

BIND 8 ile DNS NASIL

Yazan:
İlker Temir
<itemir (at) cisco.com>

Özet

Bu belgede DNS hakkında bir ön bilgi verilmiş ve Internet'e bağlı bir DNS sunucusunun BIND 8 ile nasıl yapılandırılacağı ve sorgulanacağı örneklenerek anlatılmıştır.

Konu Başlıkları

1. DNS	3
1.1. DNS'nin Tarihçesi	3
1.2. DNS'nin Yapısı	3
1.3. DNS Çözümlemesi	5
1.4. IP Adresinden İsim Çözümlemesi	6
1.5. DNS Önbelleği	7
2. BIND	8
2.1. BIND Paketinin Temini	8
2.2. BIND'in Kurulumu	8
2.3. BIND Yapılandırması	9
2.4. BIND'in Çalıştırılması	14
2.5. DNS Sorguları – NSLOOKUP	15
2.6. DNS Kayıtları	17
3. Yasal Açıklamalar	19
3.1. Telif Hakkı ve Lisans	19
3.2. Feragatname	19

Geçmiş

1.0

Nisan 2000

İlker Temir itemir@cisco.com

İlk sürüm – Belgenin özgün sürümü <http://www.ilkertemir.com/document/dns.html>^(B1) adresinde bulunabilir.

1. DNS

Bilindiği üzere bir IP ağı üzerinde çeşitli servislere ulaşmak için bu servislerin çalıştığı sistemlerin IP adresi adı verilen ve 4 sayıdan oluşan akılda kalması zor ve son kullanıcılar için hemen hiçbir şey ifade etmeyen adreslerinin bilinmesi gerekmektedir. DNS servisi bu sorunu ortadan kaldırmak amacıyla tasarlanmıştır.

1.1. DNS'nin Tarihçesi

Yukarıda bahsedilen isimlendirme sorunu ilk olarak Internetin babası sayılan ArpaNet zamanında ortaya çıkmıştır. 1970'lerde ArpaNet günümüz ağları ile karşılaştırılamayacak kadar küçük durumdaydı ve yalnızca birkaç yüz ile ifade edilebilen sisteme hizmet veriyordu. Bu tarihlerde isimlendirme için tek noktada tutulan bir dosyanın bulunması ve diğer tüm sistemlerin bu dosyayı belli aralıklarla kendi taraflarında güncellemesi isimlendirme sorununu çözmüştü.

Adres-isim tanımlamalarını içeren `HOSTS.TXT` dosyası SRI tarafından SRI-NIC adında bir bilgisayar üzerinde tutulmaktaydı. Bu dosya her adrese bir isim karşılık gelecek şekilde düzenlenmişti. ArpaNet üzerindeki yeni isim tanımlamaları ve değişiklikleri SRI'ya gönderilen e-postalar aracılığı ile yapılıyor ve `HOSTS.TXT`'in kopyası ftp ile alınıyordu.

ArpaNet üzerinde TCP/IP kullanımına paralel olarak ortaya çıkan bağlantı patlaması, isim çözümü için bir çok sorunu da beraberinde getirdi. Öncelikle isim çakışmaları ortaya çıktı, sayı artmakta ve her bilgisayara özgün bir isim atanmasında problemler yaşanmaktaydı. Ayrıca yalnızca isim çözümlenmesi için oldukça yüksek miktarda bant genişliği harcanmakta ve buna rağmen kullanılan isim veritabanlarının uyumlu olması her zaman sağlanamamaktaydı.

Bu durumun ortaya çıkmasından sonra ArpaNet daha ölçeklenebilir bir isim çözümleme yapısı için araştırmalara başladı. Paul Mockapetris bu işle görevlendirildi. Mockapetris 1984 yılında Domain Name System (DNS)'i (Alan İsimlendirme Sistemi) tanımlayan RFC 882 ve 883'ü yayınladı. Bunlar daha sonra halen geçerli olan RFC 1034 ve 1035 tarafından güncellendiler.

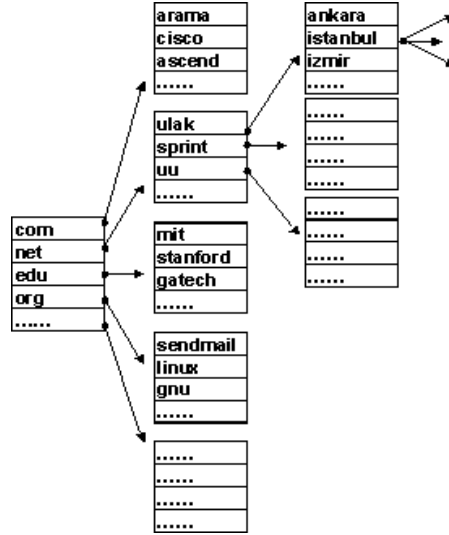
1.2. DNS'nin Yapısı

ArpaNet döneminde ortaya çıkan güçlükler nedeniyle DNS tasarlanırken uçlardaki sistemlerin kendi bilgilerini kendilerinin güncelleyebileceği bir yapı üzerinde durulmuştur. Ortaya çıkan yapı ise en üstten başlayarak hiyerarşik bir şekilde uçlara doğru açılan dağıtık bir veritabanı mimarisidir. Uçlar birbirleri ile istemci sunucu yöntemiyle konuşurlar.

Farklı tablolar ile tek veritabanında tanımlanmış bir alan adı sistemini incelenirse yapısının hiyerarşik olduğu görülür. Her alan adı bir başka alan adının altında tanımlanmıştır. En üst seviyede bulunan bir tablo en üst seviye alan adları olan `.com`, `net` vb içerir ve bu alan adlarının ayrıntılarını içeren tabloları işaret eder. Aynı şekilde bu tablolar da kendi altlarında bulunan alan adlarını içerir ve ayrıntılarını gösteren tablolara işaret eder.

Açıklanan tablo yapısı Şekil-1 üzerinde gösterilmiştir.

Şekil 1.



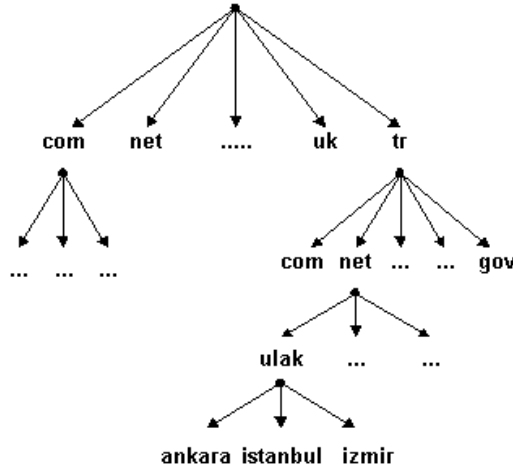
İlk tabloda en üst seviye alan adları tanımlanmış ve bu tabloda bulunan alt alan adlarının bilgileri ilgili tablolara işaret edilmiştir. Bu tek bir veritabanında gösterilmiş bir yapılanmadır. Bu yapıda tablolar farklı veritabanları üzerinde tutularak yönetim kolaylaştırılabilir. Bu durumda oldukça dinamik ve etkin bir mekanizma kurulmuş olur.

Dağıtık veritabanları arasında istemci-sunucu yöntemi ile bağlantı kurulur.

Az önce belirtilen en üst seviye alan adları arasında `com`, `net`, `gov` vb yanında ülkelerin ISO tarafından belirlenen sembolleri de tanımlanmıştır (`tr`, `uk`, `fr`, `gr` gibi).

Yukarıdaki açıklamaların paralelinde oluşan hiyerarşik alan adı ağacı Şekil 2'de gösterilmiştir.

Şekil 2.



Alan adı dağılımı en üst seviyeden başlar ve alt dallara doğru bölünür. Bir alan adının okunuşu en alttan başlayarak en üste doğru yapılır. Ağacın en altında bulunan alan adı `ankara.ulak.net.tr` şeklinde okunacaktır.

En üst seviye alan adlarının yönetimi **InterNIC** tarafından yapılmaktadır. ISO kodları ile tanımlanmış ülke adlarının yönetimi ülkelere göre değişiklik göstermektedir. Türkiye'nin ISO kodu olan `tr` üst seviye alan adının yönetimi Orta Doğu Teknik Üniversitesi (ODTÜ) tarafından yapılmaktadır. Dolayısıyla `com`, `net`, `gov` gibi alan adlarına kayıt **InterNIC** tarafından; `com.tr`, `net.tr`, `gov.tr` gibi alan adlarına kayıt ODTÜ tarafından yapılmaktadır. Alan adı kayıtlarıyla ilgili daha fazla bilgi için <http://www.internic.net>^(B2) ve <http://dns.metu.edu.tr>^(B3) adresleri incelenebilir.

`ankara.ulak.net.tr` örneğinde, `ankara.ulak.net.tr` alan adı UlakNet sunucuları üzerinde, `ulak.net.tr` alan adı ODTÜ sunucuları üzerinde, `net.tr` alan adı yine ODTÜ sunucuları üzerinde, `tr` alan adı ise `root server` (kök sunucusu) adı verilen en üst seviye alan adı sunucuları üzerinde tanımlanmış durumdadır. Kök sunucular, `tr` alan adını içeren alan adlarını çözümüemez, çözümü ODTÜ sunucularından ister. Dikkat edilmesi gereken nokta alan adları alındıktan sonra bu alan adındaki isimlendirmenin yerel olarak yapıldığıdır. Bu daha önce de değinilen yönetimin dağıtık olarak yapılabilmesini sağlar.

Bu bilgiler ışığında `xxx.ankara.ulak.net.tr` ya da `yyy.ulak.net.tr` şeklindeki bir tanımlama UlakNet'in kontrolünde, benzer şekilde `abc.com.tr` ya da `xyz.net.tr` şeklindeki bir tanımlama ise ODTÜ'nün kontrolündedir.

İncelenen alan adı ağacı maksimum 127 basamaktan oluşabilir ki bu da pratikte ulaşılması imkansıza yakın bir değerdir.

1.3. DNS Çözümlemesi

DNS'nin yapısı anlatılırken dağıtık bir veritabanı şeklinde oluşturulduğu ve uçların birbirleriyle istemci sunucu mantığı ile konuştuğunu söylenmişti. Bu işlevi yerine getiren programlara alan adı sunucu adı verilir (name server). Alan adı sunucuları alan adı ağacı üzerinde nokta ile gösterilirler.

Her alan adı sunucu bir veya birkaç alan adı bilgisini tutar ve bu alan adları için en yetkili alan adı sunucudur. Diğer alan adları için sorgularda bu alan adları için en yetkili alan adı sunucularını bulmaya çalışır.

Alan adı sunucular yerine getirdikleri kritik işlev nedeniyle genellikle yedekli olarak çalıştırılırlar. Bilgilerin tutulduğu ana veritabanı `birincil alan adı sunucu` olarak adlandırılır. İkincil sunucular, `birincil alan adı sunucularının` verilerini periyodik olarak kendi veritabanına kopyalarlar. Birincil sunucuda herhangi bir problem yaşandığında sorgulama ikincil sunucular üzerinde yapılır.

DNS çözümlemesi birkaç kademede incelenebilir. UlakNet alan adı sunucusunu kullanan bir istemcinin `www.ulak.net.tr` adresini sorguladığı durumda sunucu kendi veritabanında tanımlı olan bu adresi istemciye döndürecektir. Bu işlem, UlakNet alan adı sunucusu `ulak.net.tr` alan adı altında bulunan tanımlar için en yetkili sunucu olduğu için bu şekilde gerçekleşmiştir.

Şekil 3'te bu sorgunun nasıl gerçekleştirildiği görülmektedir.

Şekil 3.



Görüldüğü gibi istemci `www.ulak.net.tr` adresini bu alan adı için en yetkili durumdaki alan adı sunucusunda sorgulamış ve bu adrese tanımlanmış IP adresi cevap olarak döndürülmüştür.

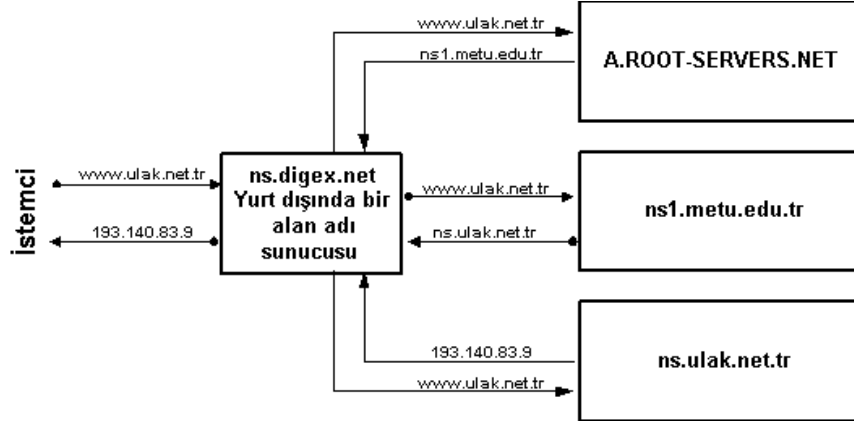
Benzer bir sorgulamayı Amerika kıtasında `ns.digex.net` adlı alan adı sunucuyu kullanan bir istemcinin yaptığındaki durum incelenebilir.

`ns.digex.net` kendisine sorulan `www.ulak.net.tr` için herhangi bir bilgiye sahip değildir. Bu yüzden kendi veritabanında tanımlı olan en üst seviye alan adı sunucularına (`root-servers` – daha sonra ayrıntılı olarak açıklanacaktır) bu adresi sorar. Bu sunucu da (`a.root-servers.net`) `www.ulak.net.tr` için kesin bilgiye sahip değildir. Ancak `.tr` üst seviye alan adının `ns1.metu.edu.tr` sunucusu tarafından kontrol edildiğini bilmektedir. Bu yüzden, `ns.digex.net`'e sorguyu `ns1.metu.edu.tr` üzerinde yapması bilgisini iletir. `ns.digex.net` bu kez aynı adresi `ns1.metu.edu.tr` üzerinde sorgulayacaktır. Ancak bu sunucu da `www.ulak.net.tr` için kesin adresi bilemeyecek ve sorgunun `ulak.net.tr` alan

adı sunucusu olan `ns.ulak.net.tr` adresine yönlendirilmesini bildirecektir. Son olarak `ns.digex.net`, `www.ulak.net.tr` adresini `ns.ulak.net.tr` üzerinde sorgulayacak ve `ns.ulak.net.tr` kendi veritabanında `www.ulak.net.tr` için tanımlı olan `193.140.83.9` adresini döndürecek. Bu bilgiye ulaşan `ns.digex.net` de kendi istemcisine bu bilgiyi ileticektir.

Bu sorgulama şekil 4'te gösterilmiştir.

Şekil 4.



Bu analizde dikkati çekmesi gereken iki önemli nokta bulunmaktadır.

Öncelikle `.tr` den sorumlu gözükken `ns1.metu.edu.tr`'nin `.net.tr` için ayrı bir alan adı sunucuya yönlendirme yapmadığı, doğrudan `ulak.net.tr` alan adı sunucusuna yönlendirme yaptığına dikkat edilmelidir. Bu işlem, `ns1.metu.edu.tr` hem `.tr` hem de `.net.tr` için alan adı sunucusu olduğu için bu şekilde gerçekleşmiştir. Eğer `.tr` ve `.net.tr` farklı sunucular üzerinde tanımlı olsalardı `ns.digex.net` ayrıca `.net.tr`'den sorumlu sunucuyu da sorgulamak durumunda kalacaktı.

Dikkat edilmesi gereken diğer nokta ise istemcinin alan adı sunucuya yalnızca bir sorgu iletmesi ve tüm iş sunucu tarafından yapıldıktan sonra yalnızca cevabı almasıdır. Aynı durum `ns.digex.net` diğer alan adı sunuculara ulaşırken ortaya çıkmamış ve her biri ayrı ayrı sorgulanmıştır. Bunun sebebi `ns.digex.net` sunucusunun tekrarlı (recursive), diğer alan adı sunucularının ise tekrarlı olmayan (iterative) bir şekilde sorgulanmış olmasıdır.

1.4. IP Adresinden İsim Çözümlemesi

Bölüm 1.3 (sayfa: 5)'te isimden adres çözümlemesi incelenmiştir. Şu ana kadar anlatılan yapı tüm sistemin isimden adres çözümü için tasarlandığı izlenimini vermektedir. Ancak pratik ihtiyaçlar, IP adresinden isim çözümünü de gerekli kılmaktadır.

Böyle bir sorunun şu ana kadar incelenen yapı üzerinde nasıl çalışabileceğine bakalım. İsimlerin indekslenmesine göre oluşturulmuş hiyerarşik bir sistem görülmektedir. Ancak IP adreslerin bu yapı üzerinde herhangi bir hiyerarşik yapısı bulunmamaktadır. Örneğin `www.ulak.net.tr` adresi `193.140.83.9` IP numarası ile adreslenmişken `truva.ulakbim.gov.tr` adresi `193.140.83.13` IP numarası ile adreslenmiş olabilir. Bu da `193.140.83.` ile başlayan IP'lerin teorik olarak tüm alan adı uzayına yayılmış olabileceğini gösterir.

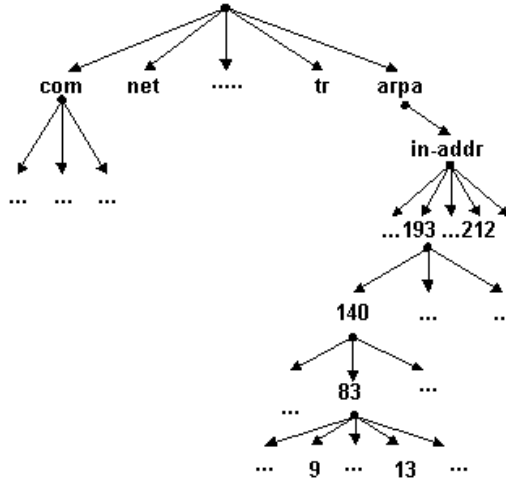
Görüldüğü gibi mevcut yapıda bir IP adresinin isim karşılığını bulmak için veritabanının tümünün taranması gerekmektedir. Bu ise indeksi olmayan dağınık durumdaki milyonlarca kaydın taranması anlamına gelir ki imkansıza yakın bir uğraştır. Bu sorunun çözülebilmesi için IP adresleri için de hiyerarşik bir yapının kurulması gerektiği görülmektedir.

Bu yapıya geçmeden önce IP adreslemenin özelliklerinin hatırlanması yararlı olacaktır. IP adresleri ilk oktetten son oktete (soldan sağa doğru) en genelden en özele doğru sıralanırlar. Örneğin `193.140` hem `193.140.83.13`'ü hem de `193.140.83.9`'u içine alır.

Alan adları ise buna ters olarak sağdan sola doğru en genelden en özele doğru sıralanırlar. Örneğin `.tr` hem `ulak.net.tr`'yi hem de `metu.edu.tr`'yi içine alır.

İki gösterim de genelden özele sıralanabildiğine göre mevcut hiyerarşik yapılandırmaya IP numaraları için bir üst seviye alan adı eklenmesi mümkündür. Bu alan adı `in-addr.arpa` şeklinde ifade edilir. Bu üst seviye alan adı 1'den 255'e kadar çeşitli alt alanlara bölünür. Bu bölümlmeye 4 oktetli tamamlayacak şekilde devam edilebilir. Bu durumda ortaya çıkan yapı şekil 5'te gösterilmiştir.

Şekil 5.



Üst seviye alan adları arasına `arpa` adında bir alan adı eklenmiştir. Bu alan adının altında `in-addr` şeklinde bir alan daha tanımlanmıştır. Bu alan adının altında IP numarasının ilk oktetli alan adı olarak eklenmiş, bunu sırayla IP numarasının ikinci, üçüncü ve dördüncü oktetleri izlemiştir.

Görüldüğü gibi artık `193.140.83.13` ve `193.140.83.9` IP numaralarının isim karşılıklarını bulmak son derece kolaylaşmıştır. İncelenen isim sorgusuna benzer bir şekilde IP–isim sorguları da bu yapının yardımıyla yapılabilmektedir. Ancak dikkat edilmesi gereken önemli birkaç nokta bulunmaktadır.

Normal alan adı isimlendirme sisteminde isimler ağacın en altından en üstüne doğru okunur. Bu durumda IP numaraları bilinenin aksine özelden genele doğru okunacaklardır. Örneğin `193.140.83.13` IP numarasının DNS ağacı üzerinde okunuşu `13.83.140.193.in-addr.arpa` şeklinde olacaktır.

Dikkat edilmesi gereken diğer bir nokta ise isimden IP'ye yapılan tanımlama ile IP'den isime yapılan tanımlama arasında herhangi bir bağlantı bulunmayışıdır. Örneğin `www.ulak.net.tr` adresi DNS üzerinde `193.140.83.9` olarak tanımlıyken, `193.140.83.9` IP adresinin ismi `efe.ulakbim.gov.tr` şeklinde tanımlı olabilir.

1.5. DNS Ön belleği

DNS çözümlemesinde anlatıldığı şekilde isim çözümleme işlemi ardışıl olarak alan adı sunucularının sorgulanmasını gerektirir. Bu işlem oldukça vakit alıcı bir işlem olabilir ve karşılıklı olarak sistemler üzerinde yük oluşturduğu gibi hatlar üzerinde yüksek miktarda bant genişliği harcayabilir.

Bunun engellenmesi ve sorgunun mümkün olan en kısa sürede sonuçlandırılabilmesi için alan adı sunucularına ön bellek (cache) özelliği eklenmiştir. Bu sayede bir alan adı sunucu daha önce sorguladığı alan adlarını kendi belleğinde tutarak yeni sorgularda diğer alan adı sunucularını sorgulamadan doğrudan cevap verebilir. Bu yukarıda anlatılan olumsuzlukları belli bir derecede önleyecektir. Ancak burada önemli bir nokta gözden kaçırılmamalıdır.

Sorgulanan alan adında son sorgulamadan sonra değişiklik yapılmış olabilir. Bu durumda bellekte tutulan bilgi güncelliğini yitirmiştir ve istemciye hatalı bilgi geri döndürülecektir.

Bu ancak belli bir noktaya kadar kabul edilebilir bir durumdur. Alan adı tanımları normalde çok sık değiştirilen tanımlamalar değildir. Nadiren değiştirildiklerinde de değişikliğin etkinleşmesi için bir gün gibi bir süre çoğu zaman yeterlidir. Şu halde bellekte tutulan bilgilerin bir süre sonra güncelliğini yitirdiği kabul edilmelidir. Bunu belirleyen değere daha önceki konularda da değinildiği gibi **TTL** (Time To Live – Yaşam Süresi) denilmektedir ve her alan adının tanımlanmasında bu alan adı için geçerli **TTL** değeri belirtilir. Bu alan adından sorgulanan bilgiler bellekte **TTL** süresince tutulduktan sonra güncelliğini yitirdiği kabul edilir. **TTL**'in nasıl tanımlandığı daha sonra incelenecektir.

2. BIND

Bu bölüme kadar olan kısımda isim çözümleme işlemi açıklanmıştı. Burada ise bir alan adı sunucunun nasıl kurulduğu ve yapılandırması ayrıntılarıyla anlatılacaktır.

BIND (Berkeley Internet Name Domain) adını geliştirildiği Berkeley üniversitesinden alan günümüz Internetinin tartışmasız en popüler alan adı sunucu yazılımıdır. Unix sistemlerinde tasarlanmış ve çalıştırılmaya başlamış olmasına rağmen bugün Microsoft NT için üretilmiş dağıtımları da mevcuttur.

2.1. BIND Paketinin Temini

BIND'in geliştirilmesi Internet Yazılım Konsorsiyumu (Internet Software Consortium) tarafından yapılmaktadır ve web sayfalarından ücretsiz olarak indirilebilir.

BIND'in bu belgenin yazımı anındaki en son sürümü Kasım 1999'da yayınlanmış olan sürüm 8.2.2 yama 5'tir. [http://www.isc.org/products/BIND^{\(B5\)}](http://www.isc.org/products/BIND^(B5)) adresinden indirilebilir.

BIND'in yaygın kullanımda olan 4. ve 8. sürümleri arasında oldukça büyük farklar bulunmaktadır. Bu belgede 8. sürüm incelenecek olup, 4. sürümünü kullananların da mümkünse bu sürüme terfi etmeleri önerilir.

2.2. BIND'in Kurulumu

BIND paketinin indirilmesi, derlenmesi ve kurulması örneklenerek gösterilecektir. Test makinası sparc mimari üzerinde çalışan bir Red Hat Linux'tur. Ancak Linux üzerinde dağıtım ile beraber gelen derlenmiş durumdaki BIND da kullanılabilir. Burada derlemenin verilmesindeki amaç farklı platformlardaki kullanıcılara yardımcı olmaktır.

```
[ilker@hornet bind]$ ftp ftp.isc.org
Connected to isrv4.pa.vix.com.
220 isrv4.pa.vix.com FTP server (Version wu-2.6.0(1) Tue Nov 30 17:39:57 PST 1999) ready.
Name (ftp.isc.org:ilker): ftp
331 Guest login ok, send your complete e-mail address as password.
Password:
230 Guest login ok, access restrictions apply.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> cd isc/bind/src/8.2.2-P5
250 CWD command successful.
ftp> bin
200 Type set to I.
ftp> get bind-src.tar.gz
local: bind-src.tar.gz remote: bind-src.tar.gz
200 PORT command successful.
150 Opening BINARY mode data connection for bind-src.tar.gz (1251509 bytes).
```



```

226 Transfer complete.
1251509 bytes received in 30.6 secs (40 Kbytes/sec)
ftp> bye
221-You have transferred 1251509 bytes in 1 files.
221-Total traffic for this session was 1252085 bytes in 1 transfers.
221-Thank you for using the FTP service on isrv4.pa.vix.com.
221 Goodbye.

[ilker@hornet bind]$ tar -xzvf bind-src.tar.gz
... ..
[ilker@hornet bind]$ cd src
[ilker@hornet src]$ make depend
... ..
[ilker@hornet src]$ make
... ..
[ilker@hornet src]$ su
Password:
[root@hornet src]# make install
... ..
[root@hornet src]# /usr/sbin/named -v
named 8.2.2-P5 Sun Apr 23 13:45:06 EEST 2000
      ilker@hornet.ulakbim.gov.tr:/home/ilker/download/bind/src/bin/named
[root@hornet src]#

```

Görüldüğü gibi BIND'in kurulumu başarıyla tamamlanmıştır.

2.3. BIND Yapılandırması

BIND yapılandırma dosyası olarak, aksi belirtilmediği sürece `/etc/named.conf` dosyasını kullanır. BIND'in 4. sürümü için bu dosya `/etc/named.boot` dosyasıdır ve dosyaların biçimleri birbirinden hayli farklıdır bu yüzden iki dosyanın karıştırılmamasına dikkat edilmelidir.

Örnek bir `named.conf` dosyasını aşağıda gösterilmiştir.

```

// /etc/named.conf
// Ilker TEMIR - 23/04/2000
//

options {
    directory "/var/named";
};

zone "." {
    type hint;
    file "named.root";
};

zone "0.0.127.in-addr.arpa" {
    type master;
    file "rev.local";
};

zone "ulak.net.tr" {
    type master;
    file "zone.ulak.net.tr";
};

zone "83.140.193.in-addr.arpa" {

```

```

type master;
file "rev.193.140.83";
};

```

Düşünülenin aksine `named.conf` çoğu durumda örnekte görüldüğü kadar yalın olacaktır. Dosya içeriği aşağıda incelenecektir.

Seçenekler (`options`) kısmında tanımlanan dizin (`directory`), adres tanımlarını içeren dosyaların bulunacağı dizindir. Belirtilen tüm dosyalar buradaki dizin altında yer alacaklardır. `/var/named` genellikle bu amaçla en çok kullanılan dizindir.

`zone "."` tanımı değiştirilmemesi gereken standart bir tanımdır. DNS çözümlemesi kısmında açıklanan üst seviye alan adı sunucularını gösterir, sunucu NS çözümünü yapamadığı alanlar için burada tanımlı olan alan adı sunuculara başvuracaktır. Bu sunucu adreslerinin nereden temin edileceği ilerleyen kısımlarda açıklanacaktır.

`zone "0.0.127.in-addr.arpa"` tanımı `loopback` olarak tabir edilen bilgisayarın kendi kendine bağlanması için gereken adreslerin ters tanımlamaları için gereklidir. Bu tanım da değiştirilmeden tutulmalıdır.

`zone "ulak.net.tr"` tanımı ile başlayan kısım anlaşılması gereken en önemli kısımlardan biridir. Örnekte bu, `ulak.net.tr` şeklinde bir alan adı tanımlandığını gösterir. `Type` belirteci ise sunucunun alan adı için durumunu belirtir. Örnekte belirtilmiş olan `master` birincil isim sunucusunu göstermektedir. `File` belirteci ile belirtilen isim ise `ulak.net.tr` için adres IP tanımlamalarını içeren dosyanın adıdır. Örnekte bu dosya `/var/named/zone.ulak.net.tr` olmaktadır.

`Zone "83.140.193.in-addr.arpa"` tanımı ile başlayan kısım da bir öncekine benzer şekilde `83.140.193.in-addr.arpa` alan adını tanımlamaktadır. Çözümleme kısmında açıklandığı şekilde bu alan adı `193.140.83` ağında bulunan IP adreslerinin isim karşılıklarını bulmak için kullanılacaktır.

`named.conf` dosyası BIND'a hangi alanlardan sorumlu olduğu ve bu alanlara ait bilgileri nerede bulacağı gibi parametreleri vermektedir. Ancak IP adres karşılıkları bu dosyada işaret edilen veri dosyalarında bulunacaktır. Belirtilen dosya adları tamamen sistem yöneticisinin tercihinin kalmıştır ancak sistematik bir yol izlenmesi faydalı olacaktır.

`named.conf` dosyasında belirtilen dosyaların içerikleri aşağıda incelenmiştir. Belirtilen dosyaların tamamı `/var/named` dizini altında yer alacaktır.

named.root: Bu dosya `root-server` adı verilen en üst seviyede bulunan alan adı sunucularının isimlerini ve adreslerini içermektedir. Sıklıkla değişmemekle birlikte güncel hali <ftp://ftp.rs.internic.net>^(B6) adresinden çekilebilir. Bu satırların yazıldığı tarih itibarıyla bu dosya şu şekildedir:

```

;      This file holds the information on root name servers needed to
;      initialize cache of Internet domain name servers
;      (e.g. reference this file in the "cache . <file>"
;      configuration file of BIND domain name servers).
;
;      This file is made available by InterNIC registration services
;      under anonymous FTP as
;          file           /domain/named.root
;          on server      FTP.RS.INTERNIC.NET
;      -OR- under Gopher at  RS.INTERNIC.NET
;          under menu     InterNIC Registration Services (NSI)
;          submenu        InterNIC Registration Archives
;          file           named.root
;
;      last update:      Aug 22, 1997
;      related version of root zone:  1997082200
;
;

```

```
; formerly NS.INTERNIC.NET
;
.                3600000    IN    NS      A.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET. 3600000    A      198.41.0.4
;
; formerly NS1.ISI.EDU
;
.                3600000    NS      B.ROOT-SERVERS.NET.
B.ROOT-SERVERS.NET. 3600000    A      128.9.0.107
;
; formerly C.PSI.NET
;
.                3600000    NS      C.ROOT-SERVERS.NET.
C.ROOT-SERVERS.NET. 3600000    A      192.33.4.12
;
; formerly TERP.UMD.EDU
;
.                3600000    NS      D.ROOT-SERVERS.NET.
D.ROOT-SERVERS.NET. 3600000    A      128.8.10.90
;
; formerly NS.NASA.GOV
;
.                3600000    NS      E.ROOT-SERVERS.NET.
E.ROOT-SERVERS.NET. 3600000    A      192.203.230.10
;
; formerly NS.ISC.ORG
;
.                3600000    NS      F.ROOT-SERVERS.NET.
F.ROOT-SERVERS.NET. 3600000    A      192.5.5.241
;
; formerly NS.NIC.DDN.MIL
;
.                3600000    NS      G.ROOT-SERVERS.NET.
G.ROOT-SERVERS.NET. 3600000    A      192.112.36.4
;
; formerly AOS.ARL.ARMY.MIL
;
.                3600000    NS      H.ROOT-SERVERS.NET.
H.ROOT-SERVERS.NET. 3600000    A      128.63.2.53
;
; formerly NIC.NORDU.NET
;
.                3600000    NS      I.ROOT-SERVERS.NET.
I.ROOT-SERVERS.NET. 3600000    A      192.36.148.17
;
; temporarily housed at NSI (InterNIC)
;
.                3600000    NS      J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET. 3600000    A      198.41.0.10
;
; housed in LINX, operated by RIPE NCC
;
.                3600000    NS      K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET. 3600000    A      193.0.14.129
;
; temporarily housed at ISI (IANA)
;
```

```

.                3600000      NS      L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET. 3600000      A       198.32.64.12
;
; housed in Japan, operated by WIDE
;
.                3600000      NS      M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET. 3600000      A       202.12.27.33
; End of File

```

rev.local: Bu dosya 127.0.0 ağına ait olan IP adreslerinin isim tanımlamalarını içermektedir. İçeriği aşağıda verilmiştir. İçeriği **rev.193.140.83** dosyasının açıklanmalarında bulunabilir.

```

$TTL 86400
@          IN      SOA      hornet.ulak.net.tr. root.hornet.ulak.net.tr. (
                                2000042300 ; Serial
                                28800      ; Refresh
                                14400      ; Retry
                                3600000     ; Expire
                                86400 )    ; Minimum
IN         NS       hornet.ulak.net.tr.
1          IN      PTR     localhost.

```

zone.ulak.net.tr: Bu dosya **ulak.net.tr** alan adı altında tanımlı isim adres tanımlamalarını içermektedir. İçeriği aşağıda sunulmuştur.

```

$TTL 86400
@          IN      SOA      hornet.ulak.net.tr. root.hornet.ulak.net.tr. (
                                2000042300 ; Serino
                                28800      ; Yenileme
                                14400      ; Tekrar
                                3600000     ; Süre
                                86400 )    ; TTL
IN         NS       hornet.ulak.net.tr.
ulak.net.tr. IN     MX       10 efe.ulak.net.tr.
efe        IN      A        193.140.83.9
truva     IN      A        193.140.83.13
hornet    IN      A        193.140.83.91
www       IN      CNAME     efe.ulak.net.tr.

```

İlk satırda görülen değer daha önce açıklanan TTL değeridir. Bu sunucuya ulaşan istemcilerin sorguladıkları isimleri ne kadar süreyle belleklerinde tutacaklarını gösterir. Bu değer bilinçsiz olarak değiştirilmesi önerilmez. Çoğu durum için 24 saat uygun bir değerdir. Çok küçük olması DNS trafiğini artıracaktır. Ayrıca 0 yapmaktan kaçınılması gerekir.

İkinci satırın başında görülen @ alan adını belirten bir kısa gösterimdir, örnekte **ulak.net.tr.**'nin yerine geçmiştir.

@'in yanında görülen **IN**, tanımın Internet için bir tanım olduğunu gösterir (Internet haricinde tanımlar yapabilmek mümkün olsa da burada değinilmeyecektir).

SOA, **S**tart **O**f **A**uthority anlamına gelir ve alan adı ile ilgili genel tanımları gösterir. Daha sonra gelen **hornet.ulak.net.tr.** sunucunun üzerinde çalıştığı bilgisayarın adı. **root.hornet.ulak.net.tr.** ise ilk **.**, @ ile değiştirilmek kaydıyla sunucudan sorumlu kişinin e-posta adresini gösteren tanımlardır.

Sırasıyla verilen **serino**, **yenileme**, **tekrar** ve **süre** sayısal değerleri sunucu için çeşitli sayaç ve zaman kıstaslarını gösterir. Zaman birimlerinin tamamı saniyedir. Ne anlama geldikleri **daha sonra** (sayfa: 13)

incelenecektir.

Daha sonra görülen **NS** kaydı **Name Server** (alan adı sunucu) tanımıdır. Her alan adı için en az bir alan adı sunucusu bu şekilde tanımlanmalıdır.

Sonraki satırda **ulak.net.tr.** için tanımlanmış olan **MX 10** kaydı görülmektedir. **MX**, **Mail EXchanger**'ı temsil eder ve belirtilen alan adı için posta servisinin çalıştığı bilgisayarın ismini gösterir. 10 ise tercih değeridir. Farklı tercih değerleri içinde en düşük tercih değeri seçilecektir. Buradaki örnek, **ulak.net.tr** alan adına giden mesajların **efe.ulak.net.tr** bilgisayarına yönlendirilmesini sağlar.

Farklı isimlerin yanında olan **A** kayıtları, adres kayıtlarını temsil eder. İsmi hangi IP numarasına karşılık geldiğini gösterir.

Son satırda görülen **CNAME** tanımı ise **Canonical NAME**'den gelir ve bilgisayarlara ikincil isim verilmesini sağlar. Örnekte görülen tanım **www.ulak.net.tr** adresinin **efe.ulak.net.tr** adresi ile aynı olmasını sağlar. **A** kaydına tercih edilmesinin nedeni **efe.ulak.net.tr** tanımı değiştiğinde onu gösteren **CNAME** kayıtlarının da otomatik olarak değişmesidir. Kullanımda bazı problemler yaratabildiği için kullanılması önerilmez. **CNAME** kullanmak yerine farklı **A** tanımları yapmak daha uygundur.

rev.193.140.83: Bu dosya **193.140.83** ağına ait olan IP adreslerinin isim tanımlamalarını içermektedir. İçeriği aşağıda verilmiştir.

```
$TTL 86400
@          IN      SOA      hornet.ulak.net.tr. root.hornet.ulak.net.tr. (
                                2000042300 ; Serino
                                28800      ; Yenileme
                                14400      ; Tekrar
                                360000     ; Süre
                                86400 )    ; TTL
IN         NS       hornet.ulak.net.tr.
9          IN      PTR      efe.ulak.net.tr.
13         IN      PTR      truva.ulak.net.tr.
91         IN      PTR      hornet.ulak.net.tr.
```

Görüldüğü gibi alan adı tanımları hemen hemen standarttır ve bu dosyanın da içerik olarak az önce incelenen dosyadan çok farkı yoktur. Fark yalnızca yeni ortaya çıkan **PTR** kaydıdır, **PoinTeR** (işaret, gösterge) anlamına gelir. Örnekler sırasıyla **193.140.83.9**'un **efe.ulak.net.tr**, **193.140.83.13**'ün **truva.ulak.net.tr** ve **193.140.83.91**'in **hornet.ulak.net.tr**'yi adreslediğini gösterir.

Sıklıkla kullanılan temel kayıtlar iki dosyanın içeriğinin incelenmesi sırasında açıklanmıştır. Ancak burada belirtilmeyen farklı kayıtlar da mevcuttur. Bunların listesi ve açıklamaları [daha sonra](#) (sayfa: 17) verilecektir.

SOA kaydında belirtilen parametreler aşağıda ayrıntılı olarak incelenmişlerdir.

Seri No:

Sunucudan bilgi alan ikincil isim sunucular tarafından kullanılır. Burada belirtilen değer ne olduğu gerçekte çok önemli değildir. Önemli olan dosya içeriğinde değişiklik yapıldığında mutlaka artırılması gerektiridir. İkincil isim sunucular bu değere bakarak alan adı içeriğinin değişip değişmediğine karar verirler.

Genellikle yıl, ay, gün ve gün içinde yapılan değişiklik sayısı şeklinde kullanılır.

Yenileme:

İkincil sunucular için birincil sunucuyu ne kadar aralıkla kontrol edeceğini gösteren değerdir. Örneğimizde bu değer 8 saattir. Her 8 saatte bir ikincil sunucu birincil sunucuda değişiklik yapıp yapılmadığını kontrol edecektir.

Tekrar:

Eğer ikincil sunucu herhangi bir nedenle birinci sunucuya ulaşmakta sorun yaşarsa tekrar denemek için ne kadar bekleyeceğini gösteren değerdir. Genellikle **Yenileme** süresinden daha kısa bir süredir. Örneğimizde 4 saattir.

Süre:

Bu kısım oldukça önemlidir. İkincil sunucunun birincil sunucuya ulaşamadığı durumlarda kendisinde bulunan verinin ne kadar daha doğru kabul edildiğini gösterir. Bu süreden sonra ikincil sunucu çağrılara cevap vermeyi keser. 1 haftadan kısa olması kesinlikle önerilmez. **Yenileme** ve **Tekrar** değerlerinden çok daha büyük olmalıdır.

TTL:

Daha önce açıklanan 'Yaşam Süresi' (TTL – Time to live) değeridir. İlk satırda **\$TTL** ile belirtilenden farkı BIND 8.2.1 ile birlikte buradaki değerin yalnız negatif TTL olarak kullanılmasıdır. Negatif TTL hata veren sorgulamaların bellekte tutulması için geçerli süredir.

2.4. BIND'in Çalıştırılması

Daha önce açıklanan yapılandırma tamamlandıktan sonra, BIND, root kullanıcısı yetkileriyle aşağıdaki şekilde çalıştırılır.

```
[root@hornet /root]# /usr/sbin/ndc start
new pid is 9089
[root@hornet /root]#
```

Çalışıp çalışmadığını anlamak için aşağıdaki yol izlenebilir.

```
root@hornet /root]# ps -aux |grep named
root      9089  0.0  1.4 2568 1808 ?        S    09:23   0:00 /usr/sbin/named
root      9131  1.0  0.5 1464  712 pts/2    S    09:23   0:00 grep named
[root@hornet /root]#
```

Görüldüğü gibi **named** işlemi **9089** PID ile çalışmaya başlamıştır. Herhangi bir hata verip vermediği günlük dosyalarından kontrol edilmelidir.

```
[root@hornet /root]# tail /var/log/messages

Apr 24 15:15:30 hornet named[9088]: starting. named 8.2.2-P5 Sun Apr 23 13:45:06
EEST 2000 ^Iilker@hornet.ulakbim.gov.tr:/home/ilker/download/bind/src/bin/named
Apr 24 15:15:30 hornet named[9088]: hint zone "" (IN) loaded (serial 0)
Apr 24 15:15:30 hornet named[9088]: master zone "0.0.127.in-addr.arpa" (IN)
loaded (serial 2000042300)
Apr 24 15:15:30 hornet named[9088]: master zone "ulak.net.tr" (IN) loaded
(serial 2000042301)
Apr 24 15:15:30 hornet named[9088]: master zone "83.140.193.in-addr.arpa"
(IN) loaded (serial 2000042301)
Apr 24 15:15:30 hornet named[9088]: listening on [127.0.0.1].53 (lo)
Apr 24 15:15:30 hornet named[9088]: listening on [193.140.83.91].53 (eth0)
Apr 24 15:15:30 hornet named[9088]: Forwarding source address is [0.0.0.0].1038
Apr 24 15:15:30 hornet named[9089]: Ready to answer queries.
[root@hornet /root]#
```

Bilgi: Sistem günlükleri farklı Unix türevlerinde farklı dizinlerde bulunabilir. **/var/log/messages** Linux için geçerli bir günlük dosyasıdır.

Görüldüğü gibi günlüklerde bir sorunu gösteren herhangi bir belirti bulunmamaktadır. Alan adı sunucusu sorunsuz olarak çalışmaktadır.

2.5. DNS Sorguları – NSLOOKUP

`Nslookup` alan adı sunucularını sorgulamak için oldukça etkili bir yazılımdır. Çok ayrıntılı kullanım seçenekleri mevcut olsa da burada ayrıntıya girilmeyecektir. Hemen tüm Unix sistemlerinde yüklü olarak gelmektedir ayrıca Windows NT üzerinde de yüklü olarak gelir.

Çalışan sunucunun `nslookup` yardımıyla sorgulanması aşağıda gösterilmiştir.

```
[root@hornet named]# nslookup - 0
Default Server: hornet.ulak.net.tr
Address: 0.0.0.0

> set type=soa
> ulak.net.tr
Server: hornet.ulak.net.tr
Address: 0.0.0.0

ulak.net.tr
    origin = hornet.ulak.net.tr
    mail addr = root.hornet.ulak.net.tr
    serial = 2000042301
    refresh = 28800 (8H)
    retry = 14400 (4H)
    expire = 3600000 (5w6d16h)
    minimum ttl = 86400 (1D)
ulak.net.tr    nameserver = hornet.ulak.net.tr
hornet.ulak.net.tr    internet address = 193.140.83.91
> set type=a
> truva.ulak.net.tr
Server: hornet.ulak.net.tr
Address: 0.0.0.0

Name: truva.ulak.net.tr
Address: 193.140.83.13

> www.ulak.net.tr
Server: hornet.ulak.net.tr
Address: 0.0.0.0

Name: efe.ulak.net.tr
Address: 193.140.83.9
Aliases: www.ulak.net.tr

> set type=ptr
> 9.83.140.193.in-addr.arpa
Server: hornet.ulak.net.tr
Address: 0.0.0.0

9.83.140.193.in-addr.arpa    name = efe.ulak.net.tr
83.140.193.in-addr.arpa nameserver = hornet.ulak.net.tr
hornet.ulak.net.tr    internet address = 193.140.83.91
```

İlk satırda görülen, `nslookup` komutunun alan adı sunucu olarak kendi çalıştığı makinayı kullanacak şekilde çalıştırılmasıdır. Sorgu tipi `SOA`'ya değiştirilerek (açılıştaki varsayılan sorgu tipi `A` kaydı sorgusudur)

`ulak.net.tr` alan adı için SOA kaydı incelenmiştir. Tüm değerler daha önce tanımlananlarla uyum içerisindedirler. Daha sonra sorgu türü A kaydına çevrilmiş ve `truva.ulak.net.tr` için IP adresi bulunmuştur. Veritabanında CNAME olarak tanımlanmış olan `www.ulak.net.tr` için yapılan A kaydı sorgulaması bu adresin takma ad olarak tanımlandığı `efe.ulak.net.tr` adresini döndürmüştür. Son olarak yapılan PTR sorgulamasıyla da `193.140.83.9`'un ters (reverse) tanımı bulunmuştur. Bu da daha önce tanımlanan `efe.ulak.net.tr` şeklindedir.

Nslookup sorguları genişletilerek devam edilirse;

```
> set type=a
> www.metu.edu.tr.
Server:  hornet.ulak.net.tr
Address:  0.0.0.0

Non-authoritative answer:
Name:      www.metu.edu.tr
Address:   144.122.199.13
```

`www.metu.edu.tr` alan adı için ilgili adres döndürülecektir. DNS tanımlarında Türkiye ile herhangi bir tanım olmadığı halde bu işlem nasıl yapılmıştır.

Daha önce açıklandığı şekilde root-server kullanılarak `.tr` alan adından sorumlu sunucular bulunmuş ve ardışıl sorgulama ile adrese ulaşılmıştır.

Sorgulamaya aşağıdaki şekilde devam edilsin.

```
> set norecurse
> www.bilkent.edu.tr.
Server:  hornet.ulak.net.tr
Address:  0.0.0.0

Name:      www.bilkent.edu.tr
Served by:
- ns1.metu.EDU.tr
    144.122.199.90
    EDU.tr
- ns2.metu.EDU.tr
    144.122.199.93
    EDU.tr
- ns.ulak.net.tr

    EDU.tr
- firat.bcc.bilkent.EDU.tr
    139.179.10.13
    EDU.tr

> www.metu.edu.tr.
Server:  hornet.ulak.net.tr
Address:  0.0.0.0

Non-authoritative answer:
Name:      www.metu.edu.tr
Address:   144.122.199.13

> www.ulakbim.gov.tr.
Server:  hornet.ulak.net.tr
Address:  0.0.0.0
```



```

Name:      www.ulak.net.tr
Served by:
- NS1.METU.EDU.tr
    144.122.199.90
    tr
- NS2.METU.EDU.tr
    144.122.199.93
    tr
- MUNNARI.OZ.AU
    128.250.1.21
    tr
- NS.RIPE.NET
    193.0.0.193
    tr
- NS1-AUTH.SPRINTLINK.NET
    206.228.179.10
    tr
- NS.UU.NET
    137.39.1.3
    tr

```

`set norecurse` komutu daha önce açıkladığımız tekrarlı sorgulama modunu iptal etmiştir. Bu komuttan sonraki sorgular tekrarlı olarak yapılmayacaktır.

Tekrarlı sorgu kapatıldıktan sonra bu kez `www.bilkent.edu.tr` için sorgulama yapılmış ancak sonucu cevap olarak `.edu.tr`'ler için geçerli sunucuları döndürmüştür.

`www.ulakbim.gov.tr` alan adı sorgulandığında ise `.tr`'ler için geçerli sunucular dönmüş ancak `www.metu.edu.tr` için IP karşılığı döndürülmeye devam etmiştir. Bu nasıl olmaktadır?

Öncelikle tekrarlı sorguyu kapatmadan önce `www.metu.edu.tr` için sorgulama yapıldığı hatırlanmalıdır. Bu sorgulama sırasında sunucumuz bu adres için yaptığı tüm sorguları belleğinde saklamaktadır. Bunlar `.tr` alan adı sunucuları, `.edu.tr` alan adı sunucuları, `.metu.edu.tr` alan adı sunucuları ve `www.metu.edu.tr` adresinin IP numarasıdır. Bu yüzden `.edu.tr` altında bulunan `www.bilkent.edu.tr` sorgulaması yapıldığında bu alan adı hakkında en fazla bilgiye sahip olan `.edu.tr` alan adı sunucuları, `.gov.tr` altında bulunan `www.ulakbim.gov.tr` sorgulaması yapıldığında ise en fazla bilgiye sahip olan `.tr` alan adı sunucuları döndürülmüştür. `www.metu.edu.tr` adresi ise bellekten doğrudan cevaplanmıştır.

2.6. DNS Kayıtları

Geçerli DNS kayıtlarına daha önce de değinilmişti. Bu kayıtlar örneklerde belirtilenlerden sayıca daha fazladır. Burada daha önce açıklananlarla birlikte ayrıntıya girilmeden açıklanacaktır. Halen deneysel durumda olan kayıtlara burada değinilmeyecektir. Tam listesi RFC1035'ten bulunabilir.

A:

Adres kayıdır. İsme karşılık gelen 32 bitlik IP numarası tanımlanır.

Örnek:

```
marine.ulakbim.gov.tr.  IN      A      193.140.83.27
```

CNAME:

Takma ad kayıdır. Takma ad tanımlanır.

Örnek:

```
ilker.ulakbim.gov.tr.  IN      CNAME   truva.ulakbim.gov.tr.
```

HINFO:

Bilgi kaydıdır, işlemci işletim sistemi tanımlar.

Örnek:

```
truva.ulakbim.gov.tr. IN HINFO Ultra10 Solaris
```

MD:

Kullanımdan kalkmıştır. Yerine MX kaydı kullanılır.

MF:

Kullanımdan kalkmıştır. Yerine MX kaydı kullanılır.

MX:

Posta sunucu kaydı. Alan adı için ilgili posta sunucusunu tanımlar.

Örnek:

```
ulakbim.gov.tr. IN MX 10 efe.ulakbim.gov.tr.
```

NS:

Alan adı sunucu kaydı. Alan için alan adı sunucularını tanımlar.

Örnek:

```
ulakbim.gov.tr. IN NS efe.ulakbim.gov.tr
```

PTR:

İşaret kaydı. Ters tanımlamada ismi tanımlar.

Örnek:

```
9.83.140.193.in-addr.arpa IN PTR efe.ulakbim.gov.tr.
```

SOA:

Sunucunun genel özelliklerini tanımlayan kayıttır. Seri No, yenileme, tekrar, süre ve TTL değerlerini alır.

Örnek:

```
ulakbim.gov.tr. IN SOA efe.ulakbim.gov.tr. root.ulakbim.gov.tr. (
    2000042401 ; Serino
    28800 ; Yenileme
    14400 ; Tekrar
    3600000 ; Süre
    86400 ) ; TTL
```

TXT:

Bilgi kaydı. Kayıt hakkında bilgiyi tanımlar.

Örnek:

```
ulakbim.gov.tr. IN TXT "Ulusal Akademik Ağ ve Bilgi Merkezi"
```

WKS:

Servis kaydı. Çalışan servisleri tanımlar.

Örnek:

truva.ulakbim.gov.tr.	IN	WKS	TCP (ftp, ssh, telnet, smtp)
-----------------------	----	-----	------------------------------

3. Yasal Açıklamalar

3.1. Telif Hakkı ve Lisans

Bu belgenin, *BIND 8 ile DNS NASIL*, 1.0 sürümünün **telif hakkı © 2000 İlker Temir**'e aittir. Bu belgeyi, Free Software Foundation tarafından yayınlanmış bulunan [GNU Genel Kamu Lisansı](http://www.gnu.org/copyleft/gpl.html)^(B9)'nin 2. ya da daha sonraki sürümünün koşullarına bağlı kalarak kopyalayabilir, dağıtabilir ve/veya değiştirebilirsiniz. Bu Lisansın özgün kopyasını <http://www.gnu.org/copyleft/gpl.html> adresinde bulabilirsiniz.

Linux, Linus Torvalds adına kayıtlı bir ticarî isimdir.

3.2. Feragatname

Bu belgedeki bilgilerin kullanımından doğacak sorumluluklar, ve olası zararlardan belge yazarı sorumlu tutulamaz. Bu belgedeki bilgileri uygulama sorumluluğu uygulayana aittir.

Tüm telif hakları aksi özellikle belirtilmediği sürece sahibine aittir. Belge içinde geçen herhangi bir terim bir ticarî isim ya da kuruma itibar kazandırma olarak algılanmamalıdır. Bir ürün ya da markanın kullanılmış olması ona onay verildiği anlamında görülmemelidir.

Notlar

Belge içinde dipnotlar ve dış bağlantılar varsa, bunlarla ilgili bilgiler bulundukları sayfanın sonunda dipnot olarak verilmeyip, hepsi toplu olarak burada listelenmiş olacaktır.

(B1) <http://www.ilkertemir.com/document/dns.html>

(B2) <http://www.internic.net>

(B3) <http://dns.metu.edu.tr>

(B5) <http://www.isc.org/products/BIND>

(B6) <ftp://ftp.rs.internic.net>

(B9) ../howto/gpl.pdf

Bu dosya (bind-ile-dns-howto.pdf), belgenin XML biçiminin \TeX Live ve belgeler-xsl paketlerindeki araçlar kullanılarak PDF biçimine dönüştürülmesiyle elde edilmiştir.

26 Ocak 2007