DNS HOGYAN

Nicolai Langfeldt (dns-howto[at]langfeldt.net), Jamie Norrish és mások v9.0, 2001.12.20

HOGYAN legyünk rövid idő alatt DNS-adminisztrátorok.

Contents

1 Előszó

Kulcsszavak: DNS, BIND, BIND 4, BIND 8, BIND 9, named, dialup, PPP, slip, ISDN, Internet, domain, name, resolution, hosts, caching.

Ez a dokumentum a Linux Dokumentációs Projekt része.

1.1 Szerzői jog

1.2 Köszönetnyilvánítások és segítségkérés

Meg szeretném köszönni mindenkinek, akit zavartam e HOGYAN olvasásával (ők tudják), és az összes olvasónak, akik javaslataikat és megjegyzéseiket elküldték levélben.

Ez soha nem lesz egy végleges dokumentum; kérlek, küldj egy levelet problémáidról és sikereidről. Ezzel jobbá teheted ezt a HOGYANt. Kérlek, a megjegyzéseidet és/vagy kérdéseidet vagy a pénzt küldd a janl@langfeldt.net <janl@langfeldt.net> címre. Vagy vedd meg a DNS könyvemet (a címe "The Concise Guide to DNS and BIND", az irodalomjegyzékben megtalálhatók az ISBN számok). Ha levelet küldesz, és szeretnél választ rá, kérlek, mutass egy kis udvariasságot azzal, hogy megbizonyosodsz róla, hogy a válaszcím helyes és működik. **Kérlek**, olvasd el a 10 (Kérdések és válaszok) fejezetet, mielőtt írsz nekem. Egy másik dolog, hogy csak norvégül és angolul értek.

Ez egy HOGYAN. 1995 óta tartottam karban, az LDP részeként. 2000 folyamán megírtam egy könyvet hasonló tárggyal. Szeretném elmondani, hogy bár ez a HOGYAN sok tekintetben olyan, mint egy könyv, ez *nem* a letisztázott, piacra készített könyvváltozat. Ezen HOGYAN olvasói segítettek annak felismerésében, hogy mi az, amit nehéz megérteni a DNS-ről. Ez segített a könyv megírásában, de a könyv szintén segített többet gondolkodnom azon, hogy ennek a HOGYANnak mire van szüksége. A HOGYAN hozta létre a könyvet. A könyv hozta létre e HOGYAN 3-as változatát. Köszönetem a könyvkiadónak, Que-nak, aki adott egy esélyt :-)

1.3 Ajánlás

Ajánlom ezt a HOGYANt Anne Line Norheim Langfeldt-nek. Bár ő valószínűleg soha sem fogja elolvasni, mert nem az a fajta lány.

1.4 Frissített változatok

Eme HOGYAN frissített változatait megtalálhatod a <http://langfeldt.net/DNS-HOWTO/>

és a <http://www.tldp.org/> oldalon is. Olvasd el azokat is, ha ez a dokumentum 9 hónapnál öregebb.

1.5 Magyar fordítás

A magyar fordítást *Füri Zoltán* <mailto:zfuri@avaya.com_NO_SPAM> készítette (2003.05.06). A lektorálást *Szíjjártó László* <mailto:laca@janus.gimsz.sulinet.hu_NO_SPAM> végezte el (2003.07.01). Bármilyen fordítással kapcsolatos észrevételt a *linuxhowto@sch.bme.hu* <mailto:linuxhowto@sch.bme.hu_NO_SPAM> címre küldjetek. Eme dokumentum legfrissebb változata megtalálható a *Magyar Linux Dokumentációs Projekt* <http://tldp.fsf.hu/index.html> honlapján.

2 Bevezetés

Mi ez, és mi nem

A DNS a Domain Name Server (Domain Név Szerver). A DNS átalakítja a gépneveket IP címekké, amellyel minden hálózati gép rendelkezik. A nevet címmé, és a címet névvé fordítja (vagy "mappeli", ahogy a zsargon hívja), és még egyéb feladatokat is ellát. Ez a HOGYAN azt dokumentálja, hogyan definiáljunk ilyen megfeleltetéseket Unix rendszer használatával, pár Linux-specifikus dologgal együtt.

A mappelés egy egyszerű megfeleltetés két dolog között, ez esetben egy gépnév, például `/ftp.linux.org/`, és a gép IP száma (vagy címe), `199.249.150.4` között. A DNS szintúgy tartalmazza a másik irányú megfeleltetést is IP számból gépnévvé; ennek neve "fordított megfeleltetés" (reverse mapping).

A DNS, a beavatatlanok számára (ez vagy te ;-), a hálózati adminisztráció egyik legködösebb területe. Szerencsére a DNS valójában nem ilyen nehéz. Ez a HOGYAN megpróbál egy pár dolgot világosabbá tenni. Leírja egy *egyszerű* DNS névszerver felállítását, kezdve egy csak gyorsítótáras szerverrel, és folytatva egy tartomány számára egy elsődleges DNS szerver felállításával. Bonyolultabb beállításokhoz átnézheted ezen dokumentum 10 (Kérdések és válaszok) fejezetét. Ha az nincs leírva ott, el kell *olvasnod* a Valódi Dokumentációt. Az 11 (utolsó fejezetben) visszatérek rá, mit is tartalmaz ez a Valódi Dokumentáció.

Mielőtt belekezdesz, be kell állítanod a gépedet, hogy be tudj rá, és ki tudj róla telnetelni, és sikerüljön mindenféle hálózati kapcsolatokat létrehozni, valamint különösen fontos, hogy képes legyél a `telnet 127.0.0.1` parancsot kiadni, és a saját gépedet elérni (próbáld ki most!). Kiindulásként szükséged lesz még jó, működő `/etc/nsswitch.conf/`, `/etc/resolv.conf/` és `/etc/hosts/` állományokra is, bár funkciójukat nem fogom itt elmagyarázni. Ha még nincs mindez beállítva és nem működik, a Networking-HOWTO (Hálózatok-HOGYAN) és/vagy a Networking-Overview-HOWTO (Hálózatok-Áttekintés-HOGYAN) elmagyarázza, hogyan kell ezeket beállítani. Olvasd el őket.

Amikor azt mondom "a te géped", arra a gépre gondolok, amelyiken a DNS-t próbálod beállítani, és nem akármelyik másik gépet, amely a hálózati környezetedben megtalálható.

Feltételezem, hogy nem vagy olyan tűzfal mögött, amely blokkolja a névlekérdezéseket. Ha mégis, különleges beállításokra lesz szükséged - lásd a 10 (Kérdések és válaszok) fejezetet.

A névszolgáltatást UNIX alatt a `named` program végzi. Ez része a "BIND" csomagnak, mely fejlesztését a *The Internet Software Consortium* koordinálja. A `named` programot tartalmazza a legtöbb Linux disztribúció, és általában `/usr/sbin/named` programként van telepítve, a csomag készítőjének hóbortjától függő kis- vagy nagybetűs `BIND` csomagból.

Ha van egy named programod, valószínűleg használhatod; ha nincs, beszerezhetsz egyet a Linux ftp oldalról, vagy letöltheted a legutolsó és legnagyszerűbb forráskódot az <ftp://ftp.isc.org/isc/bind9/> webhelyről. Ez a HOGYAN a 9-es verziójú BIND-ről szól. A HOGYAN régebbi változatai, a 4-es és 8-as verziójú BIND-ről, még mindig elérhetők a

<http://langfeldt.net/DNS-HOWTO/> honlapon, abban az esetben, ha 4-es vagy 8-as verziójú BIND-et használsz (mellékesen, ezt a HOGYANt is megtalálhatod ott). Ha a named kézikönyv oldala (man page) a `named.conf` állományról beszél (a legeslegvégén, a FILES (ÁLLOMÁNYOK) fejezetben), 8-as BIND-ed van; ha `named.boot` állományról van szó, 4-es BIND-ed van. Ha 4-esed van, és tudatosan a biztonságra törekszel, tényleg frissítened kell a 8-as BIND legfrissebb változatára. Most.

A DNS egy hálózati szintű adatbázis. Vigyázz, mit raksz bele. Ha szemetet raksz bele, te és mások is szemetet fognak kinyerni belőle. Tartsd DNS-ed rendben és konzisztensen, és egy jó szolgáltatást fogsz kapni. Tanuld meg használni, adminisztrálni, megkeresni hibáit, és egy újabb jó rendszergazda leszel, aki megvédi a hálózatot attól, hogy "megfeküdjön" a félremenedzselés miatt.

Tipp: Készíts biztonsági másolatot az összes állományról, amelynek megváltoztatására utasítalak, ha már megvannak, így ha esetleg semmi sem működik, visszajuthatsz a régi, működő állapotba.

2.1 Más névszerver megvalósítások

Ezt a fejezetet Joost van Baal írta.

Különböző csomagok léteznek DNS szerver telepítéshez a gépedre. Van a BIND csomag (<http://www.isc.org/products/BIND/>); a megvalósítás, amiről ez a HOGYAN szól. Ez a legnépszerűbb névszerver mindenfelé, és szerte az Interneten a névszolgáltató gépeinek döntő többségén ezt használják, és az 1980-as évek óta fejlesztik. A BSD licenc feltételei szerint használható. Mivel ez a legnépszerűbb programcsomag, egy csomó dokumentáció és tudásanyag található a BIND-ről mindenfelé. Azonban biztonsági problémák voltak vele.

Aztán van a djbdns (<http://djbdns.org/>), egy viszonylag új DNS csomag, amelyet Daniel J. Bernstein készített, aki a qmail programot is írta. Ez egy nagyon moduláris készlet: különböző kis programok gondoskodnak a különböző feladatokról, amit egy névszervernek kezelnie kell. A biztonság szempontjának figyelembe vételével tervezték. Egy egyszerűbb zóna-állomány formátumot használ, és általánosságban egyszerűbb beállítani. Azonban, mivel kevésbé ismert, a helyi guru nem biztos, hogy segíthet vele kapcsolatban. Sajnos ez a szoftver nem nyílt forráskódú. A szerző hirdetése a <http://cr.yp.to/djbdns/ad.html> honlapon található.

Hogy DJB szoftvere tényleg fejlődés-e a régi alternatívákkal szemben, sok vita tárgyát képezi. Az ISC csapata otthont ad egy beszélgetésnek (vagy inkább anyázásnak?) a BIND kontra djbdns-ről a

<http://www.isc.org/ml-archives/bind-users/2000/08/msg01075.html> honlapon.

3 A feloldó, gyorsítótáras névszerver

Az első ugrás a DNS beállításhoz. Nagyon hasznos betárcsázós, kábel-modemes, ADSL és hasonló felhasználók számára.

A Red Hat és a Red Hat-hoz kapcsolódó disztribúciók esetén ezen HOGYAN első fejezetéhez hasonló gyakorlati eredmény érhető el a `bind`, `bind-utils` és `caching-nameserver` csomagok telepítésével. Ha Debiant

használsz, egyszerűen csak telepítsd a `bind` (vagy a `bind9` csomagot, mivel jelenleg a BIND 9-est nem támogatja a Debian Stable (potato)) és a `bind-doc` csomagot. Persze csak ezen csomagok telepítésével nem tanulsz annyit, mint e HOGYAN olvasásával. Szóval telepítsd a csomagokat, azután olvass tovább, és ellenőrizd az általuk telepített állományokat.

A gyorsítótáras névszerver megtalálja a választ a névlekérdezésekre, és megjegyzi a választ a legközelebbi alkalomig, amikor szükséged lesz rá. Ez jelentősen le fogja rövidíteni a várakozási időt a következő alkalommal, különösen ha lassú a kapcsolatod.

Először szükséged lesz egy `/etc/named.conf` nevű állományra (Debianban: `/etc/bind/named.conf`). Ez betöltődik amikor a named elindul. Egyelőre csak ezt kell tartalmaznia:

```
// Konfigurációs állomány kizárólag gyorsítótáras névszerver számára
//
// A HOGYAN ezen változata tartalmazhat a sor elején szóközöket
// tartalmazó sorokat ebben és más állományokban. El kell távolítanod
// a szóközöket, hogy bizonyos dolgok működjenek.
//
// Figyelem, az állománynevek és a könyvtárak nevei különbözhetnek, ám
// a lényegi tartalmuknak hasonlónak kell lenniük.

options {
        directory "/var/named";
        // E sor engedélyezése segíthet, ha tűzfalon keresztül kell
        // átmenned, és a dolog nem működik. De valószínűleg beszélned
        // kell a tűzfal adminisztrátorával.
        // query-source port 53;
};

controls {
        inet 127.0.0.1 allow { localhost; } keys { rndc_key; };
};

key "rndc_key" {
        algorithm hmac-md5;
        secret "c3Ryb25nIGVub3VnaCBmb3IgYSBtYW4gYnV0IG1hZGUgZm9yIGEgd29tYW4K";
};

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

zone "0.0.127.in-addr.arpa" {
        type master;
        file "pz/127.0.0";
};
```

A Linux disztribúciós csomagok eltérő állományneveket használhatnak minden egyes, itt említett állománytípusra; azonban közel ugyanazt fogják tartalmazni.

A "`directory`" sor megmondja a named programnak, hol keresse az állományokat. Minden ezután megnevezett állomány ehhez lesz viszonyítva. Tehát a `pz` egy könyvtár a `/var/named` alatt, azaz megegyezik a

`/var/named/pz` könyvtárral. A `/var/named` a helyes könyvtár, a *Linux Fájlrendszer Szabvány* alapján.

A `/var/named/root.hints` állomány is megemlítődik benne. A `/var/named/root.hints` állománynak ezt kell tartalmaznia:

```
;
; Nyitó megjegyzések lehetnek itt, ha már megvan ez az állományod.
; Ha nem, ne aggódj.
;
; A kezdő szóközökről a sorok elején: távolítsd el őket!
; A sornak egy ;-vel, .-tal vagy betűvel kell kezdődniük, nem szóközzel.
;
.                       6D  IN      NS      A.ROOT-SERVERS.NET.
.                       6D  IN      NS      B.ROOT-SERVERS.NET.
.                       6D  IN      NS      C.ROOT-SERVERS.NET.
.                       6D  IN      NS      D.ROOT-SERVERS.NET.
.                       6D  IN      NS      E.ROOT-SERVERS.NET.
.                       6D  IN      NS      F.ROOT-SERVERS.NET.
.                       6D  IN      NS      G.ROOT-SERVERS.NET.
.                       6D  IN      NS      H.ROOT-SERVERS.NET.
.                       6D  IN      NS      I.ROOT-SERVERS.NET.
.                       6D  IN      NS      J.ROOT-SERVERS.NET.
.                       6D  IN      NS      K.ROOT-SERVERS.NET.
.                       6D  IN      NS      L.ROOT-SERVERS.NET.
.                       6D  IN      NS      M.ROOT-SERVERS.NET.
A.ROOT-SERVERS.NET.     6D  IN      A       198.41.0.4
B.ROOT-SERVERS.NET.     6D  IN      A       128.9.0.107
C.ROOT-SERVERS.NET.     6D  IN      A       192.33.4.12
D.ROOT-SERVERS.NET.     6D  IN      A       128.8.10.90
E.ROOT-SERVERS.NET.     6D  IN      A       192.203.230.10
F.ROOT-SERVERS.NET.     6D  IN      A       192.5.5.241
G.ROOT-SERVERS.NET.     6D  IN      A       192.112.36.4
H.ROOT-SERVERS.NET.     6D  IN      A       128.63.2.53
I.ROOT-SERVERS.NET.     6D  IN      A       192.36.148.17
J.ROOT-SERVERS.NET.     6D  IN      A       198.41.0.10
K.ROOT-SERVERS.NET.     6D  IN      A       193.0.14.129
L.ROOT-SERVERS.NET.     6D  IN      A       198.32.64.12
M.ROOT-SERVERS.NET.     6D  IN      A       202.12.27.33
```

Ez az állomány írja le a fő névszervereket a világban. A szerverek időről időre változnak, és frissíteni kell őket most és később is. A 8 (Karbantartás) fejezetben olvashatsz ezek naprakészen tartásáról.

A következő rész a `named.conf` állományban a `zone` (zóna). Használatát egy későbbi fejezetben fogom elmagyarázni; most csak nevezzük el ezt az állományt `127.0.0`-nak a `pz` alkönyvtárban. *(Újfent, kérlek távolítsd el a sor eleji szóközöket, ha kivágod és beilleszted ezt.)*

```
$TTL 3D
@               IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. (
                                1       ; Serial
                                8H      ; Refresh
                                2H      ; Retry
```

```
                                4W      ; Expire
                                1D)     ; Minimum TTL
                        NS      ns.linux.bogus.
1                       PTR     localhost.
```

A `key` és a `control` részek azt határozzák meg, hogy a named programod távolról irányítható az `rndc` programmal ha egy helyi állomásról kapcsolódik, ekkor egy kódolt titkos kulccsal azonosítja magát. Ez a kulcs olyan, mint egy jelszó. Az rndc működéséhez az `/etc/rndc.conf` állománynak meg kell egyeznie ezzel:

```
key rndc_key {
    algorithm "hmac-md5";
    secret "c3Ryb25nIGVub3VnaCBmb3IgYSBtYW4gYnV0IG1hZGUgZm9yIGEgd29tYW4K";
};

options {
    default-server localhost;
    default-key    rndc_key;
};
```

Amint látod, a secret bejegyzések megegyeznek. Ha az `rndc` programot egy másik gépről szeretnéd használni, a két gép egymáshoz viszonyított rendszeridejének 5 percen belül kell lennie. Ehhez ajánlom az ntp (`xntpd` és `ntpdate`) szoftvert.

És most, szükséged lesz egy ehhez hasonló `/etc/resolv.conf` állományra: *(Újfent: Távolítsd el a szóközöket!)*

```
search altartomány.a-te-tartományod.edu a-te-tartományod.edu
nameserver 127.0.0.1
```

A "`search`" sor határozza meg, milyen tartományban történjen a keresés az állomások után, amelyekhez kapcsolódni akarsz. A "`nameserver`" sor határozza meg a névszervered címét, ebben az esetben a saját gépedet, mert ez az, ahol a named programod fut (a 127.0.0.1 cím helyes, nem számít, ha a gépednek van egy másik címe is). Ha több névszervert akarsz felsorolni, rakd mindegyiket egy-egy "`nameserver`" sorba. (Megjegyzés: A named soha nem olvassa el ezt az állományt, a named programot használó feloldó teszi ezt. Megjegyzés 2: Néhány resolv.conf állományban a "domain" sort találod. Ez helyes, de ne használd a "search" és a "domain" kulcsszót is egyszerre, csak az egyikük fog működni.)

Annak bemutatására, hogy ez az állomány mit csinál: Ha az ügyfél megpróbálja kikeresni a `foo`-t, akkor a `foo.altartomány.a-te-tartományod.edu`-t próbálja először, majd a `foo.a-te-tartományod.edu`-t, és végül a `foo`-t. Ne akarj túl sok tartományt rakni a keresősorba, mivel mindet végigkeresni időt vesz igénybe.

A példa feltételezi, hogy az `altartomány.a-te-tartományod.edu` tartományba tartozol. A keresősornak nem szabad tartalmaznia a legfelső tartományodat (TLD - Top Level Domain), ebben az esetben az "`edu`"-t. Ha gyakran kell kapcsolódnod másik tartományban levő állomásokhoz, hozzáadhatod azt a tartományt a keresősorhoz, így: *(Ne felejtsd el eltávolítani a szóközöket a sor elején, ha vannak)*

```
search altartomány.a-te-tartományod.edu a-te-tartományod.edu másik-tartomány.com
```

és így tovább. Nyilvánvalóan valódi tartományneveket kell helyettük beraknod. Kérlek figyeld meg a tartománynevek végén a pontok hiányát. Ez fontos!

3.1 A named indítása

Mindezek után itt az idő a named indítására. Ha betárcsázós kapcsolatot használsz, először csatlakozz. Most indítsd a named-et, vagy a boot szkript futtatásával: `/etc/init.d/named start`, vagy a named-et közvetlenül: `/usr/sbin/named`. Ha kipróbáltad a BIND előző verzióit, valószínűleg az ndc-t használtad. A BIND 9-ben ezt az `rndc` program váltotta fel, ami távolról vezérelheti a named-et, de már nem tudja a named-et indítani. Ha megnézed a rendszerüzenetek naplóállományát (általában `/var/log/messages`, a Debianban `/var/log/daemon`, meg lehet még keresni a `/var/log` egy másik állományában is), mialatt indítod a named-et (ezt a `tail -f /var/log/messages`-el teheted meg), valami ilyesmit kell látnod:

(a \-el végződő sorok a következő sorban folytatódnak)

```
Dec 23 02:21:12 lookfar named[11031]: starting BIND 9.1.3
Dec 23 02:21:12 lookfar named[11031]: using 1 CPU
Dec 23 02:21:12 lookfar named[11034]: loading configuration from \
    '/etc/named.conf'
Dec 23 02:21:12 lookfar named[11034]: the default for the \
    'auth-nxdomain' option is now 'no'
Dec 23 02:21:12 lookfar named[11034]: no IPv6 interfaces found
Dec 23 02:21:12 lookfar named[11034]: listening on IPv4 interface lo, \
    127.0.0.1#53
Dec 23 02:21:12 lookfar named[11034]: listening on IPv4 interface eth0, \
    10.0.0.129#53
Dec 23 02:21:12 lookfar named[11034]: command channel listening on \
    127.0.0.1#953
Dec 23 02:21:13 lookfar named[11034]: running
```

Ha bármilyen hibaüzenet megjelenik, akkor ott hiba van. A named megnevezi az állományt, amit épp olvas. Menj vissza, és ellenőrizd le az állományt. Indítsd újból a named-et, ha megjavítottad.

Most letesztelheted a beállításodat. Hagyományosan az `nslookup` használatos erre. Napjainkban azonban már a `dig` ajánlott:

```
$ dig -x 127.0.0.1
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26669
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;1.0.0.127.in-addr.arpa.                IN      PTR

;; ANSWER SECTION:
1.0.0.127.in-addr.arpa. 259200  IN      PTR     localhost.

;; AUTHORITY SECTION:
0.0.127.in-addr.arpa.   259200  IN      NS      ns.linux.bogus.

;; Query time: 3 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 02:26:17 2001
;; MSG SIZE  rcvd: 91
```

Ha ilyen üzeneteket kaptál, akkor működik. Reméljük. Ha bármi teljesen eltérőt kapsz, menj vissza, és ellenőrizz le mindent. Minden alkalommal, amikor megváltoztatsz egy állományt, futtasd az `rndc reload` parancsot.

Most már beadhatsz egy lekérdezést. Próbálj meg valami hozzád közeli gépet. A `pat.uio.no` közel van hozzám, az Oslói Egyetemen:

```
$ dig pat.uio.no
; <<>> DiG 9.1.3 <<>> pat.uio.no
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15574
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 0

;; QUESTION SECTION:
;pat.uio.no.                    IN      A

;; ANSWER SECTION:
pat.uio.no.             86400   IN      A       129.240.130.16

;; AUTHORITY SECTION:
uio.no.                 86400   IN      NS      nissen.uio.no.
uio.no.                 86400   IN      NS      nn.uninett.no.
uio.no.                 86400   IN      NS      ifi.uio.no.

;; Query time: 651 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 02:28:35 2001
;; MSG SIZE  rcvd: 108
```

Ezúttal a `dig` megkérte a named-et, hogy keresse meg a `pat.uio.no` gépet. Az pedig kapcsolódott a `root.hints` állományodban levő egyik névszerver géphez, és lekérdezte az útvonalát onnan. Eltarthat egy röpke pillanatig, míg megkapod az eredményt, mivel végig kell keresnie az összes tartományt, amit a `/etc/resolv.conf`-ban megneveztél.

Ha még egyszer lekérdezed ugyanazt, ezt kapod:

```
$ dig pat.uio.no

; <<>> DiG 8.2 <<>> pat.uio.no
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3, ADDITIONAL: 3
;; QUERY SECTION:
;;      pat.uio.no, type = A, class = IN

;; ANSWER SECTION:
pat.uio.no.             23h59m58s IN A  129.240.130.16

;; AUTHORITY SECTION:
UIO.NO.                 23h59m58s IN NS  nissen.UIO.NO.
UIO.NO.                 23h59m58s IN NS  ifi.UIO.NO.
UIO.NO.                 23h59m58s IN NS  nn.uninett.NO.

;; ADDITIONAL SECTION:
nissen.UIO.NO.          23h59m58s IN A  129.240.2.3
ifi.UIO.NO.             1d23h59m58s IN A  129.240.64.2
nn.uninett.NO.          1d23h59m58s IN A  158.38.0.181
```

```
;; Total query time: 4 msec
;; FROM: lookfar to SERVER: default -- 127.0.0.1
;; WHEN: Sat Dec 16 00:23:09 2000
;; MSG SIZE  sent: 28  rcvd: 162
```

Ahogy azt nyilvánvalóan láthatod, ezúttal ez sokkal gyorsabb volt, 4 ms a korábbi több, mint fél másodperccel ellentétben. A válasz benne volt a gyorsítótárban. A gyorsítótárban lévő eredményeknél esély van arra, hogy már elavult, de az eredeti szerverek befolyásolhatják azt az időt, amíg a letárolt válaszok érvényesként lesznek nyilvántartva. Végül is nagy a valószínűség arra, hogy a kapott válasz *érvényes*.

3.2 Névfeloldás

Minden operációs rendszer, ami a C API szabványt alkalmazza, rendelkezik a gethostbyname és a gethostbyaddr hívásokkal. Ezek különböző forrásokból szerezhetik be az információt. Hogy melyik forrásból szerzik ezt be, az Linux (és egyes Unix) rendszereken az `/etc/nsswitch.conf` állományban van beállítva. Ez egy hosszú állomány, amely megadja mely állományokból vagy adatbázisokból szerezhetők be különböző adattípusok. Általában hasznos megjegyzéseket tartalmaz a fejlécében, melyeket körültekintően olvass el. Ezután keresd meg a "`hosts:`" kulcsszóval kezdődő sort; így kell kinéznie:

```
hosts:      files dns
```

(*Emlékszel még a szóközökre a sor elején? Nem akarom újra megemlíteni.*)

Ha nincs "`hosts:`" kulcsszóval kezdődő sor, szúrd be a fentieket. Ezek a sorok azt jelentik, hogy a programoknak először a `/etc/hosts` állományban kell keresniük, majd leellenőrzik a DNS-t a `resolv.conf` állomány alapján.

3.3 Gratulálok

Most már tudod, hogyan kell beállítani a gyorsítótáras named-et. Bonts egy sört, tejet, vagy bármit, amivel ünnepelni szeretsz.

4 Továbbítás (forwarding)

Nagy, jól szervezett, egyetemi vagy Internet szolgáltatói (ISP) hálózatokban néha megfigyelheted, hogy a hálózati szakemberek a DNS szerverek továbbítói hierarchiáját hozták létre, ami segít a belső hálózati terhelés csökkentésében, és a külső szerverekén úgyszintén. Nem könnyű megtudni, hogy egy ilyen hálózatban vagy-e. De ha a hálózati szolgáltatód DNS szerverét "továbbítóként" használod, a lekérdezésekre adott reakciókat gyorsabbá teheted, és csökkentheted a forgalmat a hálózatodon. Ez a te névszervered lekérdezéseinek az ISP névszervere felé történő továbbításával működik. Minden egyes alkalommal, amikor ilyen történik, az ISP névszerverének nagy gyorsítótárába nyúlsz bele, így felgyorsítva a lekérdezéseket, névszerverednek pedig nem kell mindent magának végeznie. Ha modemet használsz ez nagy előny lehet. A példa kedvéért tételezzük fel, hogy a hálózati szolgáltatódnak két névszervere van amiket használni akarsz, `10.0.0.1` és `10.1.0.1` IP címekkel. Ebben az esetben a `named.conf` állományodba, az "`options`" kulcsszóval kezdődő részbe szúrd be ezeket a sorokat:

```
        forward first;
        forwarders {
```

```
        10.0.0.1;
        10.1.0.1;
    };
```

Van még egy szép trükk a továbbítókat használó betárcsázós gépek számára, amely a 10 (Kérdések és válaszok) fejezetben van leírva.

Indítsd újra a névszerveredet, és teszteld a `dig`-el. Még mindig rendben kell működnie.

5 Egy *egyszerű* tartomány

Hogyan kell felállítani a saját tartományodat?

5.1 De először egy kis száraz elmélet

Mindenekelőtt: elolvastad az összes cuccot ez előtt, ugye? Erre szükség van.

Mielőtt *tényleg* elkezdjük ezt a fejezetet, közzéteszek egy kis elméletet, és egy példát, hogyan működik a DNS. És te el fogod olvasni, mert az jó neked. Ha nem akarod, legalább fusd át nagyon gyorsan. Fejezd be a futást, ha oda érsz, hogy minek kell a `named.conf` állományodba kerülnie.

A DNS egy hierarchikus, fa struktúrájú rendszer. A tetejét "."-nak írják és "gyökér"-nek (root) ejtik, ahogy az megszokott a fa-típusú adatstruktúráknál. A . alatt számos legfelsőbb szintű tartomány (TLD - Top Level Domain) van; a legismertebbek az `ORG`, `COM`, `EDU` és a `NET`, de még sok más is van. Éppúgy mint a fának, ennek is van gyökere és elágazik. Ha van egy kis számítástechnikai háttered, a DNS-t, mint egy keresőfát azonosíthatod, és megtalálhatod a csomópontokat, az ágakat és a csúcsokat. A pontok a csomópontok, a csúcsok a neveken vannak.

Egy gép keresésekor a lekérdezés rekurzív módon halad a hierarchiában, a gyökértől kiindulva. Ha a `prep.ai.mit.edu` címét akarod megtalálni, a névszerverednek el kell kezdenie valahol. A gyorsítótárban való kereséssel kezdi. Ha ebben megvan a válasz mert korábban eltárolta, azonnal válaszolni fog, ahogy ezt a legutóbbi fejezetben láttuk. Ha nem tudja, megnézi milyen közeli választ tud adni a keresett névhez, és felhasznál bármilyen információt, amit már eltárolt. A legrosszabb esetben nincs más találata, csak a név "."-ja (gyökere), és a főszerverekhez kell fordulni. El fogja távolítani a baloldali részeket, egyenként ellenőrizve, hogy tud-e valamit az `ai.mit.edu.` tartományról, utána a `mit.edu.`-ról, utána az `edu.`-ról, és ha nem, utána a .-ról, mert ez volt a hints állományban. Ezután megkérdezi a . szervert a `prep.ai.mit.edu` tartományról. Ez a . szerver nem fogja tudni a választ, de segíteni fog a szerverednek a saját módján egy hivatkozás megadásával, amellyel megmondja, hol keressen inkább. Ezek a hivatkozások a szerveredet végül ahhoz a névszerverhez vezetik, amelyik tudja a választ. Most ezt fogom bemutatni. A `+norec` azt jelenti, hogy a dig egy nem-rekurzív lekérdezést végez, így a rekurziót magunknak kell elvégeznünk. A többi opció a dig folyamat csökkentésére vannak, így ez nem fog több oldalon át futni:

```
$ ;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 980
;; flags: qr ra; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL: 0

;; AUTHORITY SECTION:
.                       518400  IN      NS      J.ROOT-SERVERS.NET.
.                       518400  IN      NS      K.ROOT-SERVERS.NET.
.                       518400  IN      NS      L.ROOT-SERVERS.NET.
.                       518400  IN      NS      M.ROOT-SERVERS.NET.
```

```
.                       518400  IN      NS      A.ROOT-SERVERS.NET.
.                       518400  IN      NS      B.ROOT-SERVERS.NET.
.                       518400  IN      NS      C.ROOT-SERVERS.NET.
.                       518400  IN      NS      D.ROOT-SERVERS.NET.
.                       518400  IN      NS      E.ROOT-SERVERS.NET.
.                       518400  IN      NS      F.ROOT-SERVERS.NET.
.                       518400  IN      NS      G.ROOT-SERVERS.NET.
.                       518400  IN      NS      H.ROOT-SERVERS.NET.
.                       518400  IN      NS      I.ROOT-SERVERS.NET.
```

Ez egy hivatkozás. Ez csak egy felügyeleti részt ("Authority section") hoz létre nekünk, válasz részt ("Answer section") pedig nem. A saját névszerverünk egy névszerverhez küld tovább. Válasszunk ki véletlenszerűen egyet:

```
$ dig +norec +noques +nostats +nocmd prep.ai.mit.edu. @D.ROOT-SERVERS.NET.
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58260
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 3, ADDITIONAL: 3

;; AUTHORITY SECTION:
mit.edu.                172800  IN      NS      BITSY.mit.edu.
mit.edu.                172800  IN      NS      STRAWB.mit.edu.
mit.edu.                172800  IN      NS      W20NS.mit.edu.

;; ADDITIONAL SECTION:
BITSY.mit.edu.          172800  IN      A       18.72.0.3
STRAWB.mit.edu.         172800  IN      A       18.71.0.151
W20NS.mit.edu.          172800  IN      A       18.70.0.160
```

Ez azonnal a MIT.EDU szerverhez küld minket. Újra válasszuk ki egyet véletlenszerűen:

```
$ dig +norec +noques +nostats +nocmd prep.ai.mit.edu. @BITSY.mit.edu.
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29227
;; flags: qr ra; QUERY: 1, ANSWER: 1, AUTHORITY: 4, ADDITIONAL: 4

;; ANSWER SECTION:
prep.ai.mit.edu.        10562   IN      A       198.186.203.77

;; AUTHORITY SECTION:
ai.mit.edu.             21600   IN      NS      FEDEX.ai.mit.edu.
ai.mit.edu.             21600   IN      NS      LIFE.ai.mit.edu.
ai.mit.edu.             21600   IN      NS      ALPHA-BITS.ai.mit.edu.
ai.mit.edu.             21600   IN      NS      BEET-CHEX.ai.mit.edu.

;; ADDITIONAL SECTION:
FEDEX.ai.mit.edu.       21600   IN      A       192.148.252.43
LIFE.ai.mit.edu.        21600   IN      A       128.52.32.80
ALPHA-BITS.ai.mit.edu.  21600   IN      A       128.52.32.5
BEET-CHEX.ai.mit.edu.   21600   IN      A       128.52.32.22
```

Ezúttal kapunk egy "ANSWER SECTION"-t, és választ a kérdésünkre. Az "AUTHORITY SECTION" azt az információt tartalmazza, hogy mely szervereket kérdezzük legközelebb az `ai.mit.edu`-ról. Így, következő

alkalommal amikor az `ai.mit.edu` nevekről kíváncsiskodsz, közvetlenül őket kérdezheted. A named információt gyűjtött a `mit.edu`-ról is, így legközelebb ha a `www.mit.edu` lekérdezése fordul elő, sokkal könnyebb lesz majd megválaszolni a kérdést.

Így a .-tól kezdődően a hivatkozások alapján megtaláltuk az egymás utáni névszervereket, a tartománynév minden egyes szintjéhez. Ha a saját DNS szerveredet használtad volna mindezen szerverek helyett, a named-ed természetesen eltárolta volna mindezt az információt amit a kutakodás során talált, és egy ideig nem kellene újra lekérdeznie.

A fa-analógiában minden "." a névben egy elágazási pont, és minden rész a "."-ok között az egyes ágak nevei a fán. A fa bejárásakor fogjuk a nevet amit keresünk (prep.ai.mit.edu), megkérdezve a gyökeret (.) vagy bármelyik szervert a gyökértől a `prep.ai.mit.edu` felé, amelyikről van információnk a gyorsítótárban. Ha a gyorsítótár eléri kapacitásának határait, a rekurzív feloldó a külső szervereket kérdezi le, követve a hivatkozásokat (éleket) tovább a névben.

Valamivel kevesebbet beszéltünk róla, de éppoly fontos az `in-addr.arpa` tartomány. Ez is épp úgy szervezett, mint a "közönséges" tartományok. Az `in-addr.arpa` lehetővé teszi számunkra, hogy megkapjuk az állomás nevét, ha megvan a címe. Egy fontos dolog, amit meg kell jegyezni, hogy az IP címek fordított sorrendben vannak írva az `in-addr.arpa` tartományban. Ha egy gépnek a címe: `198.186.203.77`, a named a keresést a `77.203.168.198.in-addr.arpa`-ra végzi, éppúgy, ahogy azt a `prep.ai.mi.edu`-ra tette. Példa: Ha nem találsz egyetlen találati bejegyzést a gyorsítótárban csak a "."-ot, kérdezz le egy főszervert, az `m.root-servers.net` valamelyik másik főszerverhez irányít. A `b.root-servers.net` közvetlenül a `bitsy.mit.edu` tartományhoz irányít. Onnan már képes leszel leszedni.

5.2 A saját tartományunk

Most következik a saját tartományunk meghatározása. A `linux.bogus` tartományt fogjuk létrehozni, és megadjuk a gépeket benne. Egy teljesen hamis tartománynevet használok, hogy biztos ne zavarjunk senkit Ott Kint.

Még egy dolog, mielőtt elkezdjük: Nem minden karakter megengedett a gépnevekben. Az angol ábécé betűire vagyunk korlátozva: a-z, és a 0-9 számok és a "-" (kötőjel) karakter. Tartsd magad ezekhez a karakterekhez (a 9-es BIND nem fog hibásan működni, ha megszeged ezt a szabályt, de a 8-as BIND igen). A kis- és nagybetűk egyformák a DNS számára, tehát a `pat.uio.no` megegyezik a `Pat.UiO.No`-val.

Már elkezdtük ezt a részt a named.conf-ban ezzel a sorral:

```
zone "0.0.127.in-addr.arpa" {
        type master;
        file "pz/127.0.0";
};
```

Kérlek, vedd észre a "." hiányát a tartománynevek végén ebben az állományban. Azt jelenti, hogy mi most a `0.0.127.in-addr.arpa` zónát fogjuk megadni, hogy mi vagyunk a mesterszerver számára, és hogy a `pz/127.0.0` állományban van tárolva. Már beállítottuk ezt az állományt:

```
$TTL 3D
@               IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. (
                                1       ; Serial
                                8H      ; Refresh
                                2H      ; Retry
                                4W      ; Expire
                                1D)     ; Minimum TTL
```

```
                        NS      ns.linux.bogus.
1                       PTR     localhost.
```

Kérlek, vedd észre a "."-ot a teljes tartománynevek végén ebben az állományban, ellentétben a fenti `named.conf` állománnyal. Egyesek szeretnek minden zónaállományt az `//$ORIGIN` direktívával kezdeni, de ez felesleges. Egy zónaállomány eredete (ahová a DNS hierarchiában tartozik) a `named.conf` állomány zóna fejezetében van meghatározva; ebben az esetben ez a `0.0.127.in-addr.arpa`.

Ez a "zónaállomány" 3 "erőforrásbejegyzést" (RR - resource record) tartalmaz: egy SOA RR-t, egy NS RR-t és egy PTR RR-t. A SOA a Jogosultság Kezdetének a rövidítése (SOA - Start Of Authority). A "@" egy speciális jel ami az eredetet jelenti, és mivel a "tartomány" oszlop ezen állomány esetén az 0.0.127.in-addr-arpa-t tartalmazza, az első sor valójában ezt jelenti:

```
0.0.127.in-addr.arpa.   IN      SOA ...
```

Az NS a Névszerver RR. Itt nincs "@" a sor elején; magától értetődő, mivel az előző sor egy "@"-el kezdődött. Ez megtakarít egy kis gépelést. Tehát az NS sort így is lehet írni:

```
0.0.127.in-addr.arpa.   IN      NS      ns.linux.bogus
```

Ez megmondja a DNS-nek, melyik gép a `0.0.127.in-addr.arpa` tartomány névszervere, ez az `ns.linux.bogus`. Az "ns" egy szokványos név a névszerverek számára, éppúgy, mint a web szerverek esetében, amiknek szokványosan `www.`*valami* a nevük. A név bármi lehet.

Végül a PTR (Tartomány Név Mutató) bejegyzés megmondja, hogy a `0.0.127.in-addr.arpa` alhálózat 1-es címén, azaz a 127.0.0.1 címen található gép neve `localhost`.

A SOA bejegyzés a bevezető az *összes* zónaállományhoz, és pontosan egynek kell lennie minden egyes zónaállományban, a tetején (de a $TTL direktíva után). Ez leírja a zónát, honnan származik (egy `ns.linux.bogus` nevű gépről), ki felelős annak tartalmáért (`hostmaster@linux.bogus`, a saját e-mail címedet kell ideírnod), melyik változatú zónaállomány ez (serial: 1), és egyéb, a gyorsítótárazással és a másodlagos DNS szerverekkel kapcsolatos dolgokat. A maradék mezők (refresh - frissítés, retry - újrapróbálkozás, expire - lejárat és minimum) tekintetében használd az ebben a HOGYANban használt számokat, és nem lesz baj. A SOA elé jön egy kötelező sor, a `$TTL 3D`. Rakd bele az összes zónaállományodba.

Most indítsd újra a named-et (`rndc stop; named`) és használd a `dig`-et ügyeskedésed megvizsgálásához. A `-x` fordított lekérdezést kér:

```
$ dig -x 127.0.0.1
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 30944
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;1.0.0.127.in-addr.arpa.                IN      PTR

;; ANSWER SECTION:
1.0.0.127.in-addr.arpa. 259200  IN      PTR     localhost.

;; AUTHORITY SECTION:
0.0.127.in-addr.arpa.   259200  IN      NS      ns.linux.bogus.

;; Query time: 3 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 03:02:39 2001
;; MSG SIZE  rcvd: 91
```

Szóval ez gondoskodik arról, hogy a 127.0.0.1-ből `localhost`-ot kapjunk; rendben. Most a fő célunk, a linux.bogus tartomány érdekében, szúrjunk be egy új "zone" részt a `named.conf` állományba:

```
zone "linux.bogus" {
        type master;
        notify no;
        file "pz/linux.bogus";
};
```

Figyeld meg újból a `named.conf` állományban a tartománynév végén a "." hiányát.

A `linux.bogus` zónaállománya berakunk némi teljesen valótlan adatot:

```
;
; Zone file for linux.bogus
;
; The full zone file
;
$TTL 3D
@       IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. (
                        199802151       ; serial, todays date + todays serial #

                        8H              ; refresh, seconds
                        2H              ; retry, seconds
                        4W              ; expire, seconds
                        1D )            ; minimum, seconds
;
                NS      ns              ; Inet Address of name server
                MX      10 mail.linux.bogus     ; Primary Mail Exchanger
                MX      20 mail.friend.bogus.   ; Secondary Mail Exchanger
;
localhost       A       127.0.0.1
ns              A       192.168.196.2
mail            A       192.168.196.4
```

Két dolgot meg kell jegyezni a SOA bejegyzésről. Az `ns.linux.bogus`-nak egy "A" bejegyzéssel rendelkező valódi gépnek *kell* lennie. Nem megengedett az SOA bejegyzésben említett géphez CNAME bejegyzést rendelni. A nevének nem kell "ns"-nek lennie, bármely valós gép neve lehet. Az ezt követő `hostmaster.linux.bogus`-t hostmaster@linux.bogus-nak kell olvasni. Ennek egy olyan levélcímnek kell lennie, amelyet a DNS-t karbantartó személy, vagy személyek gyakran olvasnak. Bármely, a tartománnyal kapcsolatos levél az itt megadott címre lesz elküldve. A névnek nem kell "hostmaster"-nek lennie, lehet ez a rendes e-mail címed, de a "hostmaster" e-mail cím létezése sokszor szintén elvárás.

Egy új RR típus található ebben az állományban, az MX, vagy a Mail eXchanger (levélkiszolgáló) RR. Ez megmondja a levelezőrendszereknek, hova legyen küldve a `valaki@linux.bogus`-nak címzett levél, név szerint a `mail.linux.bogus`-nak, vagy a `mail.friend.bogus`-nak. A szám minden gép neve előtt az adott MX RR prioritása. A legkisebb számmal (10) rendelkező RR az, amelyik, ha lehetséges a levelet kapni fogja. Ha ez nem sikerül, a levelet el lehet küldeni egy magasabb számmal rendelkezőnek, egy másodlagos levélkezelőnek, azaz a `mail.friend.bogus`-nak, amelynek a prioritása itt 20.

Töltsük be a tartományokat újból az `rndc reload` futtatásával. Vizsgáljuk meg az eredményeket a `dig`-el:

```
$ dig any linux.bogus
; <<>> DiG 9.1.3 <<>> any linux.bogus
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 55239
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;linux.bogus.               IN      ANY

;; ANSWER SECTION:
linux.bogus.        259200  IN      SOA     ns.linux.bogus. \
      hostmaster.linux.bogus. 199802151 28800 7200 2419200 86400
linux.bogus.        259200  IN      NS      ns.linux.bogus.
linux.bogus.        259200  IN      MX      20 mail.friend.bogus.
linux.bogus.        259200  IN      MX      10 mail.linux.bogus.linux.bogus.

;; AUTHORITY SECTION:
linux.bogus.        259200  IN      NS      ns.linux.bogus.

;; ADDITIONAL SECTION:
ns.linux.bogus.     259200  IN      A       192.168.196.2

;; Query time: 4 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 03:06:45 2001
;; MSG SIZE  rcvd: 184
```

Alapos vizsgálat után egy hibát fogsz találni. A

```
linux.bogus.        259200  IN MX        10 mail.linux.bogus.linux.bogus.
```

sor teljesen rossz. Ennek így kellene kinéznie:

```
linux.bogus.        259200  IN MX        10 mail.linux.bogus.
```

Szándékosan hibát vétettem, úgyhogy tanulhatsz belőle :-) Beletekintve a zónaállományba ezt a sort találjuk:

```
                MX      10 mail.linux.bogus     ; Primary Mail Exchanger
```

Hiányzik egy pont. Vagy a "linux.bogus"-ban túl sok van. Ha egy gépnév a zónaállományban nem végződik pontra, az eredete hozzáadódik a végéhez, a megduplázott `linux.bogus.linux.bogus`-t eredményezve. Szóval vagy

```
                MX      10 mail.linux.bogus.    ; Primary Mail Exchanger
```

vagy

```
                MX      10 mail                 ; Primary Mail Exchanger
```

a helyes. Én az utóbbi változatot preferálom, kevesebbet kell gépelni. Vannak olyan BIND szakértők, akik nem értenek egyet ezzel, és vannak olyanok akik igen. Egy zónaállományban a tartományt vagy ki kell írni, és "."-al lezárni, vagy egyáltalán nem kell meghatározni, mely esetben az eredet lesz az alapértelmezés.

Ki kell hangsúlyoznom, hogy a named.conf állományban *nem* kell "."-nak lennie a tartománynevek után. El sem bírod képzelni, hány esetben kavarta össze a dolgokat a túl sok vagy túl kevés pont, és hozta ki az ördögöt az emberekből.

Szóval, kifejtve érveimet itt van az új zónaállomány, némi extra információval kiegészítve:

```
;
; Zone file for linux.bogus
;
; The full zone file
;
$TTL 3D
@       IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. (
                        199802151       ; serial, todays date + todays serial #
                        8H              ; refresh, seconds
                        2H              ; retry, seconds
                        4W              ; expire, seconds
                        1D )            ; minimum, seconds
;
                TXT     "Linux.Bogus, your DNS consultants"
                NS      ns              ; Inet Address of name server
                NS      ns.friend.bogus.
                MX      10 mail         ; Primary Mail Exchanger
                MX      20 mail.friend.bogus. ; Secondary Mail Exchanger

localhost       A       127.0.0.1

gw              A       192.168.196.1
                TXT     "The router"

ns              A       192.168.196.2
                MX      10 mail
                MX      20 mail.friend.bogus.
www             CNAME   ns

donald          A       192.168.196.3
                MX      10 mail
                MX      20 mail.friend.bogus.
                TXT     "DEK"

mail            A       192.168.196.4
                MX      10 mail
                MX      20 mail.friend.bogus.

ftp             A       192.168.196.5
                MX      10 mail
                MX      20 mail.friend.bogus.
```

A CNAME (Canonical NAME - kanonikus NÉV) egy módszer több név megadására egy gép számára. Így a www egy álnév az ns számára. A CNAME bejegyzés használata egy kicsit kétértelmű. A legbiztosabb azt a szabályt követni, hogy egy MX, CNAME vagy SOA bejegyzés *soha* nem hivatkozhat egy CNAME bejegyzésre, csak egy "A" bejegyzéssel rendelkező valamire hivatkozhatnak, tehát megengedhetetlen a

```
foobar          CNAME   www                     ; NEM!
```

de helyes a

```
foobar          CNAME   ns                      ; IGEN!
```

Töltsük be az új adatbázist az `rndc reload` futtatásával, amely a named állományainak újbóli beolvasását eredményezi.

```
$ dig linux.bogus axfr

; <<>> DiG 9.1.3 <<>> linux.bogus axfr
;; global options:  printcmd
linux.bogus.            259200  IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. 199802151 28800 72
linux.bogus.            259200  IN      NS      ns.linux.bogus.
linux.bogus.            259200  IN      MX      10 mail.linux.bogus.
linux.bogus.            259200  IN      MX      20 mail.friend.bogus.

donald.linux.bogus.     259200  IN      A       192.168.196.3
donald.linux.bogus.     259200  IN      MX      10 mail.linux.bogus.
donald.linux.bogus.     259200  IN      MX      20 mail.friend.bogus.
donald.linux.bogus.     259200  IN      TXT     "DEK"
ftp.linux.bogus.        259200  IN      A       192.168.196.5
ftp.linux.bogus.        259200  IN      MX      10 mail.linux.bogus.

ftp.linux.bogus.        259200  IN      MX      20 mail.friend.bogus.
gw.linux.bogus.         259200  IN      A       192.168.196.1
gw.linux.bogus.         259200  IN      TXT     "The router"
localhost.linux.bogus.  259200  IN      A       127.0.0.1
mail.linux.bogus.       259200  IN      A       192.168.196.4
mail.linux.bogus.       259200  IN      MX      10 mail.linux.bogus.
mail.linux.bogus.       259200  IN      MX      20 mail.friend.bogus.
ns.linux.bogus.         259200  IN      MX      10 mail.linux.bogus.
ns.linux.bogus.         259200  IN      MX      20 mail.friend.bogus.
ns.linux.bogus.         259200  IN      A       192.168.196.2
www.linux.bogus.        259200  IN      CNAME   ns.linux.bogus.
linux.bogus.            259200  IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. 199802151 28800 72
;; Query time: 41 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 03:12:31 2001
;; XFR size: 23 records
```

Ez jó. Amint látod, egy kicsit úgy néz ki, mint a zónaállomány maga. Ellenőrizzük, mit mond egyedül a `www`-re:

```
$ dig www.linux.bogus

; <<>> DiG 9.1.3 <<>> www.linux.bogus
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16633
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 2, AUTHORITY: 1, ADDITIONAL: 0

;; QUESTION SECTION:
;www.linux.bogus.               IN      A

;; ANSWER SECTION:
www.linux.bogus.        259200  IN      CNAME   ns.linux.bogus.
ns.linux.bogus.         259200  IN      A       192.168.196.2

;; AUTHORITY SECTION:
linux.bogus.            259200  IN      NS      ns.linux.bogus.

;; Query time: 5 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 03:14:14 2001
;; MSG SIZE  rcvd: 80
```

Más szóval a `www.linux.bogus` valódi neve `ns.linux.bogus`, és további információt is ad neked amivel rendelkezik az ns-ről, elegendőt a hozzá való csatlakozáshoz, ha egy program lennél.

Most vagyunk félúton

5.3 A fordított zóna

Most már a programok át tudják alakítani a linux.bogus-ban a neveket címekké, amelyekhez csatlakozni tudnak. De szükség van egy fordított zónára is, olyanra, amely lehetővé teszi a DNS számára a címek átalakítását nevekké. Ezt a nevet rengeteg különböző típusú szerver (FTP, IRC, WWW és mások) használja annak eldöntésére, hogy akar-e veled kommunikálni vagy nem, és ha igen, talán még arra is, hogy milyen prioritást kapjál. Az Internet összes szolgáltatásának teljes eléréséhez a fordított zóna szükséges.

Rakd be ezt a `named.conf` állományba:

```
zone "196.168.192.in-addr.arpa" {
        type master;
        notify no;
        file "pz/192.168.196";
};
```

Ez pontosan ugyanaz, mint a `0.0.127.in-arpa`-val, és a tartalmuk is hasonló:

```
$TTL 3D
@       IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. (
                        199802151 ; Serial, todays date + todays serial
                        8H      ; Refresh
                        2H      ; Retry
                        4W      ; Expire
                        1D)     ; Minimum TTL
```

```
                NS      ns.linux.bogus.

1               PTR     gw.linux.bogus.
2               PTR     ns.linux.bogus.
3               PTR     donald.linux.bogus.
4               PTR     mail.linux.bogus.
5               PTR     ftp.linux.bogus.
```

Most újra töltsd be a named-et (`rndc reload`), és vizsgáld meg a munkádat a `dig`-el újra:

```
$ dig -x 192.168.196.4

;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58451
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;4.196.168.192.in-addr.arpa.    IN      PTR

;; ANSWER SECTION:
4.196.168.192.in-addr.arpa. 259200 IN   PTR     mail.linux.bogus.

;; AUTHORITY SECTION:
196.168.192.in-addr.arpa. 259200 IN     NS      ns.linux.bogus.

;; ADDITIONAL SECTION:
ns.linux.bogus.         259200  IN      A       192.168.196.2

;; Query time: 4 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 03:16:05 2001
;; MSG SIZE  rcvd: 107
```

---

tehát jónak néz ki, szedjük ki az egészet, hogy azt is megvizsgáljuk:

```
$ dig 196.168.192.in-addr.arpa. AXFR

; <<>> DiG 9.1.3 <<>> 196.168.192.in-addr.arpa. AXFR
;; global options:  printcmd
196.168.192.in-addr.arpa. 259200 IN     SOA     ns.linux.bogus. \

        hostmaster.linux.bogus. 199802151 28800 7200 2419200 86400
196.168.192.in-addr.arpa. 259200 IN     NS      ns.linux.bogus.
1.196.168.192.in-addr.arpa. 259200 IN   PTR     gw.linux.bogus.
2.196.168.192.in-addr.arpa. 259200 IN   PTR     ns.linux.bogus.
3.196.168.192.in-addr.arpa. 259200 IN   PTR     donald.linux.bogus.
4.196.168.192.in-addr.arpa. 259200 IN   PTR     mail.linux.bogus.
5.196.168.192.in-addr.arpa. 259200 IN   PTR     ftp.linux.bogus.
196.168.192.in-addr.arpa. 259200 IN     SOA     ns.linux.bogus. \
```

```
        hostmaster.linux.bogus. 199802151 28800 7200 2419200 86400
;; Query time: 6 msec
;; SERVER: 127.0.0.1#53(127.0.0.1)
;; WHEN: Sun Dec 23 03:16:58 2001
;; XFR size: 9 records
```

Jól néz ki! Ha kimeneted nem ilyen, akkor keresd a hibaüzeneteket a syslog-ban, az első fejezetben, 3.1 (A named indítása) fejezetben elmagyaráztam, hogyan tedd ezt.

5.4 Intő szavak

Van pár dolog, amit itt közre kell adnom. A fenti példában használt IP számok a "magánhálózatok" egyik blokkjából lettek véve, azaz nyilvános használatuk az Interneten nem megengedett. Így hát biztonságos a használatuk egy HOGYAN egy példájában. A másik dolog a `notify no;` sor. Ez megmondja a named-nek, hogy ne értesítse a másodlagos (slave) szerverét, amikor az egyik zónaállománya frissült. A 8-as és későbbi BIND-ben a named értesítheti a zónaállományban az NS bekezdésben felsorolt többi szervert, amikor a zóna frissült. Ez ügyes dolog rendes működéskor. De kísérletezések esetén ennek a lehetőségnek kikapcsolva kell lennie - nem akarjuk, hogy a kísérlet megkavarja az Internetet, ugye?

És persze, ez a tartomány erősen hamis, és éppígy a címek benne. Egy valódi tartomány valós példájáért nézd meg a következő főfejezetet.

5.5 Miért nem működnek a fordított lekérdezések?

Van egy pár normális körülmények között névlekérdezésekkel elkerülhető "csapda", amellyel gyakran találkozni fordított zónák beállításánál. Mielőtt folytatod, szükséged lesz a fordított lekérdezések működésére a saját névszervereden. Ha ez nincs így, menj vissza, és javítsd ki mielőtt folytatod.

A fordított lekérdezések két hibájáról fogok szólni, ahogy azok a hálózaton kívülről látszódnak:

5.5.1 A fordított zóna nincs delegálva

Ha egy hálózati szolgáltatótól egy hálózati címtartományt és egy tartománynevet kérsz, a tartománynév rendes esetben delegálva van, mint egy magától értetődő dolog. A delegálás az az összeragasztó NS bejegyzés, amely segít eljutnod az egyik névszervertől a másikig, ahogy ez a száraz elméleti fejezetben el lett magyarázva. Elolvastad, ugye? Ha a fordított zónád nem működik, menj vissza, és olvasd el. Most.

A fordított zónának szintén delegálva kell lennie. Ha a `192.168.196`-os hálózatot kapod a `linux.bogus` tartománnyal a szolgáltatódtól, be kell rakniuk az NS bejegyzést a fordított zónád számára éppúgy, mint a továbbító zónád számára. Ha követed a láncolatot az `in-addr.arpa`-tól felfelé a hálózatodig, valószínűleg szakadást találsz majd a láncban, a leginkább valószínű, hogy a szolgáltatódnál. Miután megtaláltad a szakadást a láncolatban, vedd fel a kapcsolatot a szolgáltatóddal, és kérd meg őket a hiba kijavítására.

5.5.2 Egy osztályon kívüli alhálózatod van

Ez egy kissé bonyolultabb téma, de az osztályon kívüli alhálózatok nagyon elterjedtek manapság, és valószínűleg egy ilyened van, ha egy kis cég vagy.

Az osztályon kívüli alhálózatok azok, amik az Internetet manapság éltetik. Néhány évvel ezelőtt sok volt a hűhó az IP címek fogyatkozása miatt. A bölcs emberek az IETF-nél (Internet Engineering Task Force, ők tartják működésben az Internetet) összedugták a fejüket, és megoldották a problémát. Bizonyos áron. Az

ár ott mutatkozik, hogy egy "C" alhálózatnál kisebbet kapsz, és bizonyos dolgok nem működhetnek. Kérlek nézd át az *Ask Mr. DNS* <http://www.acmebw.com/askmrdns/00007.htm> (Kérdezd DNS urat) cikket egy jó magyarázatért, és hogy hogyan kezeld ezt.

Elolvastad? Nem fogom elmagyarázni, szóval kérlek olvasd el.

A probléma első része az, hogy az ISP-dnek értenie kell a Mr. DNS által leírt technikát. Nem minden kis ISP-nek van hozzáértő dolgozója ehhez. Ha így van, lehet, hogy el kell nekik magyaráznod, és kitartónak kell lenned. De először légy biztos benne, hogy te érted ;-). Ezután be fognak állítani egy rendes fordított zónát a szerverükön, melynek helyességét megvizsgálhatod a dig-el.

A probléma második és egyben utolsó része az, hogy meg kell értened a technikát. Ha nem vagy biztos benne, menj vissza, és olvass róla ismét. Ezután beállíthatod a saját osztályon kívüli fordított zónádat úgy, ahogy azt az Ask Mr. DNS leírja.

Lapul egy másik csapda is itt. A (nagyon) régi feloldók *nem* lesznek képesek követni a CNAME trükköt a feloldási láncban, és nem lesznek képesek fordítva feloldani a gépedet. Ez egy szolgáltatás esetében helytelen hozzáférési osztály hozzárendelését, a hozzáférés megtagadását vagy ezekhez hasonlót eredményezhet. Ha egy ilyen szolgáltatásba ütközöl, az egyetlen megoldás (amiről én tudok) az ISP-d számára az, hogy belerakja a PTR bejegyzésedet közvetlenül az ő trükkös osztályon kívüli zónaállományukba a trükkös CNAME bejegyzés helyett.

Bizonyos ISP-k más módokat fognak ajánlani ennek kezelésére, úgymint Web-alapú űrlapokat a fordított hozzárendelés megadásához, vagy más automágikus rendszereket.

5.6 Másodlagos (slave) szerverek

Amint helyesen beállítottad a zónáidat az elsődleges (master) szerveren, fel kell állítanod legalább egy másodlagos (slave) szervert. A másodlagos szerverek a robusztusság miatt szükségesek. Ha az elsődleges leáll, az emberek ott kint a hálón még mindig képesek lesznek információt kapni a tartományodról a szolgától. A másodlagosnak olyan messze kell lennie tőled, amennyire csak lehetséges. Az alábbi dolgok közül az elsődlegesnek és a másodlagosnak minél kevesebben kellene osztozniuk: áramellátás, LAN, ISP, város és ország. Ha ez mind más az elsődleges és a másodlagos esetében, egy tényleg jó másodlagos szervert találtál.

A másodlagos szerver egyszerűen egy olyan névszerver, amely a zónaállományokat lemásolja az elsődlegesről. Ekképp állíthatod be:

```
zone "linux.bogus" {
        type slave;
        file "sz/linux.bogus";
        masters { 192.168.196.2; };
};
```

Az adatok másolására a zónaátvitel nevű mechanizmust használják. A zónaátvitelt az SOA bejegyzésed irányítja:

```
@       IN      SOA     ns.linux.bogus. hostmaster.linux.bogus. (
                        199802151       ; serial, todays date + todays serial #
                        8H              ; refresh, seconds
                        2H              ; retry, seconds
                        4W              ; expire, seconds
                        1D )            ; minimum, seconds
```

Egy zóna csak akkor kerül átvitelre, ha a sorozatszáma (serial) az elsődleges szerveren nagyobb, mint a másodlagoson. Frissítési intervallumonként (refresh) a másodlagos szerver le fogja ellenőrizni, hogy az elsődleges frissült-e. Ha az ellenőrzés nem hoz eredményt (mert az elsődleges nem elérhető), újrapróbálja a megadott intervallumonként (retry). Ha az ellenőrzések a lejárati időszak (expire) alatt sem hoznak eredményt, a másodlagos szerver el fogja távolítani a zónát az állományrendszeréből, és nem lesz többé szerver számára.

6 Alapvető biztonsági beállítások

Jamie Norrish

A konfigurációs opciók beállítása a problémák valószínűségének csökkentése érdekében.

Van néhány egyszerű lépés amelyet megtehetsz, ezek biztonságosabbá teszik a szerveredet, és esetlegesen csökkentik a terhelését is. Az itt bemutatott anyag nem több, mint egy kiindulási pont; ha érdekelt vagy a biztonságban (és így kellene lennie), kérlek tanulmányozz át más forrásmunkákat is a hálózaton (lásd az 11 (utolsó fejezetet)).

A következő beállítási direktívák fordulnak elő a `named.conf` állományban. Az options részben található direktívák az összes zónára vonatkoznak. Ha a `zone` bejegyzésben fordul elő, csak arra a zónára vonatkozik. Egy `zone` bejegyzés felülírja az `options` bejegyzést.

6.1 A zónaátvitelek korlátozása

Annak érdekében, hogy a másodlagos szervere(i)d képes legyen válaszolni a tartományodra vonatkozó lekérdezésekre, képeseknek kell lenniük áthozni a zónainformációt az elsődleges szerveredről. Nagyon sokan szeretnének szintén így cselekedni. Ezért korlátozd a zónaátvitelt az `allow-transfer` opció használatával, feltételezve, hogy 192.168.1.4 az ns.friend.bogus címe, és hozzáadva saját magadat hibakeresési célból:

```
zone "linux.bogus" {
      allow-transfer { 192.168.1.4; localhost; };
};
```

A zónaátvitelek korlátozásával biztosítod, hogy az egyetlen elérhető információ az, amit az emberek közvetlenül kérdeznek - senki sem kérdezheti le csak úgy beállításod összes részletét.

6.2 Védekezés az ”átejtés” ellen

Legelőször kapcsolj ki minden lekérdezést, ami nem az általad birtokolt tartományokra irányul, kivéve a belső/helyi gépeidről indulókat. Ez nem csak a DNS szervered rosszindulatú kihasználását előzi meg, de csökkenti szervered felesleges használatát is.

```
options {
      allow-query { 192.168.196.0/24; localhost; };
};

zone "linux.bogus" {
      allow-query { any; };
};
```

```
zone "196.168.192.in-addr.arpa" {
      allow-query { any; };
};
```

Továbbá kapcsold ki a rekurzív lekérdezéseket, kivéve a belső/helyi gépektől. Ez csökkenti a gyorsítótár-mérgezéses támadások esélyét (amikor hamis adatokkal tömik a szerveredet).

```
options {
        allow-recursion { 192.168.196.0/24; localhost; };
};
```

6.3 A named futtatása nem-root-ként

Egy jó ötlet a named-et a root-tól különböző felhasználóként futtatni, így ha feltörik a cracker által szerzett jogok a lehető legkorlátozottabbak. Először létre kell hoznod egy felhasználót ami alatt a named fusson, majd módosítani bármely általad használt, a named-et indító init szkriptet. Az új felhasználónevet és csoportot a named-nek az -u és -g kapcsolók segítségével add meg.

Például: Debian GNU/Linux 2.2-ben módosítanod kell a `/etc/init.d/bind` szkriptet, hogy tartalmazza a következő sort (ahol a `named` felhasználó már létre lett hozva):

```
start-stop-daemon --start --quiet --exec /usr/sbin/named -- -u named
```

Ugyanez megtehető a Red Hat-al és más disztribúciókkal is.

Dave Lugo leírt egy biztonságos kettős chroot beállítást, amely a

<http://www.etherboy.com/dns/chrootdns.html> honlapon található, ez még biztonságosabbá teheti a gépet, amelyen a named-et futtatod.

7 Egy valódi tartomány-példa

Ahol bemutatunk néhány *igazi* zónaállományt

A felhasználók javasolták, hogy illesszem be egy működő tartomány valós példáját is, mint szemléltető példát.

E példát David Bullock engedélyével a LAND-5-től használom. Ezek az állományok 1996. szeptember 24.-én voltak aktuálisak, és ezután szerkesztettem át őket, hogy megfeleljenek a 8-as BIND megkötéseinek és kiterjesztés-használatának. Szóval az amit látsz, különbözik egy kicsit attól, amit a LAND-5 névszerverek lekérdezésekor találsz.

7.1 /etc/named.conf (vagy /var/named/named.conf)

Itt találhatók az elsődleges szerver zónafejezetei a két szükséges fordított zóna számára: a 127.0.0 hálózat éppúgy, mint a LAND-5 `206.6.177`-es alhálózata, és az elsődleges sor a land-5 `land5.com` továbbító zónája számára. Figyeld meg, hogy az állományok `pz` nevű könyvtárba való pakolása helyett, ahogy én ezt ebben a HOGYANban teszem, ő a `zone` nevű könyvtárba rakja őket.

```
// Boot file for LAND-5 name server

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

controls {
        inet 127.0.0.1 allow { localhost; } keys { rndc_key; };
};

key "rndc_key" {
        algorithm hmac-md5;
        secret "c3Ryb25nIGVub3VnaCBmb3IgYSBtYW4gYnV0IG1hZGUgZm9yIGEgd29tYW4K";
};

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

zone "0.0.127.in-addr.arpa" {
        type master;
        file "zone/127.0.0";
};

zone "land-5.com" {
        type master;
        file "zone/land-5.com";
};

zone "177.6.206.in-addr.arpa" {
        type master;
        file "zone/206.6.177";
};
```

Ha ezt berakod a named.conf állományodba kísérletezés céljából, **KÉRLEK** rakd be a "`notify no;`"-t a két `land-5` zóna zone fejezetébe, hogy elkerüljük az ütközéseket.

7.2 /var/named/root.hints

Tartsd szem előtt, hogy ez az állomány dinamikus, és az itt közzétett változat régi. Jobban teszed ha egy újabbat használsz, amint azt már korábban elmagyaráztam.

```
; <<>> DiG 8.1 <<>> @A.ROOT-SERVERS.NET.
; (1 server found)
;; res options: init recurs defnam dnsrch
;; got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 10
;; flags: qr aa rd; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 13
;; QUERY SECTION:
```

```
;;      ., type = NS, class = IN

;; ANSWER SECTION:
.                       6D IN NS        G.ROOT-SERVERS.NET.
.                       6D IN NS        J.ROOT-SERVERS.NET.
.                       6D IN NS        K.ROOT-SERVERS.NET.
.                       6D IN NS        L.ROOT-SERVERS.NET.
.                       6D IN NS        M.ROOT-SERVERS.NET.
.                       6D IN NS        A.ROOT-SERVERS.NET.
.                       6D IN NS        H.ROOT-SERVERS.NET.
.                       6D IN NS        B.ROOT-SERVERS.NET.
.                       6D IN NS        C.ROOT-SERVERS.NET.
.                       6D IN NS        D.ROOT-SERVERS.NET.
.                       6D IN NS        E.ROOT-SERVERS.NET.
.                       6D IN NS        I.ROOT-SERVERS.NET.
.                       6D IN NS        F.ROOT-SERVERS.NET.

;; ADDITIONAL SECTION:
G.ROOT-SERVERS.NET.     5w6d16h IN A    192.112.36.4
J.ROOT-SERVERS.NET.     5w6d16h IN A    198.41.0.10
K.ROOT-SERVERS.NET.     5w6d16h IN A    193.0.14.129
L.ROOT-SERVERS.NET.     5w6d16h IN A    198.32.64.12
M.ROOT-SERVERS.NET.     5w6d16h IN A    202.12.27.33
A.ROOT-SERVERS.NET.     5w6d16h IN A    198.41.0.4
H.ROOT-SERVERS.NET.     5w6d16h IN A    128.63.2.53
B.ROOT-SERVERS.NET.     5w6d16h IN A    128.9.0.107
C.ROOT-SERVERS.NET.     5w6d16h IN A    192.33.4.12
D.ROOT-SERVERS.NET.     5w6d16h IN A    128.8.10.90
E.ROOT-SERVERS.NET.     5w6d16h IN A    192.203.230.10
I.ROOT-SERVERS.NET.     5w6d16h IN A    192.36.148.17
F.ROOT-SERVERS.NET.     5w6d16h IN A    192.5.5.241

;; Total query time: 215 msec
;; FROM: roke.uio.no to SERVER: A.ROOT-SERVERS.NET.  198.41.0.4
;; WHEN: Sun Feb 15 01:22:51 1998
;; MSG SIZE  sent: 17  rcvd: 436
```

7.3 /var/named/zone/127.0.0

Csak az alapok, a kötelező SOA bejegyzés, és a bejegyzés, mely a 127.0.0.1-et a `localhost`-hoz rendeli. Mindkettő szükséges. Semmi másnak nem kell lennie ebben az állományban. Valószínűleg soha nem lesz frissítve, hacsak a névszervered vagy a rendszergazda címe nem változik meg.

```
$TTL 3D
@               IN      SOA     land-5.com. root.land-5.com. (
                                199609203       ; Serial
                                28800   ; Refresh
                                7200    ; Retry
                                604800  ; Expire
                                86400)  ; Minimum TTL
```

```
                        NS      land-5.com.

1                       PTR     localhost.
```

Ha egy véletlenszerűen kiválasztott BIND telepítésre ránézel, azt fogod találni, hogy a `$TTL` sor hiányzik. Ezt azelőtt nem használták, és csak a 8.2-es BIND kezdett el figyelmeztetni a hiányára. A 9-es BIND-hez *szükséges* a `$TTL`.

7.4 /var/named/zone/land-5.com

Itt látjuk a kötelező SOA bejegyzést, a szükséges NS bejegyzéseket. Láthatjuk, hogy van egy másodlagos névszervere az `ns2.psi.net`-en. Ez az ahogy lennie kell, mindig legyen egy telephelyen kívüli másodlagos szervered tartalékként. Láthatjuk azt is, hogy van neki egy `land-5` nevű elsődleges gépe is, amely gondoskodik sok különböző Internet szolgáltatásról, és azt, hogy ezt CNAME-ekkel csinálta (egy másik lehetőség az "A" bejegyzések használata).

Amint azt a SOA bejegyzésből láthatod, a zónaállomány eredete a `land-5.com`, a kapcsolattartó személy a `root@land-5.com`. A `hostmaster` egy másik gyakran használt cím a kapcsolattartó személy számára. A sorozatszám a szokásos éééévhnn formátumban van, a mai sorozatszám hozzáadásával; ez valószínűleg a zónaállomány hatodik változata 1996. szeptember 20.-án. Jegyezd meg, hogy a sorozatszámnak monoton növekvőnek *kell* lennie, itt csak *egy* számjegy jelzi a mai sorozatszámot, így 9 szerkesztés után várnia kell holnapig, mielőtt újra szerkesztheti az állományt. Szokd meg a két számjegy használatát.

```
$TTL 3D
@       IN      SOA     land-5.com. root.land-5.com. (
                        199609206       ; serial, todays date + todays serial #
                        8H              ; refresh, seconds
                        2H              ; retry, seconds
                        4W              ; expire, seconds
                        1D )            ; minimum, seconds
                NS      land-5.com.
                NS      ns2.psi.net.
                MX      10 land-5.com.  ; Primary Mail Exchanger
                TXT     "LAND-5 Corporation"

localhost       A       127.0.0.1
router          A       206.6.177.1
land-5.com.     A       206.6.177.2
ns              A       206.6.177.3
www             A       207.159.141.192
ftp             CNAME   land-5.com.
mail            CNAME   land-5.com.
news            CNAME   land-5.com.
funn            A       206.6.177.2

;
;       Workstations
;
ws-177200       A       206.6.177.200
                MX      10 land-5.com.   ; Primary Mail Host
ws-177201       A       206.6.177.201
```

```
                MX      10 land-5.com.   ; Primary Mail Host
ws-177202       A       206.6.177.202
                MX      10 land-5.com.   ; Primary Mail Host
ws-177203       A       206.6.177.203
                MX      10 land-5.com.   ; Primary Mail Host
ws-177204       A       206.6.177.204
                MX      10 land-5.com.   ; Primary Mail Host
ws-177205       A       206.6.177.205
                MX      10 land-5.com.   ; Primary Mail Host
; {Many repetitive definitions deleted - SNIP}
ws-177250       A       206.6.177.250
                MX      10 land-5.com.   ; Primary Mail Host
ws-177251       A       206.6.177.251
                MX      10 land-5.com.   ; Primary Mail Host
ws-177252       A       206.6.177.252
                MX      10 land-5.com.   ; Primary Mail Host
ws-177253       A       206.6.177.253
                MX      10 land-5.com.   ; Primary Mail Host
ws-177254       A       206.6.177.254
                MX      10 land-5.com.   ; Primary Mail Host
```

Ha megvizsgálod a land-5 névszerverét, azt találod, hogy a gépnevek `ws_`*szám* alakúak. A 4-es BIND-től kezdődően a named elkezdte szigorítani a megkötéseket, hogy milyen karakterek szerepelhetnek a gépnevekben. Így ez a 8-as BIND-el egyáltalán nem működik, és én kicseréltem a "-"-re (kötőjel) a "_"-t (aláhúzás) ebben a HOGYANban. De ahogy azt korábban említettem, a 9-es BIND nem erőlteti már ezt a megkötést.

A másik figyelemre méltó dolog az, hogy a munkaállomásoknak nincs egyedi nevük, hanem csak egy előtagot követ az IP utolsó két része. Egy ilyen megszokás használata jelentősen leegyszerűsítheti a karbantartást, de egy kicsit személytelennek tűnhet, és lényegében bosszúság forrása lehet az ügyfeleid számára.

Látjuk azt is, hogy a `funn.land-5.com` egy álnév a `land-5.com` számára, de egy "A", és nem egy CNAME bejegyzés használatával.

7.5 /var/named/zone/206.6.177

Megjegyzéseket ezen állományra lentebb teszek.

```
$TTL 3D
@               IN      SOA     land-5.com. root.land-5.com. (
                                199609206       ; Serial
                                28800   ; Refresh
                                7200    ; Retry
                                604800  ; Expire
                                86400)  ; Minimum TTL
                        NS      land-5.com.
                        NS      ns2.psi.net.

;       Servers
;
1       PTR     router.land-5.com.
```

```
2       PTR     land-5.com.
2       PTR     funn.land-5.com.
;
;       Workstations
;
200     PTR     ws-177200.land-5.com.
201     PTR     ws-177201.land-5.com.
202     PTR     ws-177202.land-5.com.
203     PTR     ws-177203.land-5.com.
204     PTR     ws-177204.land-5.com.
205     PTR     ws-177205.land-5.com.
; {Many repetitive definitions deleted - SNIP}
250     PTR     ws-177250.land-5.com.
251     PTR     ws-177251.land-5.com.
252     PTR     ws-177252.land-5.com.
253     PTR     ws-177253.land-5.com.
254     PTR     ws-177254.land-5.com.
```

A fordított zóna a beállítás azon része, mely a legtöbb fejtörést okozhatja. Ezt arra használjuk, hogy megtaláljuk a gépnevet, ha megvan a gép címe. Példa: te egy FTP szerver vagy és kapcsolatokat fogadsz el FTP kliensektől. Mivel te egy norvég FTP szerver vagy, több kapcsolatot szeretnél fogadni norvégiai és más skandináv államokbeli kliensektől, és kevesebbet a világ többi részéről. Ha kapcsolat érkezik egy klienstől, a C függvénykönyvtár képes neked megmondani a csatlakozó gép IP címét, mert a kliens IP számát tartalmazza az összes csomag, amely átjött a hálózaton. Most meghívhatsz egy gethostbyaddr nevű függvényt, mely kikeresi az adott IP számú kliens nevét. A gethostbyaddr meg fogja kérdezni a DNS szervert, amely ezután keresztülmegy a DNS-en a gépet keresve. Tételezzük fel, hogy a klienskapcsolat a ws-177200.land-5.com-tól jön. Az IP szám, amit a C könyvtár átad az FTP szervernek, a 206.6.177.200. A gép nevének kitalálásához meg kell találnunk a `200.177.6.206.in-addr-arpa`-t. A DNS szerver először meg fogja találni az `arpa.` szervereket, majd megtalálja az `in-addr.arpa` szervereket, követve a fordított sorrendet a 206-on, majd a 6-on keresztül, végül legutoljára megtalálva a LAND-5-nél a szervert a `177.6.206.in-addr-arpa` zóna számára. Amelytől végül megkapja a választ, hogy a `200.177.6.206.in-addr.arpa` számára a "`PTR ws-177200.land-5.com`" bejegyzésünk van, ami azt jelenti, hogy a `/ 206.6.177.200`-hoz tartozó név a `ws-177200.land.com`.

Az FTP szerver előnyben részesíti a skandináv országok, azaz a `*.no`, `*.se`, `*.dk` felől érkező kapcsolatokat, a `ws-177200.land-5.com` egyértelműen nem tartozik közéjük, és a szerver a kapcsolatot egy alacsonyabb sávszélességgel és kevesebb klienskapcsolati lehetőséggel rendelkező kapcsolati osztályba sorolja. Ha *nem* lenne fordított megfeleltetése a `206.2.177.200`-nak az `in-addr.arpa` zóna által, a szerver képtelen lenne megtalálni a nevet, és a `206.2.177.200`-nek a `*.no`, `*.se` és `*.dk`-val való összehasonlítása alapján kell döntenie, melyek közül egyik sem fog egyezni, sőt még meg is tagadhatja a kapcsolatot a besorolás hiánya miatt.

Páran azt fogják mondani neked, hogy a fordított lekérdezések hozzárendelése csak szerverek esetén fontos, vagy egyáltalán nem fontos. Nem így van: sok ftp, news, IRC, sőt még néhány http (WWW) szerver *nem* fognak kapcsolatot fogadni olyan gépektől, melyek nevét képtelenek megtalálni. Így hát a fordított hozzárendelés valójában *kötelező*.

8 Karbantartás

Üzemben tartás.

Van egy karbantartási feladat, melyet meg kell tenned a named-eken - a futtatáson kívül. Ez pedig a `root.hints` állomány naprakészen tartása. A legegyszerűbb mód a `dig` használata. Először futtasd a `dig`-et argumentumok nélkül, akkor megkapod a `root.hints`-et a saját szervered alapján. Ezután kérdezd le a felsorolt főszerverek egyikét a `dig @rootserver` paranccsal. Észre fogod venni, hogy a kimenet szörnyen hasonló a root.hints állományhoz. Mentsd el egy állományba (`dig @e.root-servers.net . ns > root.hints.new`), és cseréld le a régi `root.hints` állományt vele.

Ne felejtsd el újra betölteni a named-et a gyorsítótár-állomány cseréje után.

Al Longyear elküldte nekem ezt a szkriptet, mely automatikusan futtatható a `root.hints` frissítése érdekében. Telepíts egy crontab bejegyzést, hogy havonta egyszer lefusson, és el is felejtheted. A szkript feltételezi, hogy a levelezésed működik, és hogy a "hostmaster" cím meg van adva. Meg kell hackelned, hogy illeszkedjen a beállításaidhoz.

```
#!/bin/sh
#
# Update the nameserver cache information file once per month.
# This is run automatically by a cron entry.
#
# Original by Al Longyear
# Updated for BIND 8 by Nicolai Langfeldt
# Miscelanious error-conditions reported by David A. Ranch
# Ping test suggested by Martin Foster
# named up-test suggested by Erik Bryer.
#
(
 echo "To: hostmaster <hostmaster>"
 echo "From: system <root>"

 # Is named up? Check the status of named.
 case `rndc status 2>&1` in
    *refused*)
        echo "named is DOWN. root.hints was NOT updated"
        echo
        exit 0
        ;;
 esac

 PATH=/sbin:/usr/sbin:/bin:/usr/bin:
 export PATH
 # NOTE: /var/named must be writable only by trusted users or this script
 # will cause root compromise/denial of service opportunities.
 cd /var/named 2>/dev/null || {
    echo "Subject: Cannot cd to /var/named, error $?"
    echo
    echo "The subject says it all"
    exit 1
 }

 # Are we online?  Ping a server at your ISP
 case `ping -qnc 1 some.machine.net 2>&1` in
   *'100% packet loss'*)
```

```
        echo "Subject: root.hints NOT updated.  The network is DOWN."
        echo
        echo "The subject says it all"
        exit 1
        ;;
 esac

 dig @e.root-servers.net . ns >root.hints.new 2> errors

 case `cat root.hints.new` in
   *NOERROR*)
        # It worked
        :;;
   *)
        echo "Subject: The root.hints file update has FAILED."
        echo
        echo "The root.hints update has failed"
        echo "This is the dig output reported:"
        echo
        cat root.hints.new errors
        exit 1
        ;;
 esac

 echo "Subject: The root.hints file has been updated"

 echo
 echo "The root.hints file has been updated to contain the following
information:"
 echo
 cat root.hints.new

 chown root.root root.hints.new
 chmod 444 root.hints.new
 rm -f root.hints.old errors
 mv root.hints root.hints.old
 mv root.hints.new root.hints
 rndc restart
 echo
 echo "The nameserver has been restarted to ensure that the update is complete."
 echo "The previous root.hints file is now called
/var/named/root.hints.old."
) 2>&1 | /usr/lib/sendmail -t
exit 0
```

Néhányan közületek felfigyelhettek rá, hogy a `root.hints` állomány elérhető ftp-vel az Internic-ről is. Kérlek, ne használd az ftp-t a `root.hints` frissítéséhez, a fentebb említett módszer sokkal barátságosabb a hálózat és az Internic számára.

9 Átállás 9-es BIND-re

A 9-es BIND terjesztés - és az előre elkészített változatok szintén - tartalmaz egy `migration` nevű dokumentumot, amely megjegyzéseket tartalmaz azt illetően, hogy hogyan álljunk át 8-as BIND-ről 9-es BIND-re. A dokumentum nagyon lényegre törő. Ha bináris csomagokat telepítettél, feltehetően valahol a `/usr/share/doc/bind*`-ban vagy a `/usr/doc/bind*`-ban van tárolva.

Ha 4-es BIND-et futtatsz, a `migration-4to9` dokumentumot ugyanazon a helyen találhatod.

10 Kérdések és válaszok

Kérlek olvasd át ezt a fejezetet, mielőtt írsz nekem.

1. A named-em egy named.boot állományt akar

   Rossz HOGYANt olvasol. Kérlek nézd meg ezen HOGYAN régebbi változatát, amely a 4-es BIND-ről szól, a `<http://langfeldt.net/DNS-HOWTO/>` címen.

2. Hogy használhatom egy tűzfal mögül?

   Segítség: `forward only;`. Szükséged lehet még a

```
query-source port 53;
```

   sorra a `named.conf` állomány "options" részén belül, ahogy az a példának bemutatott 3 (A feloldó, gyorsítótáras névszerver) fejezetben javasoltam.

3. Mit tegyek, hogy a DNS körbeforogjon egy szolgáltatás elérhető címein, mondjuk a www.busy.site-on, hogy terheléselosztó vagy valami hasonló hatást érjek el?

   Csinálj több **A** bejegyzést a `www.busy.site` számára, és 4.9.3-as vagy későbbi BIND-et használj. Ekkor a BIND round-robin rendszer alapján fogja szolgáltatni a válaszokat. Ez *nem* fog működni a BIND korábbi változataival.

4. DNS-t akarok beállítani egy (zárt) belső hálózaton. Mit csináljak?

   Kihagyod a `root.hints` állományt, és csak a zónaállományokat készíted el. Ez azt is jelenti, hogy nem kell állandóan útbaigazító állományokat letöltened.

5. Hogyan kell beállítani egy másodlagos (slave) névszervert?

   Ha az elsődleges szerver címe 127.0.0.1, egy ehhez hasonló sort szúrsz be a másodlagos szervered named.conf állományába:

```
zone "linux.bogus" {
      type slave;
      file "sz/linux.bogus";
      masters { 127.0.0.1; };
};
```

   Több különböző elsődleges szervert is felsorolhatsz a masters listán belül, ";"-vel (pontosvessző) elválasztva, melyekről a zóna lemásolható.

6. Futtatni akarom a BIND-et, amikor nem vagyok kapcsolódva a hálózathoz.

   Négy lehetőség van:

- A 8/9-es BIND-re vonatkozóan, Adam L.Rice ezt a levelet küldte nekem arról, hogyan futtassuk fájdalommentesen a DNS-t egy betárcsázós gépen:

  A BIND újabb változatainál felfedeztem, hogy ez a kavarás az állományokkal többé nem szükséges. Van egy "forward" (továbbítás) direktíva a "forwarders" (továbbítók) direktíva mellett, amely a használatukat ellenőrzi. Az alap beállítás a "forward first" (először továbbítsd), amely legelőször megkérdezi a továbbítók mindegyikét, és ezután próbálja a rendes megközelítést, azaz a munka saját kezű elvégzését, ha ez nem sikerül. Ezzel a gethostbyname() normál viselkedésése szokatlanul hosszú időt vesz igénybe, amikor a kapcsolat nincs meg. De ha a "forward only" (csak továbbítsd) van beállítva, akkor a BIND feladja ha nem kap választ a továbbítóktól, és a gethostbyname() azonnal visszatér. Ennélfogva nincs szükség bűvészmutatványokra az /etc könyvtárban levő állományokkal, és a szerver újraindítására.

  Az én esetemben, csak hozzáadtam a

  forward only;
  forwarders { 193.133.58.5; };

  sorokat a named.conf állományom options { } fejezetéhez. Nagyon szépen működik. Ennek egyetlen hátránya az, hogy degradálja a DNS szoftver egy hihetetlenül szofisztikált részét egy buta gyorsítótárrá. Bizonyos mértékben, én csak egy buta gyorsítótárat szeretnék futtatni a DNS helyett, de úgy tűnik, nincs egy ilyen fajta elérhető szoftver Linuxra.

- Ezt a levelet Ian Clark-tól <ic@deakin.edu.au> kaptam, amiben az ő módszerét magyarázza erre:

  Named-et futtatok itt a "Masquerading" gépemen. Van két root.hints állományom, az egyik neve root.hints.real, amely a valódi főszerver-neveket tartalmazza, a másiké root.hints.fake, amely ezt tartalmazza:

  ```
  ----
  ; root.hints.fake
  ; this file contains no information
  ----
  ```

  Ha kapcsolat nélküli üzemmódba megyek át, átmásolom a root.hints.fake állományt a root.hints-be, és újraindítom a named-et.

  Ha kapcsolódom, átmásolom a root.hints.real-t a root.hints-be, és újraindítom a named-et.

  Illetve ezt az ip-down és az ip-up teszi meg.
  Amikor először végzek egy lekérdezést kapcsolat nélkül egy olyan tartománynévre, amelyről a named nem tudja a részleteket, egy ilyen bejegyzést rak a messages-be:

  Jan 28 20:10:11 hazchem named[10147]: No root nameserver for class IN

  amivel együtt tudok élni.

  Ez biztosan működik számomra. Használhatom a névszervert a helyi gépek esetében, amikor a Net áll, a külső tartománynevekhez tartozó időtúllépési késleltetés nélkül, és mikor a Hálón vagyok, a külső tartománynevekre vonatkozó lekérdezések rendben működnek

  Peter Denison azonban úgy vélte, Ian nem ment el elég messzire. Ezt írja:

```
Kapcsolódva)    szolgáltatja az eltárolt (és helyi hálózati) bejegyzéseket azonnal
a nem gyorsítótárazott bejegyzések esetén, továbbítja az ISP névszerverem felé
Kapcsolat nélkül) kiszolgálja a helyi hálózati lekérdezéseket azonnal
más lekérdezések esetén **azonnal** hibát ad

A fő gyorsítótáras állomány cseréjének és a lekérdezések továbbításának
kombinációja nem működik.

Így hát, (a helyi Linux Felhasználók Csoportjával való némi konzultáció után)
két named-et állítottam be a következő módon:

named-online:   továbbít az ISP névszervere felé
                mester a helyi hálózati zóna számára
                mester a helyi hálózati fordított zóna számára (1.168.192.in-addr.arpa)
                mester a 0.0.127.in-addr.arpa számára
                a 60053-as porton figyel

named-offline:  nincs továbbítás
                "ál" fő gyorsítótáras állomány
                szolga a 3 helyi zóna számára (a mester a 127.0.0.1:60053)
                a 61053-as porton figyel

És kombináltam ezt a port-továbbítással, hogy az 53-as portot elküldje a 61053-ra, ha
kapcsolat nélkül vagyok, és a 60053-ra, ha csatlakoztam. (Az új netfilter csomagot
használom 2.3.18 alatt, de a régi (ipchains) módszernek is működnie kell.)

Figyelem, ez nem fog pikk-pakk működni, mivel van egy apró hiba a 8.2-es
BIND-ben, melyet már jelentettem a fejlesztőknek, hogy megakadályozza egy másodlagos szerver
létrehozását az elsődlegessel megegyező IP címen (még ha külön porton is). Ez egy
egyszerű foltozás, és remélem, nemsokára belekerül.
```

- Kaptam információt arról is Karl-Max Wanger-től, hogyan hat egymásra a BIND az NFS-el és a portmapper-rel egy nagyobbrészt kapcsolat nélküli gépen:

```
Futtatni szoktam a saját named-emet az összes gépemen, melyek csak
alkalmilag csatlakoznak az Internetre modemen keresztül. A névszerver
csak gyorsítótárként működik, nincs jogosultsági területe, és mindenért
a root.cache állományban levő névszervereket kérdezi vissza. Ahogy az a
Slackware-nél megszokott, az nfsd és a mountd előtt van indítva.

Egyik gépemmel (egy Libretto 30-as notebookal) az volt a problémám,
hogy néha fel tudtam csatolni egy másik, a helyi LAN-omra csatlakozott
rendszerről, de nagyobbrészt ez nem működött. Ugyanez volt a jelenség,
függetlenül attól, hogy PLIP-et, egy PCMCIA ethernet kártyát vagy soros
eszközön keresztüli PPP-t használtam.

Némi találgatás és kísérletezés után azt fedeztem fel, hogy
a named minden bizonnyal belerondít az nfsd és mountd regisztrációs folyamatába,
amit induláskor a portmapper-rel kell elvégezniük (Ezeket a démonokat
szokás szerint bootoláskor indítom). A named indítása az nfsd és a mountd
után teljesen semlegesítette ezt a problémát.

Mivel nincsenek várható hátrányai az ilyen módosított boot szekvenciának,
ajánlom, hogy mindenki tegyen így az esetleges gondok elkerülése végett.
```

7. Hol tárolja a gyorsítótáras névszerver a gyorsítótárát? Van rá bármi mód, hogy ellenőrizzem a gy-

orsítótár méretét?

A gyorsítótár teljes mértékben a memóriában van tárolva, soha *nem* kerül kiírásra a lemezre. Valamennyiszer lelövöd a named-et, a gyorsítótár elveszik. A gyorsítótár semmilyen módon *nem* ellenőrizhető, a named gondozza néhány egyszerű szabály alapján, és ennyi. Nem ellenőrizheted a gyorsítótárat, vagy annak méretét semmilyen módon és semmiképp. Ha akarod, "kijavíthatod" ezt a named hackelésével. Ez azonban nem ajánlott.

8. Lementi a named a gyorsítótárat az újraindítások között? Megcsinálhatom, hogy így legyen?

   Nem, a named *nem* menti le, ha meghal. Ez azt jelenti, hogy a gyorsítótárat újra fel kell építeni minden alkalommal, amikor lelövöd és újraindítod a named-et. *Nincs* mód rá, hogy rávedd a named-et, hogy lementse gyorsítótárát egy állományba. Ha akarod, "kijavíthatod" ezt a named hackelésével. Ez azonban nem ajánlott.

9. Hogyan szerezhetek be egy tartományt? Fel akarom állítani (például) a `linux-rules.net` nevű tartományomat. Hogyan tehetem meg, hogy az általam kívánt tartományt hozzám rendeljék?

   Kérlek lépj kapcsolatba a hálózati szolgáltatóddal. Ők képesek lesznek segíteni neked. Kérlek vedd figyelembe, hogy a világ legtöbb részén pénzt kell fizetned egy tartományért.

10. Hogyan tehetem biztonságossá a DNS szerveremet? Hogyan állíthatok be felosztott DNS-t?

    Mindkettő haladó téma. A <http://www.etherboy.com/dns/chrootdns.html> honlap szól róluk. Nem fogom ezeket a témákat tovább magyarázni itt.

11 Hogyan válhatok képzettebb DNS adminná?

Dokumentáció és eszközök.

Létezik Valódi Dokumentáció. Azonnal olvasható (online) és nyomtatott. Ezek közül néhány elolvasása követelmény a kezdő DNS adminból egy haladó adminná váláshoz.

Megírtam a *The Concise Guide to DNS and BIND* (Nicolai Langfeldt, én) könyvet, kiadta a Que (ISBN 0-7897-2273-9). A könyv hasonlatos ehhez a HOGYANhoz, csak részletesebb, és mindenből sokkal több van benne. Le van fordítva lengyelre is, és *DNS i BIND* kiadva a Helion által (<http://helion.pl/ksiazki/dnsbin.htm> , ISBN 83-7197-446-9). Most van negyedik kiadásban a *DNS and BIND* Cricket Liu-tól és P. Albitz-től az O'Reilly & Associates gondozásában (ISBN 0-937175-82-X, előszeretettel nevezve a Cricket könyvnek). Egy másik könyv a *Linux DNS Server Administration*, Craig Hunt-tól, a Sybex kiadásában (ISBN 0782127363), még nem olvastam el. Egy másik követelmény a képzett DNS adminisztrátor számára a *Zen and the Art of Motorcycle Maintenance* Robert M. Pirsig-től.

Megtalálhatod a könyvemet azonnal olvasható formában (online), más könyvek tonnáival együtt, amelyek elektronikusan elérhetők mint előfizetéses szolgáltatás a

<http://safari.informit.com/> honlapon. Van még anyag a

<http://www.dns.net/dnsrd/> (DNS Resources Directory),

<http://www.isc.org/bind.html> címen; Egy GYIK, egy referencia kézikönyv (az ARM-nak szintén benne kell lennie a BIND disztribúcióban) éppúgy, mint papírok és protokoll definíciók, és DNS hackek (ezeket, és a legtöbb, ha nem az összes, lentebb említett RFC-t, szintén tartalmazza a DNS disztribúció). Többségét nem olvastam. A

<news:comp.protocols.tcp-ip.domains> hírcsoport a DNS-ről szól. Ezekhez adódóan van egy pár RFC a DNS-ről, a legfontosabbak valószínűleg az itt felsoroltak. Azok, melyeknek van BCP (Best Current Practice - Legjobb Jelenlegi Gyakorlat) száma, *erősen ajánlottak*.

/RFC 2671/ P. Vixie, *Extension Mechanisms for DNS (EDNS0)* 1999 augusztus. (DNS kiterjesztési me

/RFC 2317/

BCP 20, H. Eidnes et. al. *Classless IN-ADDR.ARPA delegation*, 1998 március. (Osztályon kívüli IN-ADDR.ARPA delegálás) Ez a CIDR-ről szól, vagy az osztályon kívüli alhálózatok fordított lekérdezéséről.

/RFC 2308/

M. Andrews, *Negative Caching of DNS Queries*, 1998 március. (A DNS lekérdezések negatív gyorsítótárazása) A negatív gyorsítótárazásról és a $TTL zónaállomány direktíváról.

/RFC 2219/

BCP 17, M. Hamilton and R. Wright, *Use of DNS Aliases for Network Services*, 1997 október. (A DNS álnevek használata hálózati szolgáltatások céljára) A CNAME használatáról.

/RFC 2182/

BCP 16, R. Elz et. al., *Selection and Operation of Secondary DNS Servers*, 1997 július. (A másodlagos DNS szerverek kiválasztása és működtetése)

/RFC 2052/

A. Gulbrandsen, P. Vixie, *A DNS RR for specifying the location of services (DNS SRV)*, October 1996 (Egy DNS RR a szolgáltatások helymeghatározására)

/RFC 1918/

Y. Rekhter, R. Moskowitz, D. Karrenberg, G. de Groot, E. Lear, *Address Allocation for Private Internets*, 1996.02.29. (Címlefoglalás magán Internetek számára)

/RFC 1912/

D. Barr, *Common DNS Operational and Configuration Errors*, 1996.02.28. (Gyakori üzemeltetési és beállítási DNS hibák)

/RFC 1912 Errors/

B. Barr */Errors in RFC 1912*. (Hibák az RFC 1912-ben/ Csak a

<http://www.cis.ohio-state.edu/~barr/rfc1912-errors.html> címen érhető el.

/RFC 1713/

A. Romao, *Tools for DNS debugging*, 1994.11.03. (A DNS hibakeresés eszközei)

/RFC 1712/

C. Farrell, M. Schulze, S. Pleitner, D. Baldoni, *DNS Encoding of Geographical Location*, 1994.11.01. (A földrajzi helyek DNS-be kódolása)

/RFC 1183/

R. Ullmann, P. Mockapetris, L. Mamakos, C. Everhart, *New DNS RR Definitions*, 1990.10.08. (Új DNS RR meghatározások)

/RFC 1035/

P. Mockapetris, *Domain names - implementation and specification*, 1987.11.01. (Tartománynevek - implementáció és specifikáció)

/RFC 1034/

P. Mockapetris, *Domain names - concepts and facilities*, 1987.11.01. (Tartománynevek - fogalmak és lehetőségek)

/RFC 1033/

M. Lottor, *Domain administrators operations guide*, 1987.1.01. (Tartomány-adminisztrátorok üzemeltetői útmutatója)

/RFC 1032/

M. Stahl, *Domain administrators guide*, 1987.11.01.

(Tartomány-adminisztrátorok útmutatója)

/RFC 974/

C. Partridge, *Mail routing and the domain system*, 1986.01.01. (Levéltovábbítás és a tartományrendszer)