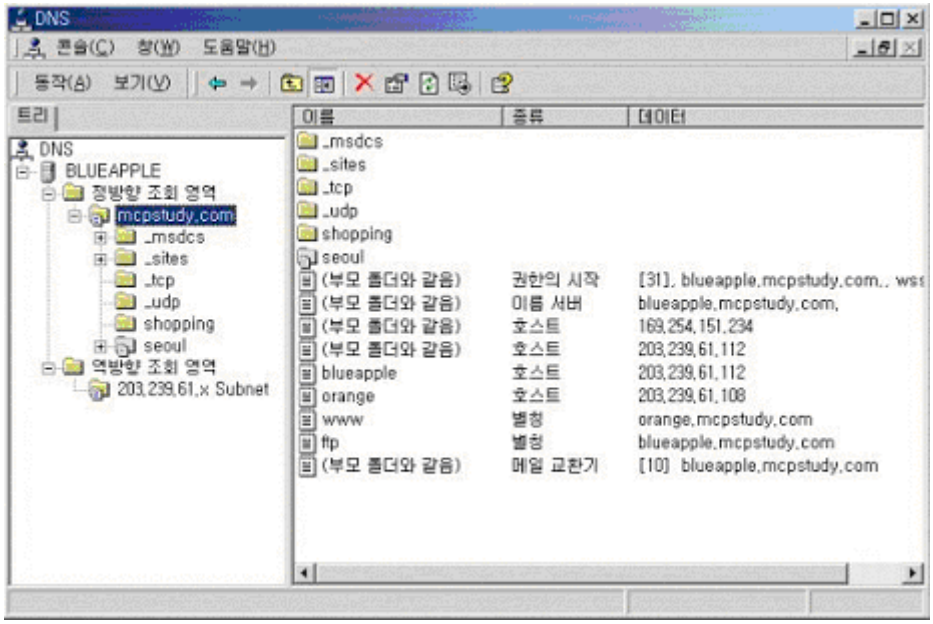


[화면 27. 하위 도메인을 관리할 DNS Server 가 지정된 화면]



[화면 28. 새 위임 마법사 완료]

이제 하위도메인을 위임시키는 작업을 완료하였다. 아래의 <화면 29>를 보면, mcpstudy.com 밑에 "seoul"이라는 폴더가 생성된 것을 확인할 수 있다. "shopping"과 폴더모양이 다른 것을 기억하자. shopping.mcpstudy.com 은 blueapple 이라는 DNS 서버가 직접 관리하고 있는 도메인이고, "seoul.mcpstudy.com"은 더 이상 blueapple 이 관리하는 도메인이 아니다. 위에서 지정해준 "orange"라는 이름의 DNS Server 가 관리하는 도메인이 된 것이다. 만일, 외부의 DNS Server 에서 "www.seoul.mcpstudy.com"이라는 호스트이름에 대한 IP 주소를 요청한다면, blueapple DNS Server 는 호스트이름에 대한 IP 주소를 응답할 수 없다. 다만, seoul.mcpstudy.com 도메인을 관리하는 name server 의 IP 주소만을 응답할 수 있을 뿐이다. 물론 그 다음과정이 정상적으로 처리되기 위해서는 위임받은 orange DNS Server 에서 서버구성을 해 두어야 할 것이다. 그 과정은 앞에서 우리가 다루어보았던 blueapple 이 mcpstudy.com 도메인을 구성하였던 것과 동일하다. 다만, 도메인 이름이 "seoul.mcpstudy.com"이라고 한단계 아래로 내려갔을 뿐이다. 그 이상의 차이는 없다.



[화면 29. 완성된 DNS 예제]

지금까지 Windows 2000 을 기준으로 DNS Service 를 추가하고, 서비스를 위한 기본구성을 마쳤다. NT4.0 을 가지고 작업하더라도 큰 차이는 없다. 단지 인터페이스 상에서 약간 차이를 보이고 있을 뿐이다.

4.윈도우 2000 DNS Server 관리

1. DNS Zone Transfer

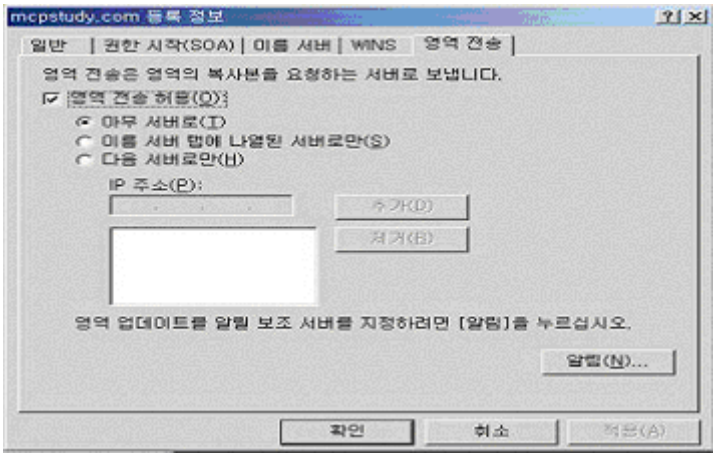
앞에서 DNS 서버 셋업에 관련된 작업을 다루어 보았다. 하나의 DNS Server 를 가지고 있는 회사에서 만일 그 서버가 망가졌다면 어떻게 될 것인가? 당연히 외부에서의 접근도, 내부에서 외부로의 연결도 하지 못하게 될 것이다. 또, 조직이 아주 커서 하나의 DNS Server 로써 모든 클라이언트의 요청을 효율적으로 처리하기가 어려운상황이 되었다면 어떻게 해야 할 것인가? 당연히 결론은 "추가 DNS Server"의 설치를 고려해야 할 것이다. 하지만, 한가지 고려되어야 할 점은 그 추가 DNS Server 는 반드시 첫 번째 DNS Server 와 동일한 레코드를 가지고 있어야 한다는 것이다. 그래야만 DNS Client 의 요청에 서로 다른 DNS Server 에서 일관성이 유지될 수 있을 것이기 때문이다.

그렇기 때문에, 회사에서 Backup 을 위해서 추가 DNS Server 를 셋업하는 것은 첫 번째 DNS Server 가 가지고 있던 원본을 복제하는 작업을 해야 하고, 당연히 첫 번째 서버가 정상적으로 동작하고 있는 상태에서 추가할 수 있다. 이때 백업서버를

가리켜서 Secondary DNS Server 라고 부르고, 그 서버가가지고 있는 도메인영역을 Secondary Zone 이라고 부른다.

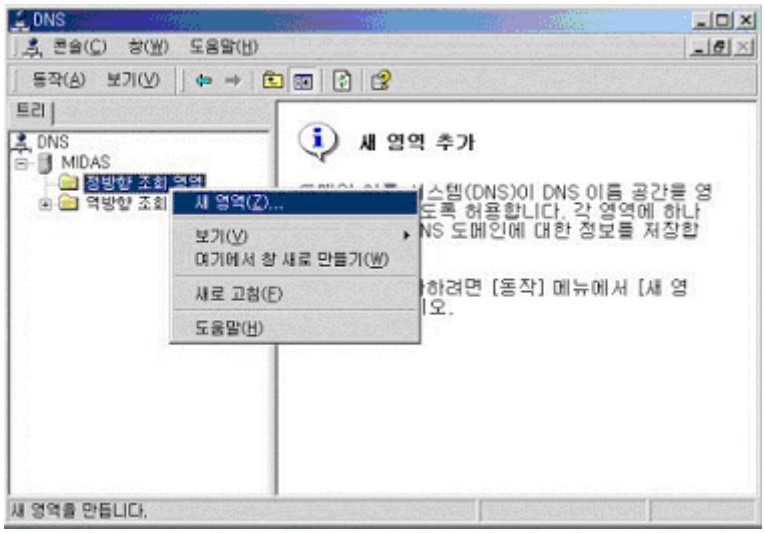
- Secondary DNS Server 설치

아래의 [화면 31]의 예제에서는 MIDAS 라는 이름의 서버에 DNS Service 를 추가하고, mcpstudy.com 도메인을 관리하는 Secondary DNS Server 가 되고자 한다. 이 과정이 성공적으로 진행되기 위해서는 원본을 가지고 있는 DNS Server 에서는 Zone database 를 전송해 줄 수 있도록 보안설정이 되어 있어야 한다. 기본적으로 아무 서버에서나 요청한다면 전송해 주도록 설정이 되어 있다. [화면 30]의 Zone 등록정보를 통해서 확인할 수 있다.



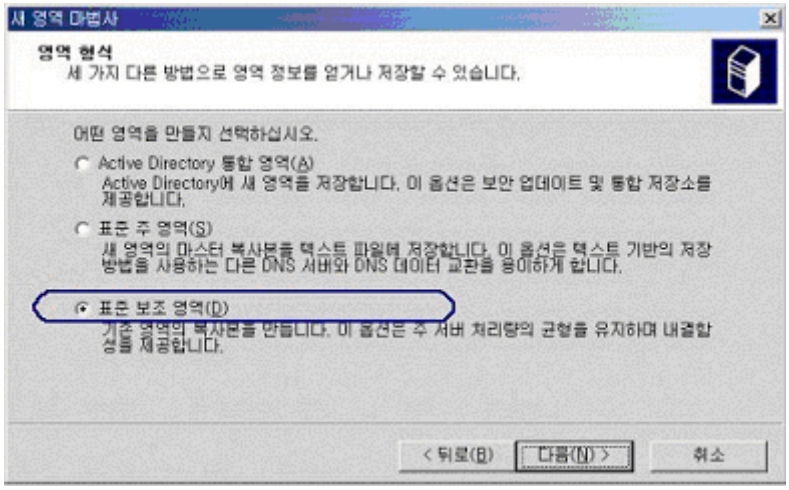
[화면 30. DNS Zone 등록정보의 "영역전송"탭]

준비가 되었으면 이제 Secondary DNS Server 를 설치해 보자. [화면 31]을 통해서 접근한다. 과정은 지난 장에서 소개한 것과 큰 차이가 없으므로 어렵지 않을 것이다.

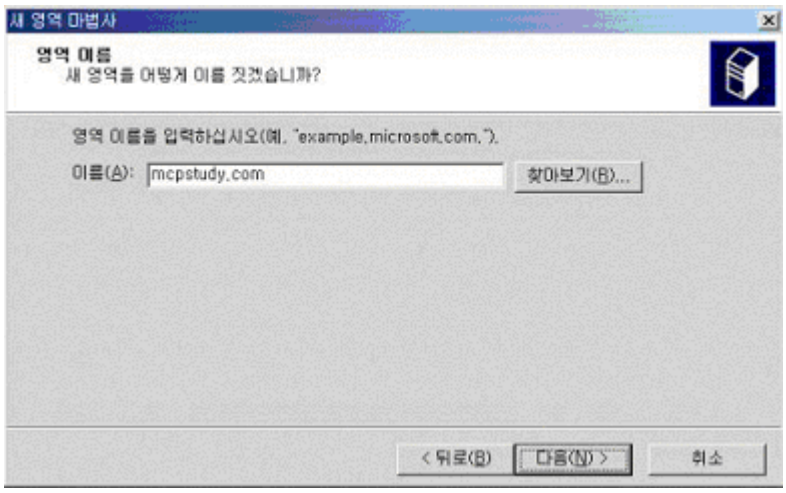


[화면 31. DNS Zone 추가]

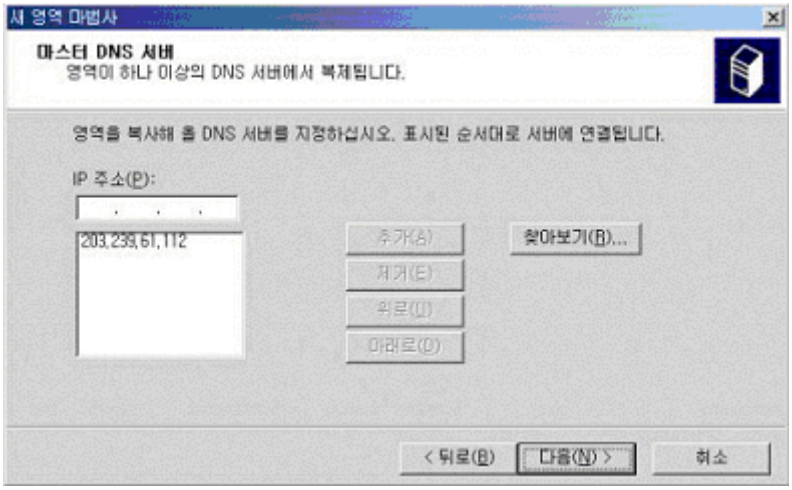
다음 화면에서 "표준 보조영역"을 선택해야 한다. Secondary DNS Server 를 추가하기 위한 과정이다.



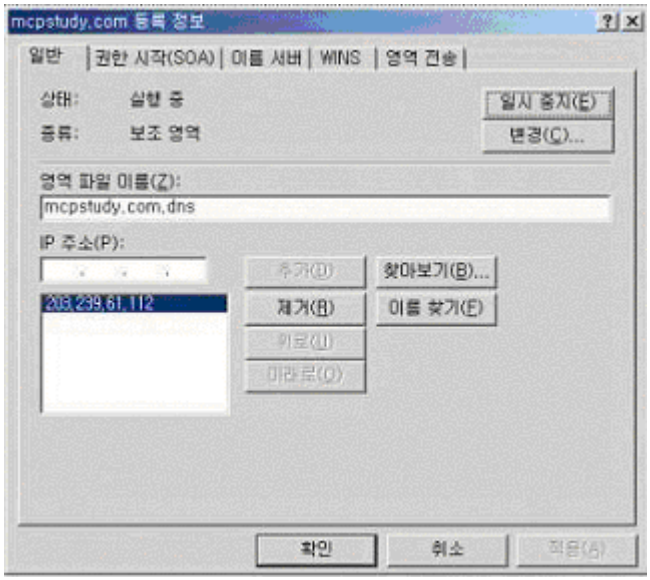
영역이름을 추가해야 한다. 현재 설치하고 있는 DNS Server 가 Primary DNS Server 의 DNS Client 로 되어 있다면 [찾아보기]를 통해서 접근할 수도 없다. 별 차이는 없다. 이름항목에 정확히 도메인명을 입력한다.



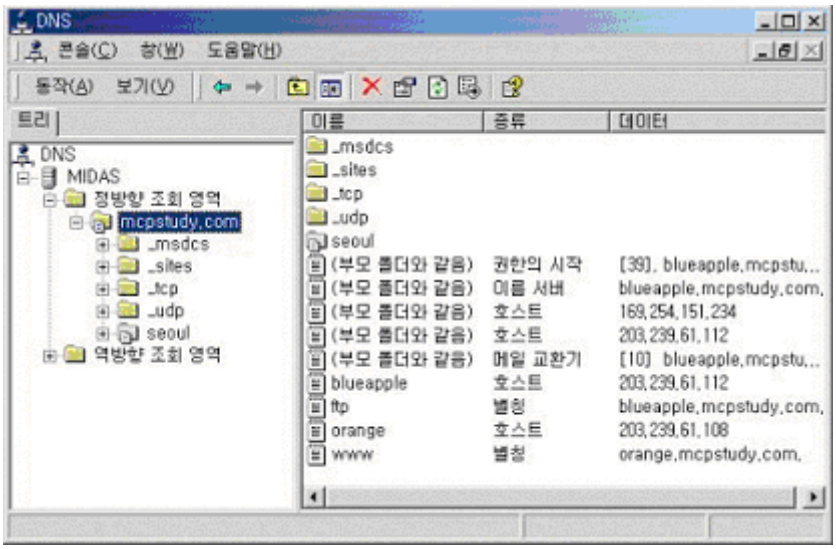
마스터 DNS 서버의 IP 를 주어야 한다. 마스터 DNS 서버는 원본을 제공해 줄 수 있는 DNS 서버를 의미한다. 먼저 설치했던 blueapple.mcpstudy.com 의 IP 주소를 입력했다.



역시 파일이 만들어진다. 기본설정을 그대로 두고, [확인]을 누른다.



이제 secondary DNS 의 설치가 완료되었다. 아래의 [화면 32]를 보면, 콘솔 오른쪽에 많은 정보가 전송되어 왔음을 확인할 수 있다. 이제부터 MIDAS 라는 이름의 DNS 서버는 mcpstudy.com 도메인의 Secondary Zone 을 가지고 서비스를 할 수 있게 되었다.

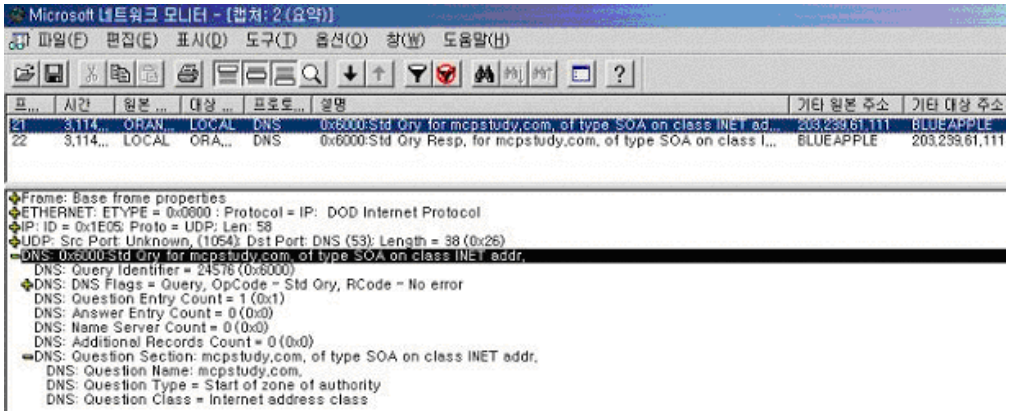


[화면 32. DNS Console]

○ Zone Transfer (영역전송)

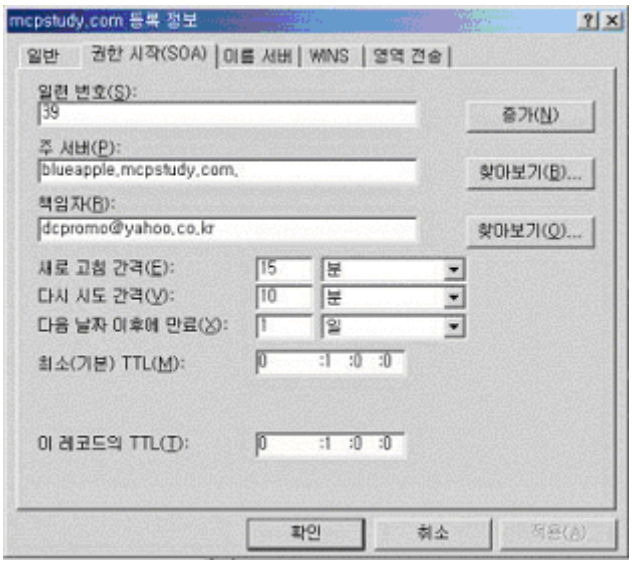
위의 과정에서 Zone transfer 라는 작업이 진행된다. Zone transfer 는 엄밀히 따지면 서버대 서버의 작업이 아니다. 서버의 Zone 과 Zone 간의 작업이라고 하는 것이 맞겠다. 영역전송과정은 자동으로 일어난다. 이 과정에 대한 설정은 "SOA"레코드로서 처리되고 있다.

Secondary DNS 가 master DNS 를 통해서 Zone transfer 를 하고자 할 때 처음 요청하는 레코드가 바로 이 SOA 레코드이다. <화면 34>에서 마이크로소프트의 네트워크모니터 툴을 사용해서 Zone transfer 과정을 캡춰한 내용을 보자.



[화면 33. DNS Zone transfer capture 화면]

화면의 파란색의 부분이 Secondary DNS Server 가 master DNS Server 에게 SOA 레코드를 요청하는 패킷이다. 바로 아래에는 master 로부터의 응답이 오고 있는 것이 보여진다. Secondary DNS 에서 master DNS Server 로부터 영역전송을 받기 전에 가장 먼저 SOA 레코드에 대한 요청이 이루어졌다. SOA 레코드에 어떤 정보가 담겨 있는지 확인해보자.[화면 34]

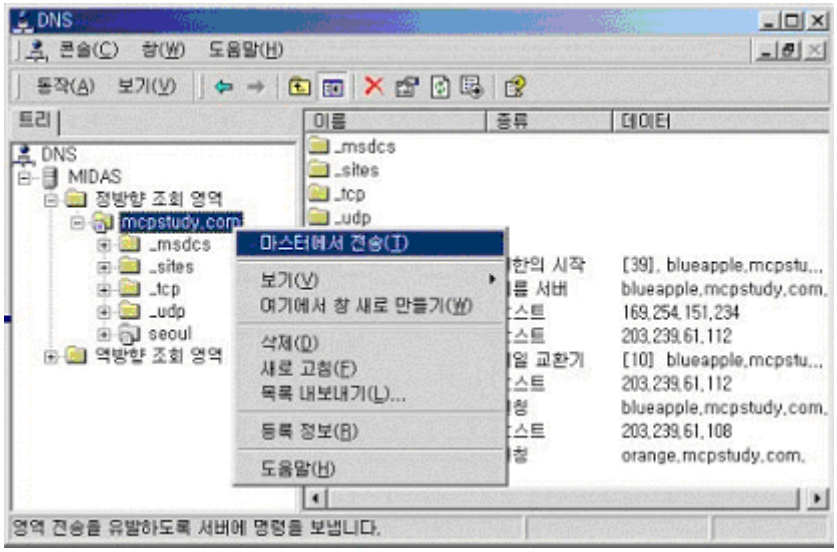


[화면 34. DNS Zone 등록정보 SOA Record 정보]

Secondary DNS Server 에서는 SOA 레코드를 받아서 "일련번호"정보를 알아야 한다. 화면에서는 일련번호로 '39'라는 숫자를 보여주고 있다. 일련번호가 의미하는 것은 DNS Zone 에 대한 변경사항의 번호이다. Zone 에서 레코드의 추가,삭제등의 변경사항이 생길 때마다 일련번호는 1 씩 증가하게 된다. Secondary DNS Server 는 일련번호를 비교해서 자신이 앞선 전송과정 이후에 바뀐 내용이 있는지 확인할 수가 있게 된다. 예를 들어서 저번 영역전송때 일련번호가 '20'이었다면 이번에는 21 부터 39 까지 바뀐 레코드에 대한 전송을 요청할 수 있게 된다.

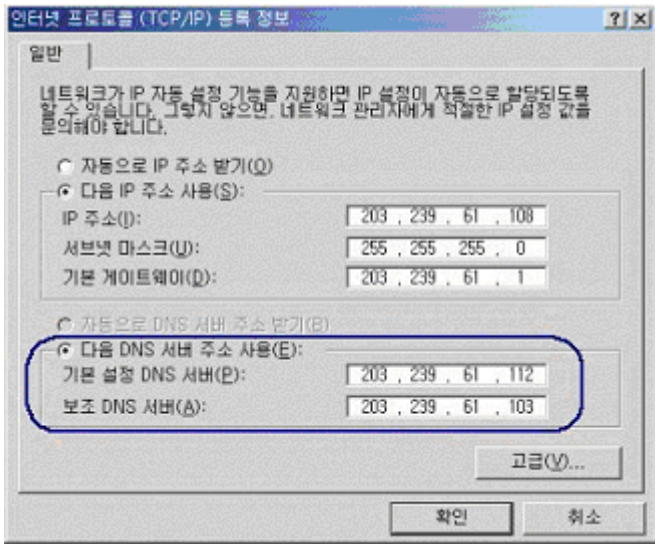
NT4.0 의 DNS 의 경우는 DNS Zone transfer 과정마다 전체 레코드에 대한 복제가 이루어졌다(Full Zone Transfer ; AXFR). 반면에 Windows 2000 DNS 는 기본적으로 바뀐 레코드만 복제하는 방법을 사용한다(Incremental Zone Transfer ; IXFR). 지극히 당연한 것처럼 보이지만, Windows2000 에서 개선된 점중의 하나이다.

Secondary DNS 서버에서는 자동적으로 SOA 레코드에 있는 정보중에서 "새로고침간격"마다 master DNS Server 에 접근하여 변경사항을 확인하게 되는데, 여러분이 직접 현재 상태를 반영시키는 업데이트를 강제로 진행하고 싶다면, 수동으로 처리하는 방법도 있다. [화면 35]



[화면 35. DNS Manual Zone Transfer]

Secondary DNS Server 를 구성하였다면 클라이언트 컴퓨터의 TCP/IP 설정도 조정해야 한다. 아래의 화면처럼 구성된다면 Primary DNS 가 offline 상태라도 정상적인 Name resolution 을 처리할 수가 있게 된다. 첫 번째 DNS 서버가 정상적으로 동작하고 있는 동안에는 보조 DNS 서버로 요청을 보낼 일이 없다. 어디까지나 백업 차원에서 제공되는 것이다.



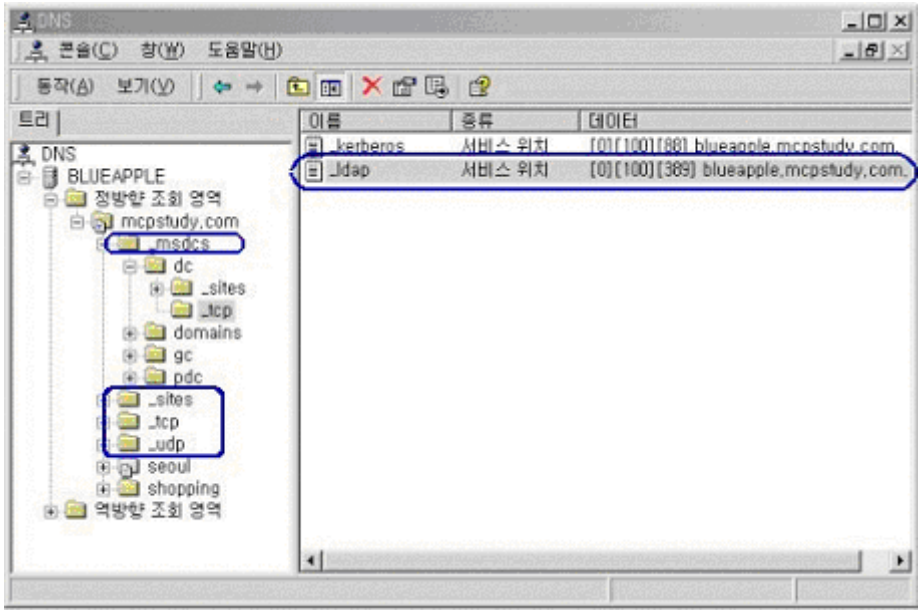
2. SRV Record

처음에 전반적인 개요를 설명할 때 WIndows2000 에서의 DNS 는 그 기능이 훨씬 확장되었다는 이야기를 했다. Windows2000 과 DNS 가 어떤 관계에 있는지를 알아보자. NT4.0 때만 하더라도 마이크로소프트 네트워크에서 DNS 의 영향은 그다지 크지 않았다. 보통의 경우라면 인터넷에 접근해야 하는 경우에만 사용하는 것이 일반적이었고, 유닉스네트워크와 혼합된 환경이라면 유닉스기반의 네트워크에 접근하기 위해서 이름서비스를 제공하는 것이 DNS 의 주된 기능이었다. Microsoft 자체의 도메인기반의 네트워크에서는 DNS 가 아닌 WINS 라고 하는 다른 형태의 이름분해를 처리하는 서비스가 주된 임무를 수행했다. 하지만, Windows2000 으로 오면서 WINS Server 가처리하던 서비스를 이제는 DNS 가 처리해야 한다. 당연히 클라이언트는 DNS 에게 query 를 던져서 원하는 정보를찾아야 하는 것이다.

마이크로소프트는 기업네트워크를 위해서 "도메인"이라는 이름의 작업그룹을 이용한다. MS 가 도메인모델을 통해서 추구하고자 하는 것은 "디렉토리서비스"를 통하여 도메인내의 모든 사용자들은 최초 한번의 로그온만으로 도메인내의 모든 서버에 접근을 용이하게 하겠다는 것이다. NT4.0 에서는 이러한 마이크로소프트의 "도메인"과 DNS 상에서의 "도메인"을 별개로 취급했었다. 그래서 회사환경의 "도메인"에 대한 이름분해 서비스로서 "WINS"라고 부르는 name server 를 두었고, 인터넷 DNS 상의 "도메인"에 대한 이름분해 서비스로서 "DNS"라는 name server 를

두고 이원화 되어 있는 구조를 사용했던 것이다. 하지만, Windows2000 으로 들어오면서 두 도메인을 따로 생각하지 않고, 통합시킨 구조를 사용한다. 엄밀히 말하면 통합이라고 표현하기에는 상당한 이해가 필요해진다. 결국 통합되었다는 표현은 다음과 같이 정리한다. 기업내부 네트워크를 구성하기 위한 도메인을 만드는데 있어서, 그 도메인이름을 인터넷상의 DNS Domain Name 을 사용하게 되는 것이다. 이렇듯 이름이 통합되었다는 얘기다. 결국 회사에 존재하는 사용자, 그룹, 컴퓨터 등 모든 정보들은 Active Directory 라고 부르는 디렉토리 데이터베이스에 저장하고, 그들을 가지고 있는 서버를 찾는 작업에 있어서 DNS 가 서비스를 제공하고 있는 것이다. 결국 DNS 가 하는 서비스는 Active Directory 위치를 찾아주는 "Locator Service"라고 할 수 있다.

그렇다면 Windows2000 의 DNS 에는 뭔가 큰 변화가 있었을 거라는 짐작이 간다. 하지만 실제로 들여다보면 그다지 큰 변화는 없다. 단지 "SRV"라고 하는 레코드 type 을 지원하고 있다는 것을 알 수 있다. 그 SRV 는 의미 그대로 "서비스를 하고 있는 서버"라는 뜻이 된다. [화면 36]



[화면 36. SRV Record]

[화면 36]에서 왼쪽에 보이는 4 개의 하위도메인들이 Domain Controller 가 자동으로 등록하는 도메인들이다. 오른쪽에 보이는 "_ldap"이라는 호스트이름이 "서비스위치"라는 type 으로 등록되어 있는 것을 볼 수 있다. 그것이 바로 "SRV"라는 레코드이다. 화면에서 보여지고 있는 정보는 "ldap"서비스를 하는 서버가 "blueapple.mcpstudy.com" 이라는 사실을 알려주고 있다. "ldap"서비스는

사용자 로그온을 처리하는 도메인컨트롤러가 제공하는 서비스이다. 사용자가 로그온을 할 때, 로그온을 받아주는 서버가 누구인지를 알고 있어야만 한다. Windows 2000 client 는 그 정보를 바로 DNS Server 에게 요청을 하게 되는데, 그때 요청은 결국 "ldcp server 가 누구인가?"라는 요청이 되고, DNS Server 는 등록된 레코드중 해당 서비스를 하는 서버로 등록된 레코드를 알려주게 되는 것이다. 이들 레코드에 대한 자세한 정보는 Active Directory 를 통해서 공부하도록 한다.

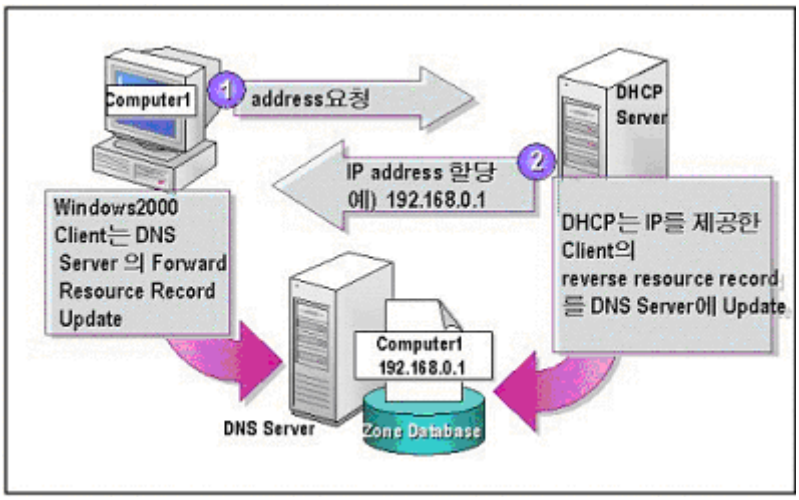
3. Dynamic update 사용

Windows2000 의 DNS Service 는 동적업데이트를 지원한다. 이것은 당연한 결과라고 보여진다. 앞에서 설명한 바와 같이 DNS 가 기업내부네트워크까지 적극적으로 반영하고 있기 때문에 NT4.0 에 비해서 상당히 많은 양의 정보를 제공해야만 한다. 이것을 예전 버전처럼 일일이 수동으로 작업을 한다면 상당히 번거로운 작업이된다. 이러한 문제점을 Windows 2000 DNS Server 는 dynamic update protocol 을 사용함으로써 관리작업을 단순화 시켜 주고 있다. Windows2000 관리자라면 당연히 사용해야 할 옵션중의 하나이다. 작업에 대한 과정은 이미 3 장에서 설명한 바 있다.

4. DHCP Service 와의 연동

만일 dynamic update protocol 을 통해서 등록되는 DNS Client 가 DHCP Client 라면 어떻게 될까? DHCP Client 는매번 다른 IP Address 를 사용할 수도 있는 유동적인 시스템이다. 이에 Windows 2000 DNS 는 이러한 DHCP Client 에게 IP 를 제공하는 역할을 하고 있는 DHCP Server 와 연동을 시키는 방법을 사용한다. 탁월한 방법이다

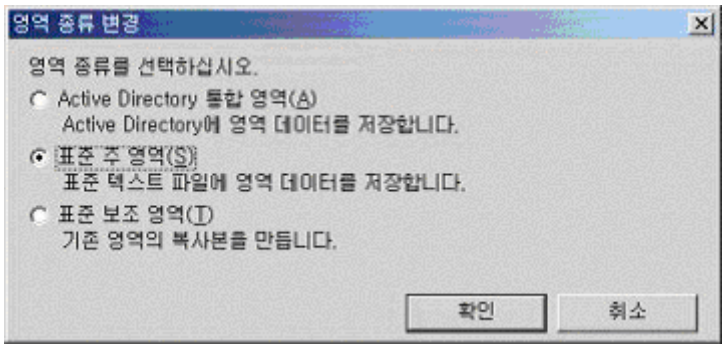
아래의 [그림 12]를 참조한다.



[그림 12. Dynamic update 동작원리]

5. Active Directory 통합영역 사용

DNS 를 설치하고, 관리할 도메인에 대한 Zone 을 추가하다 보면 영역의 종류를 결정하는 항목이 나온다. 아래의[화면 37]을 참고한다



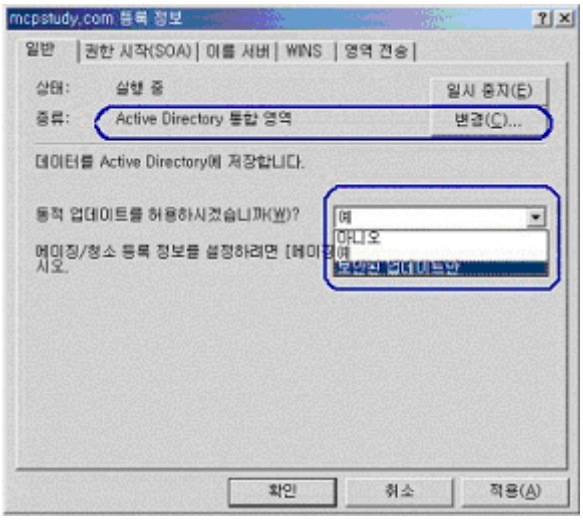
[화면 37. Zone type 변경]

'Active Directory 통합영역'이란 말 그대로 Windows2000 Domain Controller 가 가지고 있는 Active Directory 에 DNS Zone database 를 통합하겠다는 뜻이다. 그렇게 했을 때 얻는 이점이 무엇이 있을지 생각해 보자.

앞서서 Secondary DNS Server 의 필요성과 설정방법에 대해 설명하였다. 분명히 백업의 측면을 고려한 설정이긴 하지만 부족한 점이 있다. Primary DNS Server 는 DNS Zone database 원본을 가지고 있고, Secondary DNS Server 는 복사본을 가지고 있다. 실제로 DNS Zone database 변경에 영향을 줄 수 있는 서버는 오로지 Primary DNS Server 만이 가지고 있는 기능이다. Secondary DNS Server 는

수동적으로 Primary server 에 변경이 있으면 그 내용을 복제하는 역할을 하고 있을 뿐이다. 만일 Primary DNS Server 가 문제가 있어서 다운이 되었다면, Secondary DNS Server 는 DNS Client 에게는 정상적인 서비스를 할 수 있겠지만, DNS Zone database 의 변경작업은 더 이상 할 수가 없게 된다. 다시 말하면 새로운 레코드를 추가한다거나, 삭제, 레코드 수정 등의 작업은 하지 못한다는 뜻이다. 이런 문제를 해결할 수 있는 방법이 있다. 바로 'Active Directory 통합영역'을 사용하는 방법이다. 비록 Active Directory 를 가지고 있는 도메인컨트롤러에서만 구현되는 방법이긴 하지만, 어떤 DNS Server 든지 레코드 변경작업이 가능하다는 장점이 있다. 추가로 DNS Zone transfer 가 더 이상 필요가 없다. 도메인컨트롤러간의 Active Directory repilcation 작업에 DNS 복제프로세스가 통합되기 때문에 보다 유연한 복제토폴러지를 사용할 수 있게 되는 것이다. 또 한가지 장점이라면 Active Directory 통합영역을 사용하면승인받은 클라이언트만 DNS 에 접근하게 만드는 방법도 제공하기 때문에 보안의 측면에서도 보다 효과적인 방법이다.

'Active Directory 통합영역'으로의 변경작업은 DNS 콘솔의 해당 Zone 의 등록정보를 통해서 수정할 수 있다. [화면 38] 'Active Directory 통합영역'을 사용하면 동적업데이트 옵션에서 "보안된 업데이트만"을 사용할 수 있게 된다



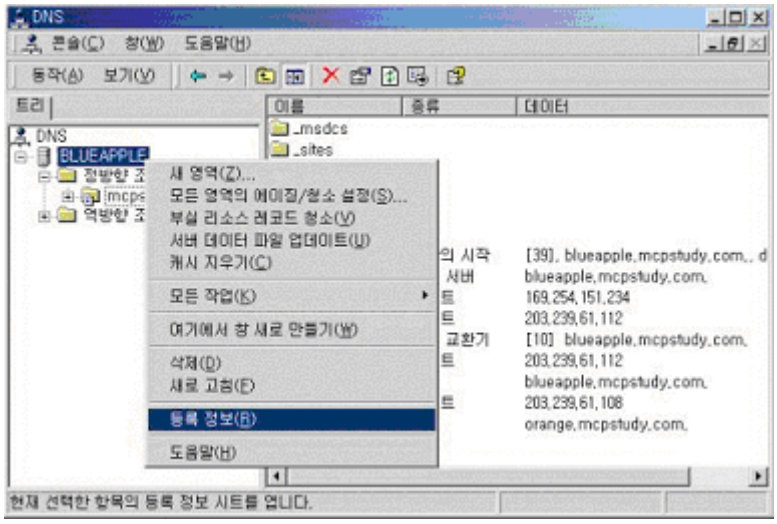
[화면 38. Secure Update]

지금까지 Windows 2000 DNS Service 의 몇가지 기능과 관리방법에 대해서 설명하였다

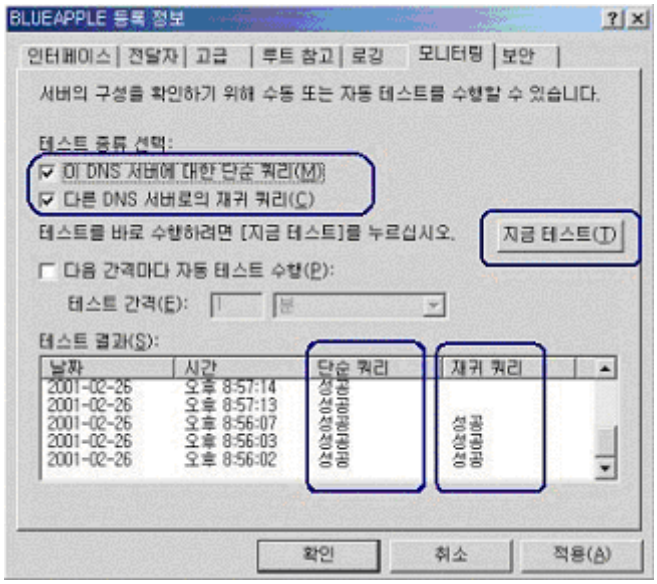
5.윈도우 2000 DNS 문제해결

1. DNS Server 모니터링 툴

DNS Server 가 시작되어 있는데 제대로 응답을 하지 못한다면, Windows2000 DNS Server 가 제공하는 모니터링 툴을 사용하여 상태를 점검해 볼 수 있다.
[화면 39]를 통해서 접근한다.



[화면 39. DNS Server 등록정보 메뉴]



[화면 40. DNS Server Monitoring Tool]

[화면 40]의 '테스트종류선택'에서 '이 DNS 서버에 대한 단순쿼리'를 선택하고 [지금테스트]버튼을 누르면
클라이언트의 요청에 응답할 수 있는지를 파악할 수 있다. 화면에서는 '성공'이라는

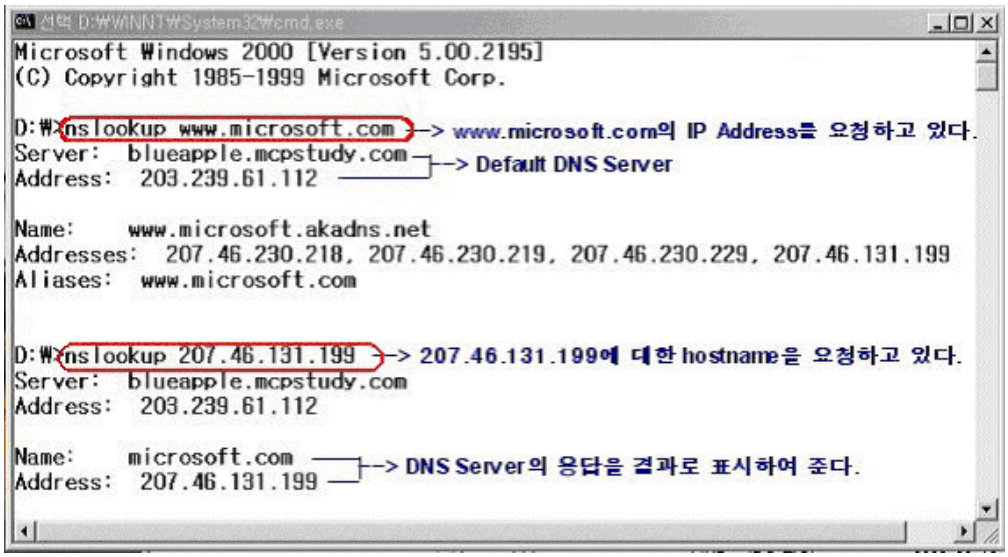
결과를 보여준다.
'다른 DNS 서버로의 재귀쿼리'를 선택하고 [지금테스트]버튼을 누르면, 해당 DNS Server 가 외부로 root name server 로
query 를 보낼 수 있는지를 파악할 수 있다. 역시 '성공'이라는 결과를 보여주고 있다.
문제가 발생했다면, [관리도구-이벤트표시기-DNS Server] 항목을 통해서 문제점을 볼 수 있다.
DNS Server 오동작의 상당부분은 DNS Service restart 를 통해서 해결된다.
명령프롬프트를 열고 "net stop dns"를 실행하고,다시 "net start dns"를 입력하면 DNS Server Service 를 재시작할 수 있다.

2. NSLOOKUP

DNS 를 진단하는데 있어서 대표적인 유틸리티라고 할 수 있다. NSLOOKUP 을 사용하는 방법으로는 크게 2 가지를 들 수 있다.
Interactive(대화형)와 non-interactive 로 나누어 사용해 보자.

- Non-Interactive

아래의 [화면 41]을 보면 사용법을 알 수 있다. hostname 이든 IP Address 이든 원하는 정보를 그대로 물어보는 가장 간단한 사용법이다.
단순한 정보를 알아보는 데 효과적인 방법이라고 하겠다



[화면 41. NSLOOKUP 사용예제 - Non-interactive]

o Interactive

대화형으로 접근하여 보다 자세한 레코드를 요청하고, 어느 서버가 문제가 있는 것인지를 진단해 볼 수 있는 방법이다.
아래의 표를 예제로 설명을 하겠다. 역시 명령프롬프트를 통해서 접근한다.

(참고: 진한색으로 표기된 부분이 직접 입력한 부분이며, 그밖은 명령에 의한 결과값이다. 파란색으로 표기한 부분이 필자의 부연설명이다.)

D:₩>nslookup --> 대화형으로 접근하기 위한 명령이다.
Default Server: blueapple.mcpstudy.com
Address: 203.239.61.112

> www.mcpworld.com --> www.mcpworld.com 에 대한 IP 를 요청했다.
Server: blueapple.mcpstudy.com
Address: 203.239.61.112

Non-authoritative answer:
Name: www.mcpworld.com
Address: 211.200.29.242

> 211.200.29.242 --> 211.200.29.242 에 대한 hostname 을 요청했다.
Server: blueapple.mcpstudy.com
Address: 203.239.61.112

DNS request timed out.
timeout was 2 seconds.
*** Request to blueapple.mcpstudy.com timed-out --> resolution 을 하지 못했다. 이것만으로는 정확히
원인을 규명할 순 없다.
mcpworld.com 의 DNS Server 가 Reverse Lookup Zone 을 구성하지 않았을 수 있다.

> set type=soa --> 특별히 원하는 레코드 타입을 미리 정의한다.
> microsoft.com --> microsoft.com 도메인의 SOA 레코드를 요청하고 있다.
Server: blueapple.mcpstudy.com
Address: 203.239.61.112

microsoft.com
primary name server = dns.cp.msft.net
responsible mail addr = msnhst.microsoft.com
serial = 2001022401
refresh = 900 (15 mins)
retry = 600 (10 mins)
expire = 7200000 (83 days 8 hours)
default TTL = 7200 (2 hours)
dns.cp.msft.net internet address = 207.46.138.10
> set type=mx --> Mail Server 의 레코드를 확인하고 싶다.
> microsoft.com --> microsoft.com 도메인의 MX 레코드를 요청하였다.
Server: blueapple.mcpstudy.com
Address: 203.239.61.112

microsoft.com MX preference = 10, mail exchanger = mail1.microsoft.com
microsoft.com MX preference = 10, mail exchanger = mail2.microsoft.com
microsoft.com MX preference = 10, mail exchanger = mail3.microsoft.com
microsoft.com MX preference = 10, mail exchanger = mail4.microsoft.com
microsoft.com MX preference = 10, mail exchanger = mail5.microsoft.com
mail1.microsoft.com internet address = 131.107.3.125
mail2.microsoft.com internet address = 131.107.3.124
mail3.microsoft.com internet address = 131.107.3.123
mail4.microsoft.com internet address = 131.107.3.122
mail5.microsoft.com internet address = 131.107.3.121

> set type=all --> 도메인에 대한 모든 레코드를 요청하길 원한다.
> mcpworld.com --> mcpworld.com 도메인의 레코드를 요청했다.
Server: blueapple.mcpstudy.com
Address: 203.239.61.112

mcpworld.com internet address = 211.200.29.242

mcpworld.com MX preference = 10, mail exchanger = tdc-srv1.mcpworld.com
mcpworld.com nameserver = ns.ngene.net
mcpworld.com nameserver = ns2.ngene.net
mcpworld.com
primary name server = ns.ngene.net
responsible mail addr = ngene.hanaro.com
serial = 2001021702
refresh = 21600 (6 hours)
retry = 1800 (30 mins)
expire = 1209600 (14 days)
default TTL = 21600 (6 hours)
mcpworld.com nameserver = ns.ngene.net
mcpworld.com nameserver = ns2.ngene.net
tdc-srv1.mcpworld.com internet address = 211.116.92.5
ns.ngene.net internet address = 211.58.252.62
ns2.ngene.net internet address = 211.58.252.94

> www.dotnetkorea.org --> 이 질의를 통하여 한가지 확인을 해 보자.
Server: blueapple.mcpstudy.com
Address: 203.239.61.112

Name: seoul.dotnetkorea.org
Address: 203.239.61.102
Aliases: www.dotnetkorea.org

> www.dotnetkorea.org --> 첫 번째 질문과 비교해서 차이점을 찾아본다.
Server: blueapple.mcpstudy.com
Address: 203.239.61.112
Non-authoritative answer: --> 첫 번째 요청에선 없었던 메시지다. 차이는 cache 에 있다.
DNS Server 는 Iterative Query 를 통해서 다른 DNS 서버로부터 얻은 정보를 레코드에 주어진 TTL 동안 보관한다.
그 TTL 이 만료되기 전까지는 Iterative Query 가 아닌 로컬캐쉬에서 정보를 찾아서 클라이언트에게 응답한다.
그런 이유 때문에 신뢰할 수 없는 응답이라는 표기를 하고 있다.

첫 번째 레코드를 받아오고 나서 실제 레코드정보는 변경이 되었을 수도 있기 때문이다.
Name: seoul.dotnetkorea.org
Address: 203.239.61.102
Aliases: www.dotnetkorea.org

> www.yahoo.co.kr
Server: blueapple.mcpstudy.com
Address: 203.239.61.112

*** blueapple.mcpstudy.com can't find www.yahoo.co.kr: No response from server --> 서버로부터 응답이 없다는 에러메시지가 발생했다.
서버의 DNS 서비스를 확인해 봐야 한다.
> www.yahoo.co.kr 164.124.101.31 --> www.yahoo.co.kr 레코드에 대한 요청을 '164.124.101.31' DNS Server 에게 물어보았다.
Server: [164.124.101.31] --> 응답하는 서버의 IP 가 바뀐 것을 주목하자.
Address: 164.124.101.31

Name: www.yahoo.co.kr
Served by:
- NS.KORNET.NET
168.126.63.1
kr
- NS.KRNIC.NET
202.30.50.50
kr
- NS.RIPE.NET
193.0.0.193
kr

> server 203.248.240.31 --> 아예 default server 를 수정하는 명령이다.
앞으로의 query 를 자신의 DNS Server 로 설정된 서버가 아닌 다른 서버에게 DNS 요청을 하겠다는 뜻이다.

Default Server: [203.248.240.31]
Address: 203.248.240.31

> www.yahoo.co.kr
Server: [203.248.240.31] --> default server 의 IP 가 바뀐 것을 주목하자.
Address: 203.248.240.31

Name: www.yahoo.co.kr
Address: 211.32.119.135

> ls mcpstudy.com --> mcpstudy.com 도메인의 모든 record 를 요청한다. 이 기능은 문제해결을 할 때나, 호스트의 이름이 기억이 나지 않을 때 등에 사용할 수 있다.
[blueapple.mcpstudy.com]
mcpstudy.com. A 169.254.151.234
mcpstudy.com. A 203.239.61.112
mcpstudy.com. NS server = blueapple.mcpstudy.com
gc._msdcs A 169.254.151.234
gc._msdcs A 203.239.61.112
blueapple A 203.239.61.112
orange A 203.239.61.108
seoul NS server = orange.mcpstudy.com

Ipconfig/flushdns

Windows 2000 은 클라이언트에서도 resolution 해온 DNS 레코드의 정보를 cache 에 저장한다. 간혹 이것 때문에 문제를 일으킬 수도 있다.

잘못된 정보를 제공받은 클라이언트가 계속 잘못된 정보를 사용하게 됨으로써 정상적인 이름분해를 하지 못하는 상황이 있다.
이런 경우 클라이언트의 cache 를 비워주는 명령어를 사용하면 된다. 아래의 표를 참고하자

```
D:₩>ipconfig/displaydns   --> cache에 보관된 DNS의 정보를 보여달라는 명령
 Windows 2000 IP Configuration
    localhost.
   ----------------------------------------------------------
     Record Name . . . . . : localhost
     Record Type . . . . . : 1
     Time To Live   . . . . : 31517302
     Data Length . . . . . : 4
     Section . . . . . . . : Answer
     A (Host) Record . . . :
                         127.0.0.1

   www.fila.co.kr.
   ----------------------------------------------------------
     Record Name . . . . . : www.fila.co.kr
     Record Type . . . . . : 1
     Time To Live   . . . . : 73002
     Data Length . . . . . : 4
     Section . . . . . . . : Answer
     A (Host) Record . . . :
                         210.124.177.198

     Record Name . . . . . : fila.co.kr
     Record Type . . . . . : 2
     Time To Live   . . . . : 73002
     Data Length . . . . . : 4
     Section . . . . . . . : Authority
     NS Record     . . . . . :
                         ns.fila.co.kr

     Record Name . . . . . : ns.fila.co.kr
     Record Type . . . . . : 1
     Time To Live   . . . . : 73002
     Data Length . . . . . : 4
     Section . . . . . . . : Additional
     A (Host) Record . . . :
```

211.106.109.2

ns2.dacom.co.kr.

Record Name . . . . . : ns2.dacom.co.kr
Record Type . . . . . : 1
Time To Live . . . . : 33263
Data Length . . . . . : 4
Section . . . . . : Answer
A (Host) Record . . . :
203.248.240.31

D:₩>ipconfig/flushdns
Windows 2000 IP Configuration
Successfully flushed the DNS Resolver Cache.

지금까지 DNS 에 대한 간단한 진단도구 사용법을 설명하였다.
처음 시작부터 5 장까지 DNS 에 대한 주요사항들을 위주로 설명을 했는데, 부족함이 많다.

Windows2000 환경에서 Active Directory 와 DNS Server 가 상당히 밀접한 관계가 있음에도 불구하고 DNS 만을 위주로 설명을 하려다 보니 Active Directory 와의 연동부분이 까다롭게 느껴질 수도 있겠다는 생각이 든다.

하지만, 이 내용만으로도 기본개념을 잡는 것에는 크게 어려움이 없을 거라고 생각하며,
DNS 를 배우고자 하는 학생들이 이 자료를 통해서 보다 쉽게 앞으로의 방향을 잡아 갈 수 있다면 하는 바램이다.