DNS

Copyright (C) 1999 by Steve Litt, All rights reserved. Material provided as-is, use at your own risk.

<http://www.troubleshooters.com/linux/dns.htm>

WARNING!!!

Bind (the protocol behind DNS) has gone from version 4 to version 8. A version 4 setup WILL NOT work with version 8 bind (a version 8 named program).

When moving up to version 8, your best bet is to start from scratch, using your old setup files only as a guide to IP numbers and host/domain names.

A tutorial of version 4 DNS setup is contained in the [November 1998 Troubleshooting Professional Magazine](http://www.troubleshooters.com/tpromag/9811/dns.htm) (use browser's back button to get back here!).

This document describes setup of version 8 bind.

# Version differences:

If you try to run version 8 with version 4 files, then troubleshoot and change, you'll be at it for days.

The preferred method is to forget everything you ever knew about named version 4, and start over. That will take you a couple hours, and you'll notice it's a huge improvement.

|  |  |
| --- | --- |
| Version 4 | Version 8 |
| Default init file is /etc/named.boot | Default init file is /etc/named.conf. File layout is different. |
| /etc/resolv.conf is optional | /etc/resolv.conf is required. You can't ping dns defined url's without it. The file layout is different. |
| Extremely quirky and hard to configure | Logical and straightforward, as long as you DO NOT try to use your old configuration files. |
| Had some security leaks | Security has been tightened up. |
| Different domain names (zones) defined in same file. | Still possible to do it that way, but the preferred way is to have a distinct file for each zone. Doing it this way is robust and simple. |
| Named run directly | Named now run by a call to /usr/sbin/ndc, with an arg of "start", "stop", or "restart". Direct calls to /etc/rc.d/init.d/named still work, but... |
| CNAME was necessary for many DNS tasks | CNAME is no longer necessary, and in fact is not desirable. |

# Document Conventions

Here is what we're assuming:

* Linux server, eth0 hard IP at 192.168.102.3, is being used as a local DNS server.
* Machine's domain name is mydomain.cxm
* Machine's hostname is linuxhost.mydomain.cxm, or linuxhost for short
* Virtual IP 192.168.200.140-149 are hung off the loopback (lo) (linuxconf::config:networking:server tasks:IP aliases forvirtual hosts:lo)
* Virtual IP 192.168.200.146 points to /usr/doc/postgresql-6.3.2 (the PostgreSQL docs) via code in /etc/httpd/conf/httpd.conf
* Virtual IP 192.168.200.148 points to /usr/doc/LDP/nag (the Network Administrator Guide) via code in /etc/httpd/conf/httpd.conf
* http://192.168.200.148/ is browsable in Netscape from any machine on the 192.168.100 subnet (ip forwarding...)

# The /etc/resolv.conf Catch 22

Any problem with DNS, especially reverse DNS (in-addr.arpa), will prevent telnetting into the machine if /etc/resolv.conf exists, so therefore you'll need to remove resolv.conf if you're configuring via telnet.

Other apps may not work, and the machine may even fail to boot.

But in Version 8 bind, DNS \*will not work\* without resolv.conf.

Here are some suggestions:

* Until your DNS is solid and tested, NEVER reboot the server without first renaming /etc/resolv.conf or else renaming ALL your reverse dns files.
* If you fail to observe the rule above, go in with a boot disk and rescue disk. [Here are some instructions](http://www.troubleshooters.com/tpromag/9811.htm#BootFloppies).
* With /etc/resolv.conf renamed, run several telnet sessions before renaming it back to /etc/resolv.conf and then doing a /usr/sbin/ndc restart. Do not close or disconnect any of those telnet sessions -- you won't be able to restore them.

# The Files

The example below will create a simple virtual hosting web server DNS, where the physical server is at 192.168.102.3, and the websites are at virtual IP's 192.168.200.140-149, hung off the lo interface. Upon completion of this section, you will be able to see the following Linux documentation with your browser:

|  |  |  |
| --- | --- | --- |
| http://www.sqldocs.cxm | 192.168.200.146 | The PostgreSql docs that came with your Linux distro |
| http://www.nag.cxm | 192.168.200.148 | The Network Administrators Guide that came with your distro |

The prerequesites is simply to be able to pull up these websites using http://192.168.200.146/ and http://192.168.200.148/ respectively. Remember, if you can't access it by IP, you'll never be able to access it by name either. The mapping between IP and website root document directory is accomplished in /etc/httpd/conf/httpd.conf.  If you get a "forbidden" message accessing, for example, an IP mapped to /usr/doc/LDP/nag, you might need to put a

<Directory /usr/doc/LDP/nag>

options All

allow from all

</Directory>

in /etc/httpd/conf/access.conf..

Once you've completed the files, restart named like this:

/usr/sbin/ndc restart

Remember, every time you edit a file with a serial number, you must increase that serial number. I suggest you use serial numbers of the form yyyymmdd##, so you can make up to a hundred changes per day.

NOTE: While your DNS is being set up, never reboot your server without renaming /etc/resolv.conf. When the boot completes, rename it back to /etc/resolv.conf. A bad reverse DNS can prevent booting, requiring a boot and rescue bust-in. Save yourself the trouble, rename.

With version 8, the files are so simple I've forgone in-file comments. Those comments would just reduce readability.

**/etc/resolv.conf**

[root@linuxhost /etc]# cat resolv.conf

search mydomain.cxm  mydomain.cxm

nameserver 127.0.0.1

**Notes:**

The 127.0.0.1 reference is very necessary. If the server is administering just part of the mydomain.cxm domain (let's say corporate), that first line would be

search corporate.mydomain.cxm mydomain.cxm

Note that if there are \*ANY\* problems in reverse DNS, the existance of this file can cause failure to run telnet, ftp, or sendmail, and can prevent boot. Therefore, it's essential during DNS development and debugging to rename this file before rebooting. If this advice isn't followed, you can bust back in with a boot and rescue diskette and disable whatever is hanging the boot (usually amd and/or sendmail).

/etc/named.conf

options {

        directory "/var/named";

        /\*

         \* If there is a firewall between you and nameservers you want

         \* to talk to, you might need to uncomment the query-source

         \* directive below.  Previous versions of BIND always asked

         \* questions using port 53, but BIND 8.1 uses an unprivileged

         \* port by default.

         \*/

        // query-source address \* port 53;

};

//

// a caching only nameserver config

//

zone "." {

        type hint;

        file "named.ca";

};

zone "0.0.127.in-addr.arpa" {

        type master;

        file "named.local";

};

zone "mydomain.cxm" {

        notify no;

        type master;

        file "named.forward";

};

zone "sqldocs.cxm" {

        notify no;

        type master;

        file "named.sqldocs.cxm";

};

zone "nag.cxm" {

        notify no;

        type master;

        file "named.nag.cxm";

};

zone "102.168.192.in-addr.arpa" {

        notify no;

        type master;

        file "named.102.168.192";

};

zone "200.168.192.in-addr.arpa" {

        notify no;

        type master;

        file "named.200.168.192";

};

**Notes:**

The options block at the top tells the root directory for further DNS files. This is very similar to the version 4 dns. Zone "." (Named.ca) is a caching file pointing to the world's top level DNS servers.Zone "0.0.127.in-addr.arpa" is reverse DNS for the loopback interface lo. Zone "mydomain.cxm" is forward DNS for the server's domain, and is different from the per-domain files. Zones "sqldocs.cxm" and "nag.cxm" are per-domain forward DNS. Zone "102.168.192.in-addr.arpa" is reverse DNS for the server's subnet (not domain). Zone "200.168.192.in-addr.arpa" is reverse DNS for the websites' subnet (not domain).

This file serves as a roadmap for the rest of your DNS setup. Note how much easier this is than the old named.boot setup.

**named.local**

@       IN      SOA     localhost. root.localhost.  (

                                      1997022700 ; Serial

                                      28800      ; Refresh

                                      14400      ; Retry

                                      3600000    ; Expire

                                      86400 )    ; Minimum

              IN      NS      localhost.

1       IN      PTR     localhost.

**Notes:**

Forward DNS for the loopback interface (lo). Use exactly as is. It probably comes looking just like this on the Linux distro.

**named.forward**

@       IN      SOA     ns.mydomain.cxm. hostmaster.mydomain.cxm. (

                        1999021004      ; serial, todays date + todays serial #

                        8H              ; refresh, seconds

                        2H              ; retry, seconds

                        1W              ; expire, seconds

                        1D )            ; minimum, seconds

                NS      ns              ; Inet Address of name server

                MX      10 mail         ; Primary Mail Exchanger

localhost       A       127.0.0.1

ns              A       192.168.102.3

linuxhost       A       192.168.102.3

mail            A       192.168.102.3

www             A       192.168.102.3

**Notes:**

This is the forward DNS for the server's domain. The meaning of the @ sign on the first line is the Zone variable string in /etc/named.conf, so in this case it's mydomain.cxm. Note that except in the first line, the word "IN" is now optional. The localhost A line is necessary for proper DNS server functioning. The ns A line provides portability, so if the name server is changed, the ip is changed here but everything else remains the same. Linuxhost is the actual hostname of the machine, mail is an alias (though it's done with A, not CNAME) used in mail routing, and www is an http alias.

**named.sqldocs.cxm**

@       IN      SOA     ns.mydomain.cxm. hostmaster.mydomain.cxm. (

                        1999021005      ; serial, todays date + todays serial #

                        8H              ; refresh, seconds

                        2H              ; retry, seconds

                        1W              ; expire, seconds

                        1D )            ; minimum, seconds

                NS      ns.mydomain.cxm.         ; Inet Address of name server

                MX      10 mail.mydomain.cxm.    ; Primary Mail Exchanger

@      A 192.168.200.146

www    A 192.168.200.146

**Notes:**

I'm using the 192.168.102.3 as the mail exchanger, so the only significance here is that @ (sqldocs.cxm) and alias (but not aliased wth CNAME) map to 192.168.200.146, a virtual IP hung off the lo interface.

**named.nag.cxm**

@       IN      SOA     ns.mydomain.cxm. hostmaster.mydomain.cxm. (

                        1999021005      ; serial, todays date + todays serial #

                        8H              ; refresh, seconds

                        2H              ; retry, seconds

                        1W              ; expire, seconds

                        1D )            ; minimum, seconds

                NS      ns.mydomain.cxm.         ; Inet Address of name server

                MX      10 mail.mydomain.cxm.    ; Primary Mail Exchanger

@      A 192.168.200.148

www    A 192.168.200.148

**Notes:**

I'm using the 192.168.102.3 as the mail exchanger, so the only significance here is that @ (nag.cxm) and alias (but not aliased wth CNAME) map to 192.168.200.148, a virtual IP hung off the lo interface. Note the only distinction between this file and named.sqldocs.cxm above it is the ip address of @. This is the beauty of version 8. To add a new website to the series, copy another per-domain forward file, then change the ip. This is why I chose not to put comments in these files -- too much likelihood of the comments becoming misleading with copying.

**named.102.168.192**

@       IN      SOA     ns.mydomain.cxm. hostmaster.mydomain.cxm. (

                          1999021004 ; Serial, todays date + todays serial

                          8H      ; Refresh

                          2H      ; Retry

                          1W      ; Expire

                          1D)     ; Minimum TTL

                  NS      ns.mydomain.cxm.

3                 PTR     linuxhost.mydomain.cxm.

3                 PTR     ns.mydomain.cxm.

3                 PTR     mail.mydomain.cxm.

3                 PTR     www.mydomain.cxm.

**Notes:**

Reverse DNS for the 192.168.102 subnet (the server's subnet).Simply define PTR records for each name for each IP in the subnet. Note that where several names go with a single IP, reverse DNS will bring back the first PTR record for that IP.

**named.200.168.192**

@       IN      SOA     ns.mydomain.cxm. hostmaster.mydomain.cxm. (

                          1999021004 ; Serial, todays date + todays serial

                          8H      ; Refresh

                          2H      ; Retry

                          1W      ; Expire

                          1D)     ; Minimum TTL

                  NS      ns.mydomain.cxm.

146               PTR     sqlocs.cxm.

146               PTR     www.sqlocs.cxm.

148               PTR     nag.cxm.

148               PTR     www.nag.cxm.

**Notes:**

Reverse DNS for the 192.168.200 subnet (the series of websites with virtual IPs hung off the lo interface).Simply define PTR records for each name for each IP in the subnet, generally one with just the domain and one with a prepended www. Note that where several names go with a single IP, reverse DNS will bring back the first PTR record for that IP. Note that if individual domains had their own mail servers or ftp servers or anything like that, those would get PTR records here too.

**named.ca**

;       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

**Notes:**

Caching DNS file has routes names and IPs of root servers. Comes with your Linux distro, should be updated every few months to reflect the changes in worldwide DNS.

# Testing

NOTE: While your DNS is being set up, never reboot your server without renaming /etc/resolv.conf. When the boot completes, rename it back to /etc/resolv.conf. A bad reverse DNS can prevent booting, requiring a boot and rescue bust-in. Save yourself the trouble, rename.

## Test with ping:

Ping linuxhost.mydomain.cxm, sqldocs.cxm, www.sqldocs.cxm, nag.cxm, www.nag.cxm from the server, and from other machines in its subnet. Run each ping several times to make sure the result isn't a fluke.

If the pings any pings don't work, ping the associated IP to make sure it's really a DNS problem. Note that your hostname (linuxhost.mydomain.cxm) can usually be pinged even with a bad DNS, because it's in the hosts file.

If pings consistently come back with a "ping: unknown host", check that you have a correctly formed /etc/resolv.conf. A missing or incorrect /etc/resolv.conf will produce exactly this result, yet nslookup will work.

## Testing with nslookup:

nslookup gives more details than ping, and is generally used as a diagnostic when ping or some other test fails.

Here are some nslookup sessions producing correct results:

[root@linuxhost /etc]# nslookup www.sqldocs.cxm

Server:  localhost

Address:  127.0.0.1

Name:    www.sqldocs.cxm

Address:  192.168.200.146

[root@linuxhost /etc]# nslookup 192.168.200.146

Server:  localhost

Address:  127.0.0.1

Name:    sqlocs.cxm

Address:  192.168.200.146

[root@linuxhost /etc]# nslookup linuxhost.mydomain.cxm

Server:  localhost

Address:  127.0.0.1

Name:    linuxhost.mydomain.cxm

Address:  192.168.102.3

[root@linuxhost /etc]# nslookup 192.168.102.3

Server:  localhost

Address:  127.0.0.1

Name:    linuxhost.mydomain.cxm

Address:  192.168.102.3

[root@linuxhost /etc]#

Interpreting the results:

* Line 1: The server should ALWAYS be localhost, unless you're renamed /etc/resolve.conf as a temporary test, in which case it should be the hostname
* Line 2: The server's address should ALWAYS be 127.0.0.1, unless you've renamed /etc/resolv.conf as a temporary test, in which case it should be 0.0.0.0.
* Line 3: The name should always be the fully qualified domain name appearing as an nslookup arg (forward dns) or serving as an answer to a lookup on the IP arg to nslookup (reverse dns).
* Line 4: The Address should always be the IP appearing as an nslookup arg (reverse dns) or serving as an answer to a lookup on the Fully Qualified Doman Name arg to nslookup (forward dns).

Forcing lookups on other servers:

You can force a lookup on a different server by using its IP as a second nslookup argument, but be careful -- if it's wrong nslookup will hang for a minute or two til it times out.

You can also invoke a second argument to tell nslookup exactly what IP to use as a DNS server. That can often help diagnose a bad or missing /etc/resolv.conf (missing resolv.confs give 0.0.0.0 as the server address of single arg nslookups).

## Testing with Telnet

Telnet is the "miner's canary" of reverse DNS. If you're going to have a reverse DNS problem, the first evidence will come with slow or timed out telnet connections to the server. When reverse DNS is configured properly, connecting with telnet should bring up the login prompt in less than a second.

If telnet takes longer than that (and it will usually take 30 seconds or longer if there's a problem), rename /etc/resolv.conf (no need to restart named) and try again. If it's instantaneous after the rename, you've narrowed it to a bad reverse DNS.

NEVER reboot the computer while telnet is slow to connect. The computer may fail to complete its boot. If you must reboot, rename /etc/resolv.conf first.

If everything looks good in your reverse DNS setup but telnet is still slow, be sure to check for spaces before the statements in the reverse DNS files. A space will cause the reverse DNS to fail silently, and slow or hang Telnet.

## DNSWalk: The Ultimate Test

This is beyond the scope of this tutorial, but if you really want to test the daylights out of your DNS setup, procure

http://tools.kali.org/information-gathering/dnswalk

Once dnswalk is installed, just go

perl dnswalk mydomain.cxm.

to invoke it and see all your mistakes. The trailing period is essential. Note that it will correctly deduce that you've committed a no-no by having only 1 DNS server,   
but of course if you have only one Linux machine you have little choice. Be sure to repeat this test for every domain you host on the machine.

# Troubleshooting

## Step 5: General Maintenance

* Make sure the IPs you're providing naming services for work right. If it won't work by number, it won't work by name.
* Make sure named is running with the command ps ax | grep named
* Check your distribution's documentation to see whether you're using bind 4 or bind 8.
* If it's bind 8, rename any /etc/named.boot that may exist. named.boot just confuses matters.
* In /etc/named.conf, make sure each zone's domain is spelled right, no domains end with a period (the period goes in the zone files, not the init file), make sure the directory entry in the options section is correct, and make sure all zone filenames are spelled correctly. Make sure /etc/named.conf is structured approximately the way this document tells you to.
* FOR EACH ZONE FILE
  + Make sure there are no spaces at the start of a left hand domain or an at sign, ESPECIALLY in the IN SOA line.
  + Make sure relative names have no period at the end, and that fully qualified names do.

## Step 6: Narrow it down

Here are some of the many tests you can do to narrow the scope of the root cause. Within this article, when I say "restart" it means

/usr/sbin/ndc restart

|  |  |
| --- | --- |
| ps ax | grep named | Is named running? Do this test often, because if your results are coming from non-running named and you think they're coming from something else, you'll go around in circles. |
| Rename /etc/resolv.conf | This is a quick way to focus or divert suspicion from reverse dns. If renaming it causes telnet and other apps to fire up right, instead of taking inordinately long, then reverse dns is probably to blame, in which case you do the more decisive test right below this one, AFTER putting back /etc/resolv.conf. |
| Rename reverse dns files, make sure /etc/resolv.conf exists and is correct, then restart. | This will remove reverse dns but leave forward dns intact. Verify that your forward dns works and reverse doesn't (see next test). If telnet runs correctly with reverse dns renamed but not with them intact, the problem's in your reverse dns. If there are several reverse dns files, rename and un-rename combinations until you find which one(s) have a problem. Enable exactly one with a problem, look it over with a fine tooth comb, and if you can't see anything wrong, start experimenting with that one file. |
| nslookup www.nag.cxm  nslookup 192.168.102.148 | After renaming reverse dns (or in other situations), you can verify that forward works and reverse doesn't with nslookup. If forward works, nslookup domainname will deliver the IP for that domain name. If reverse works, nslookup IP will deliver (the first listed PTR) domain name for that IP. |
| Read the server line of nslookup | Determine if you're looking in the right place (127.0.0.1) |