Deploying & Configuring a DNS Server on OpenServer 6 or UnixWare 7

Kirk Farquhar

Content

Introduction

- Versions of Bind on SCO
  - OpenServer 6 is Bind 8.4.4
  - UnixWare 7.1.4 MP4 is Bind 9.4.2
- Each is a significant upgrade in features from Bind 4 used in 5.0.7
- Resolver behavior is very different
- New implementations are much stricter in standards compliance

Key Fixes

- Fixes
  - SECURITY: BIND Denial of Service Vulnerability --
  - BIND version 8.4.4 is vulnerable to a remote denial of service attack, caused by a buffer overflow in the in q_usedns array.
  - This is fixed in OSR6 MP2

Issues between Bind 8 & 9

- **BIND 8 is forgiving**
  - Minor errors in **named.con**f are acceptable
  - Zone files that contain minor syntax errors are read and processed
- **BIND 9 is strict**
  - Will not read **named.conf** files with any errors
  - Will not load zones with syntax errors

Bind 9 New Features

- **In Bind 8 stub zones allow automatic delegation**
  - A name server with a stub zone would look up NS records for that zone periodically and “promote” them into the zone’s parent
- **In Bind 9 the promotion doesn’t occur**
  - The NS records are still looked up, but they’re not “promoted” into the parent zone
- **You can still use stub zones to specify which name servers to use to resolve parts of your domain**

Bind Differences

- Differences between 8 & 9
  - Bind 9 includes security fixes including secure DNS
  - Bind 9 adds IPV6 support
  - Bind 9 is more standards compliant
  - Control Bind 8 with ndc
  - Control Bind 9 with rndc
    - rndc uses a secure channel
    - Syntax & commands are different
    - Configured using *rndc.conf* and *controls*

Bind 9 Differences

- In Bind 8 zone transfer each message packet contains one resource record only
- In Bind 9 a packet can carry multiple records
  - This creates a problem with Bind 4 (read OSR5) and older Microsoft servers, which can't accept multiple records per packet
  - You can override this with a directive in named.conf

```
server 192.168.3.1{
       transfer-format one-answer;
 };
```

- Bind 8 allows multiple CNAME records for one address
- In Bind 9 this isn't allowed
  - You must use A records
- Bind 9 log file syntax is different

Configuring DNS on Your SCO Server

- SCO OpenServer 6 and UnixWare 7 provide the **DNS Manager** as a convenient way of setting up and maintaining these files for master, slave, and stub name servers, or for servers that provide a mixture of these services.
- If you choose not to use the **DNS Manager** or the **Client Manager**, you must edit the configuration files yourself

Administering DNS on UW7 – File Requirements

- If you are enabling a slave, stub, or caching−only server, the minimum configuration is to set up
- **named.conf**
- **named.local**
- **root.cache**
- For a slave or stub server, you should also configure a backup file that will be used to hold transferred zone data.
- In the case of a stub server, only a limited number of records are transferred and stored.
- To configure a server which is master for at least one zone, the suggested configuration is to set up
- **named.conf**
- **named.local**
- **root.cache**
- Plus for each zone a pair of suitably named
- **named.hosts**
- **named.rev**
- Additionally, you can configure a single **named.soa** file that defines one SOA (Start of Authority) record to be included by all the zone **named.hosts** and *named.rev* files, or you can configure an individual SOA record for each zone.
- To configure the use of name service by client software, edit the **resolv.conf** file

Running the DNS Server- Utilities

- **DNS utilities and daemons**
- The **named** daemon and utilities form the heart of DNS operations:
- This daemon must be running for DNS to be operational on all but remote servers. After you enable your server, **named** is invoked each time your system enters multiuser mode.
- It reads information found in the configuration file, *named.conf*, and takes appropriate actions
  - Priming the cache,
  - Configuring Domain Name System (DNS) servers
  - Name service clients accessing zone files, and so on.
- **named** can also be invoked from the command line

Running the DNS Server -Utilities

- ***named−xfer***
  - This command enables you to transfer a DNS zone from one server to another in asynchronous mode so that **named** can continue processing requests.
- ***nslookup***
  - This command allows you to request DNS information, including names, addresses, and other resource records, from any server you can reach from your computer.
  - You can use **nslookup** to request a single record or start an **nslookup** session to request multiple records from one or more servers.
- ***dig***
  - This command is similar to **nslookup** but with a slightly different syntax.
- ***host***
  - This command is useful for finding out information about individual or multiple hosts.

Administering DNS on UW7 – ndc on OSR6

- ***ndc reload***
  - Causes **named** to read *named.conf* and reload the database, overwriting cached data.
  - This is useful when you make a change to a data file and you want **named**'s internal database to reflect the change.
  - The **reload** option also has the effect of scheduling all secondary zones for serial−number checks, which could lead to zone transfers ahead of the usual schedule. Normally, serial−number comparisons are done only at the intervals specified in the zone's SOA record.
- ***ndc dumpdb***
  - Dumps the current database and cache to */var/tmp/named_dump.db*.
  - This gives you an indication to whether the database was loaded correctly.
- ***ndc trace***
  - Turns on debugging.
  - Each following USR1 increments the debug level. The output goes to
    - */var/tmp/named.run*.
- ***ndc notrace***
  - Turns off debugging completely.
- ***ndc querylog***
  - Toggles tracing of all incoming queries.
  - The trace is sent to */usr/adm/syslog* and provides a largeamount of data.

Starting and Stopping the Server

- By default ndc will started as a daemon service by /etc/rc2.d/S85tcp
- Can be manually controlled by
- /etc/ndc stop
- /etc/ndc start
- /etc/ndc reload

Administering DNS on UW7 – rndc on UW7

- The "ndc" program has been replaced by "rndc", which is capable of remote operation.
- Unlike ndc, rndc requires a configuration file
- Some of the ndc commands are still not implemented in rndc.
- rndc reads its default configuration file,

  /etc/inet/rndc.conf
- to determine how to contact the name server and decide what algorithm and keys is should use
- Syntax

  rndc -c config-file -M -m -p port# -s server -v -y key_id command
- If no server is supplied on the command line, the host named by the default-server clause in the option statement of the configuration file will be used

Administering DNS on UW7 – rndc on UW7

- Rndc is provided on Bind 9 in addition to ndc
- rndc has its own configuration file /etc/inet/rndc.conf

```
options {
    default-server  localhost;
    default-key     samplekey;
};
server localhost {
    key     samplekey;
};
key samplekey {
    algorithm hmac-md5;
    secret
  "c3Ryb25nIGVub3VnaCBmb3IgYSBtYW4gYnV0IG1hZGUgZm9yIGEgd
  29tYW4K";
};
```

- rndc will by default use the server at localhost (127.0.0.1) and the key called samplekey

Administering DNS on UW7 – rndc on UW7

- To generate a random secret with dnssec-keygen:

-
  ```
  $ dnssec-keygen -a hmac-md5 -b 128 -n user rndc
  ```
- The base-64 string will appear in two files,
  - Krndc.+157.+{random}.key and
  - Krndc.+157.+{random}.private .
- After extracting the key to be placed in the rndc.conf and named.conf key statements, the .key and .private files can be removed
- The name server must be configured to accept rndc connections and to recognize the key specified in the rndc.conf file, using the controls statement in named.conf

- **h2n** translates */etc/hosts* to DNS files and creates a BIND boot file.
- This tool can be run once or many times.
- After converting your host table to DNS format, you can manually maintain the DNS files, or you can maintain the host table and run **h2n** each time you modify **/etc/hosts**.
- **h2n** automatically increments the serial number in each DNS file when it makes a new one.

Building a DNS Server with H2N

- **h2n** generates files starting with the prefix *db*. These are called ``db files''.
- The domain data are stored in a file called *db.DOMAIN*, where ***DOMAIN*** defaults to the first label in your domain name (given with the **-d** option).
- The address-to-name data are stored in files named *db.NET*, where ***NET*** is a network number (given with the **-n** option).
- An email address for the person responsible for the domain is included in the start-of-authority record for the domain (given with the **-u** option).
- Each time **h2n** is run, it generates the DNS files from scratch.
- Any changes you manually made to the DNS files are lost.
- If you'd like to add resource records to a db file generated by **h2n**, put your RRs in a file prefixed with "spcl" instead of "db".
  - **h2n** will include this file's data by adding an $INCLUDE directive to the end of the db file.

Building a DNS Server with H2N

- By default, **h2n** will generate an MX record with a weight of 10 that points to the host itself as the mail exchanger.
- Additional MX records can be added to all hosts by using **-m** options.
- To suppress generating the default MX record for a host, include "[no smtp]" in that host's host table comment
- Another comment section flag is "[TTL=**num**]", where **num** is a specific time-to-live value to use for the resource records pertaining to the canonical hostname in the host table.
- This is useful for setting artificially high or low TTL values for individual hosts.
  - For example, if you are going to be moving a host to a new IP address, you can use this to set a low TTL value such as 900 (seconds)

Building a DNS Server with H2N

- Example
- Create name server data for networks 192.168.3 and 192.168.150 in scotec.net.

```
h2n -d movie.edu -n 192.168.3 -n 192.168.150
```

Building a DNS Server with H2N

- Create name server data for networks 192.168.3 and 192.168.150 in scotec.net.
- Eliminate lines in the host table that contain fx.scotec.net and include MX records for all hosts pointing to the mail hub, stn6a.scotec.net.
- Afterwards, look for additional resource records in the file "spcl.scotec.net" and append them to "db.scotec" via an $INCLUDE directive.
- Include all of the options in a file.

```
h2n -f option_file
```

- The file ***option_file*** contains the following lines:

```
-d scotec.net
spcl=spcl.scotec.net
-n 192.168.3 -n 192.168.150
-e fx.scotec.net
-m 50: stn6a.scotec.net
```

Creating our DNS with H2N

- Sample Hosts file

```
# Internet host table
127.0.0.1       localhost
#
192.168.3.63    stn6c.scotec.net stn6c
192.168.3.62    stn6b.scotec.net stn6b
192.168.3.16    stn6a.scotec.net stn6a
192.168.3.18    client1.scotec.net client1
#
```

Creating our DNS with H2N

- Command Syntax

```
h2n -d scotec.net -n 192.168.3 -d 127.0.0 -u kirkf@scotec.net
```

- Creates DNS files for domain scotec.net and networks 192.168.3.x and 127.0.0.x
- Uses kirkf@scotec.net as authoritative contact

Creating our DNS with H2N

named.conf file

```
options {
        directory "/etc/named.d";
};
zone "0.0.127.IN-ADDR.ARPA" in {
        type master;
        file "db.127.0.0";
        notify no;
};
zone "scotec.net" in {
        type master;
        file "db.scotec";
};
zone "3.168.192.IN-ADDR.ARPA" in {
        type master;
        file "db.192.168.3";
};
zone "0.0.127.IN-ADDR.ARPA" in {
        type master;
        file "db.127.0.0";
};
zone "." in {
        type hint;
        file "db.cache";
};
```

Creating our DNS with H2N

db.scotec file

- @ IN SOA stn6a.scotec.net. kirkf.scotec.net. (1 10800 3600 604800 86400)
-   IN NS stn6a.scotec.net.
- localhost   IN A   127.0.0.1
- localhost   IN MX   10 localhost.scotec.net.

- stn6a   IN A   192.168.3.16
- stn6a   IN MX   10 stn6a.scotec.net.

- client1   IN A   192.168.3.18
- client1   IN MX   10 client1.scotec.net.

- stn6b   IN A   192.168.3.62
- stn6b   IN MX   10 stn6b.scotec.net.

- stn6c   IN A   192.168.3.63
- stn6c   IN MX   10 stn6c.scotec.net.

Creating our DNS with H2N

- db.127.0.0 file

```
@ IN  SOA stn6a.scotec.net. kirkf.scotec.net. ( 1 10800 3600 604800
   86400 )
  IN  NS  stn6a.scotec.net.

1.0.0.127.IN-ADDR.ARPA.         IN  PTR   localhost.scotec.net.
```

- db.192.168.3 file

```
@ IN  SOA stn6a.scotec.net. kirkf.scotec.net. ( 1 10800 3600 604800
   86400 )
  IN  NS  stn6a.scotec.net.

16.3.168.192.IN-ADDR.ARPA.     IN  PTR   stn6a.scotec.net.
62.3.168.192.IN-ADDR.ARPA.     IN  PTR   stn6b.scotec.net.
63.3.168.192.IN-ADDR.ARPA.     IN  PTR   stn6c.scotec.net.
18.3.168.192.IN-ADDR.ARPA.     IN  PTR   client1.scotec.net.
```

Creating our DNS with H2N

- **conf.cacheonly file**

```
options {
        directory "/etc/named.d";
};

zone "0.0.127.IN-ADDR.ARPA" in {
        type master;
        file "db.127.0.0";
        notify no;
};

zone "." in {
        type hint;
        file "db.cache";
};
```

Administering DNS with DNS Manager

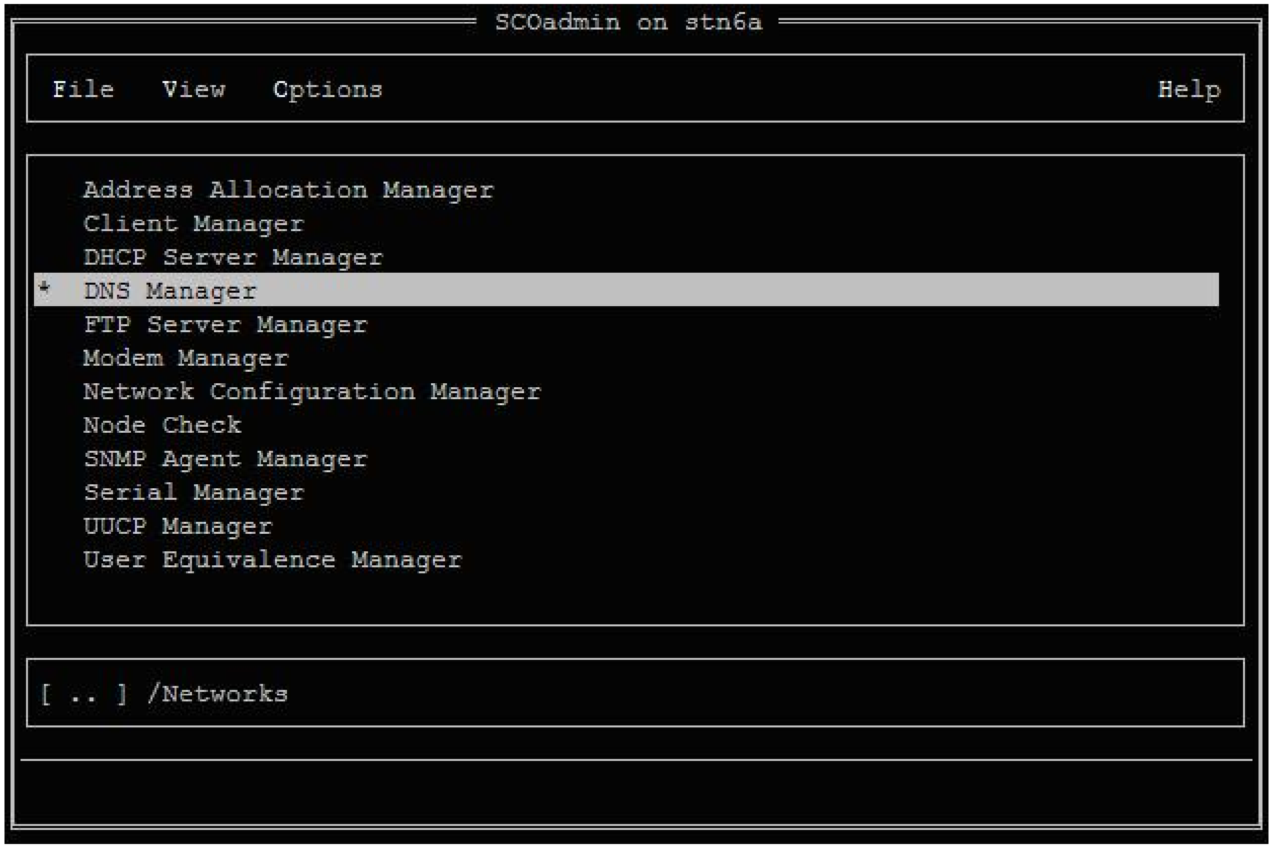
- Using the DNS Manager
- Start the SCOadmin launcher by entering scoadmin on the command line, then select Networking->DNS Manager.
- Enter scoadmin DNS Manager on the command line.
- The DNS Manager is intended for setting up zone files from new.
- It will read and preserve the contents of existing zone files in most cases but it does not understand the $include directive or types of resource record other than A, CNAME, HINFO, MX, NS, PTR, RP, and TXT.
- You will not be able to use the DNS Manager to configure DNS on a remote host unless the remote host's name is locally resolvable to an IP address using an */etc/hosts* entry, DNS, or NIS.
- If you exit the DNS Manager and the host is not currently configured as a DNS name server, you are asked if you want to configure it as a caching−only server.
  - This allows you to quickly set up a system as a caching−only name server.

Administering DNS with DNS Manager

- **WARNING:**
  - For correct operation of the **DNS Manager** and other SCOadmin managers on a host that you are configuring remotely there must also be an entry for the *localhost* address (127.0.0.1) in the */etc/hosts* file on the remote host.

Administering DNS with DNS Manager

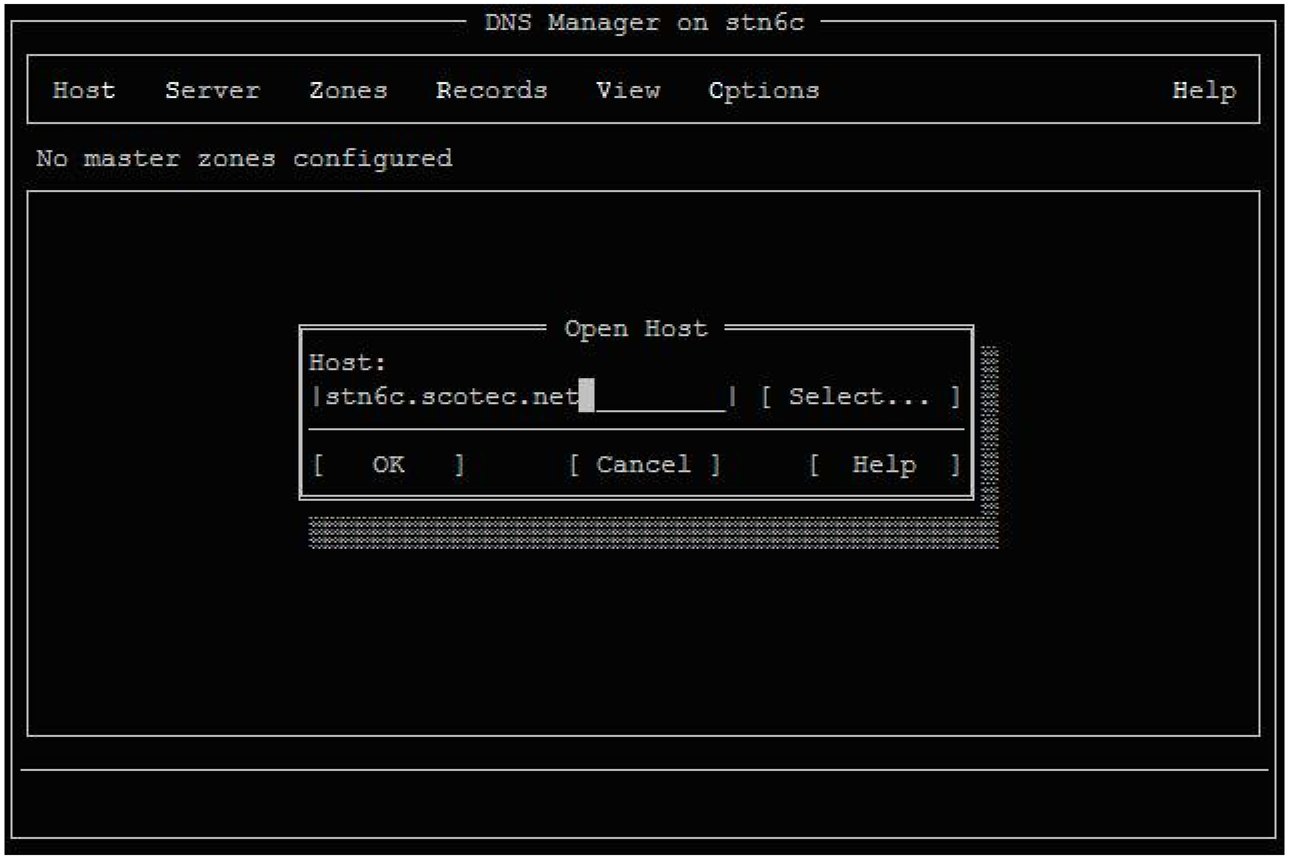
- Access the DNS Manager fro SCOadmin



```
SCOadmin on stn6a

File   View   Options                                      Help

   Address Allocation Manager
   Client Manager
   DHCP Server Manager
*  DNS Manager
   FTP Server Manager
   Modem Manager
   Network Configuration Manager
   Node Check
   SNMP Agent Manager
   Serial Manager
   UUCP Manager
   User Equivalence Manager

[ .. ] /Networks
```

Administering DNS with DNS Manager

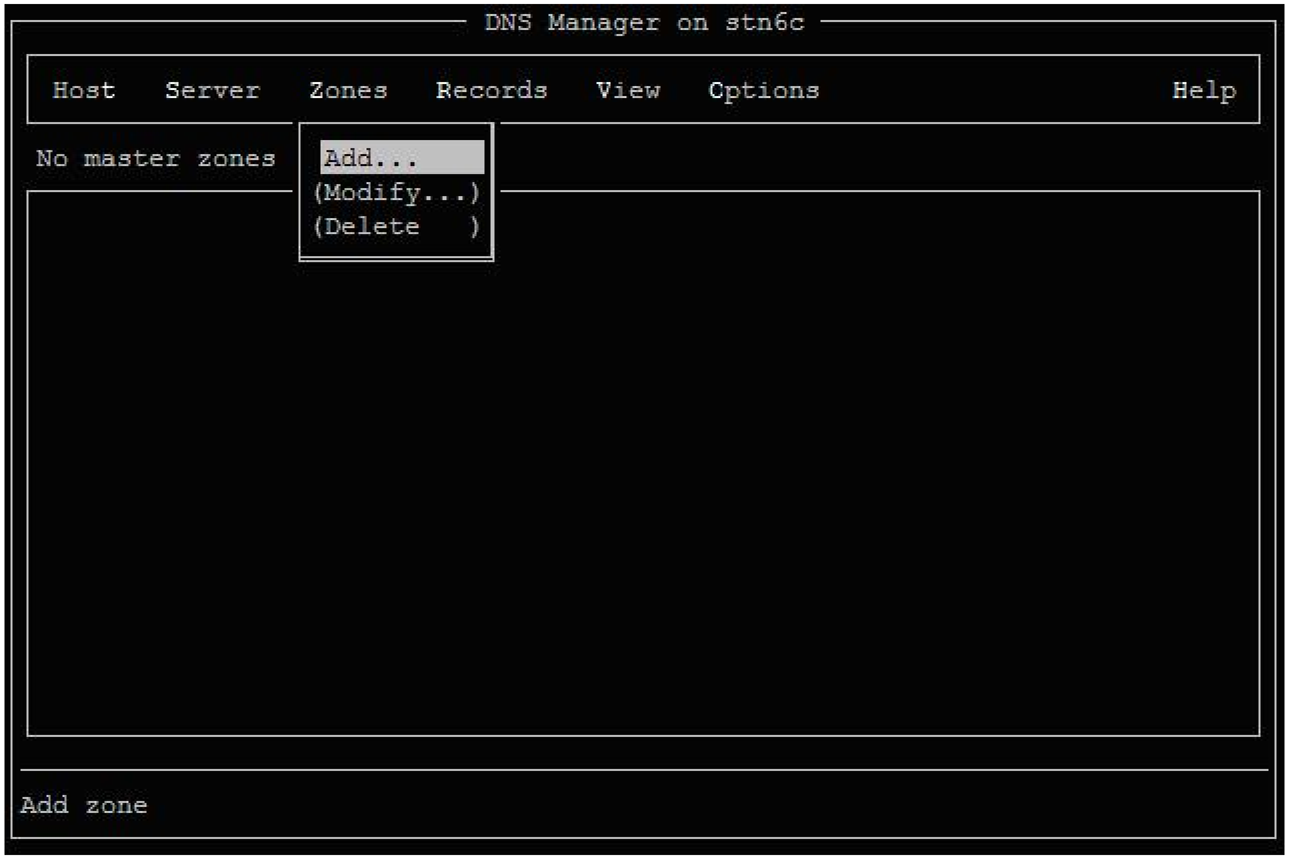
- Select the host you want to build a server on



```
DNS Manager on stn6c

 Host   Server   Zones   Records   View   Options                  Help

No master zones configured

                     Open Host
           Host:
           |stn6c.scotec.net_________| [ Select... ]

           [   OK   ]      [ Cancel ]      [  Help  ]
```

Administering DNS with DNS Manager

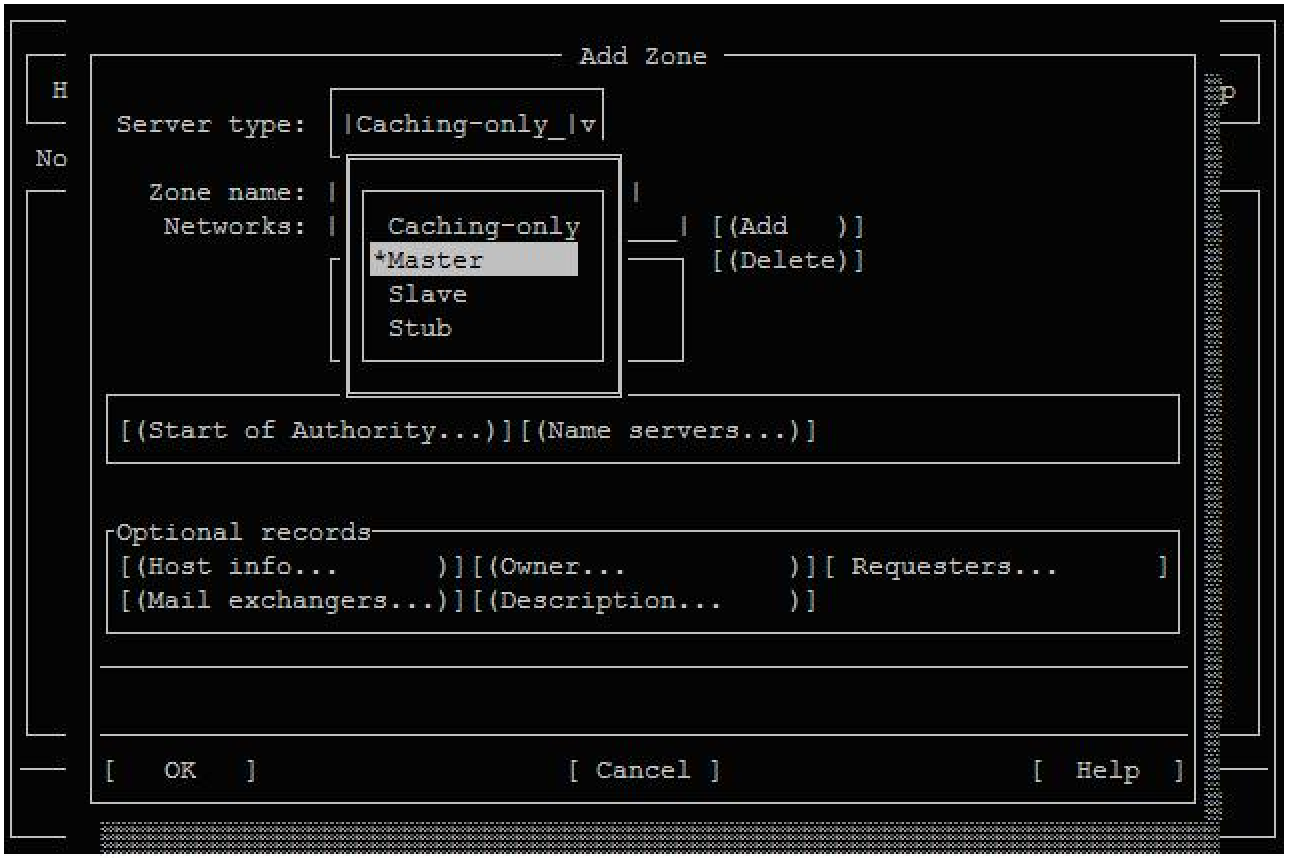
- Create your zones



```
DNS Manager on stn6c

Host   Server   Zones   Records   View   Options        Help

No master zones   Add...
                  (Modify...)
                  (Delete    )

Add zone
```

Administering DNS with DNS Manager

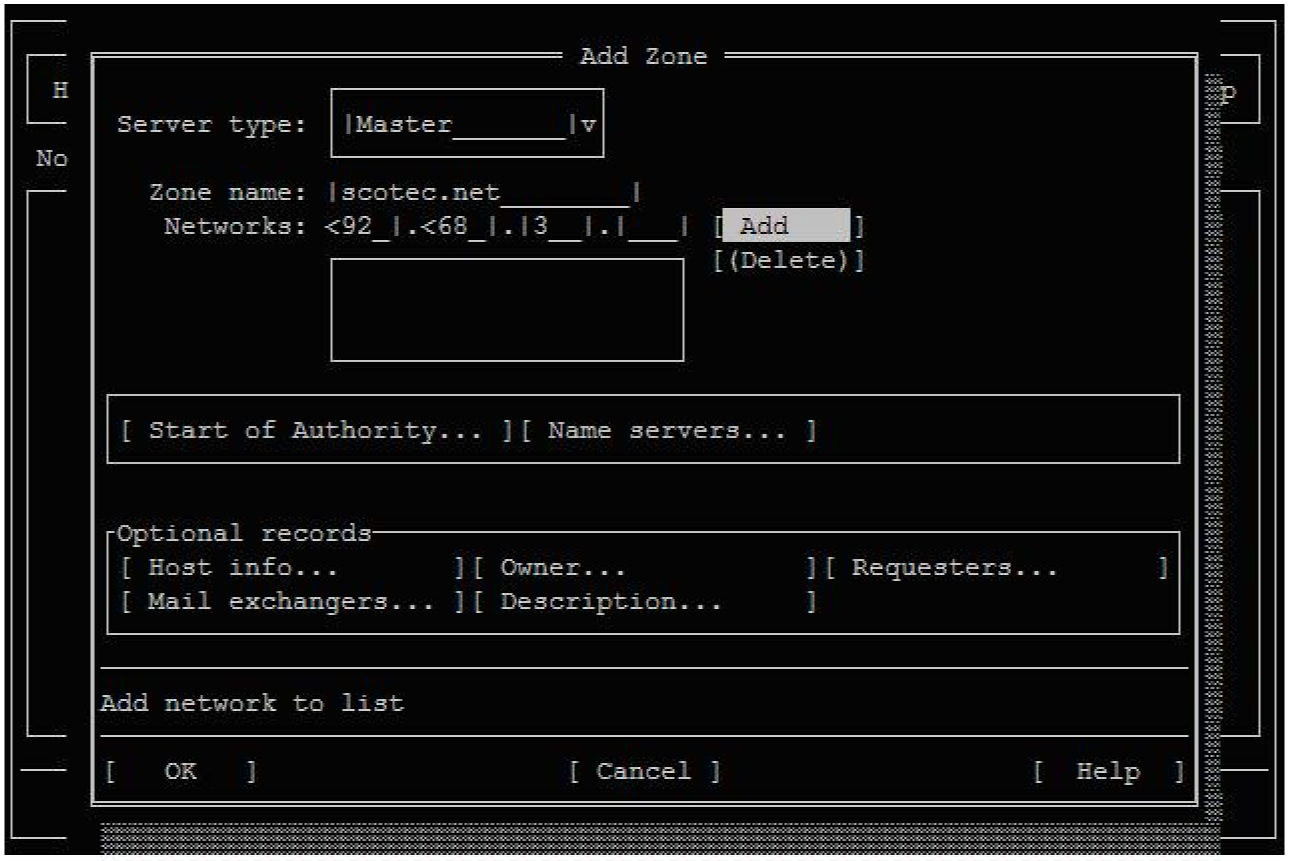
- Set the zone type



```
                              Add Zone
 Server type:  |Caching-only_|v|
  Zone name:  |                          |
   Networks:  |   Caching-only     ____| [(Add    )]
                 *Master                 [(Delete)]
                  Slave
                  Stub

 [(Start of Authority...)][(Name servers...)]

 Optional records
 [(Host info...       )][(Owner...          )][ Requesters...      ]
 [(Mail exchangers...)][(Description...    )]

 [   OK   ]                 [ Cancel ]                  [  Help  ]
```

Administering DNS with DNS Manager

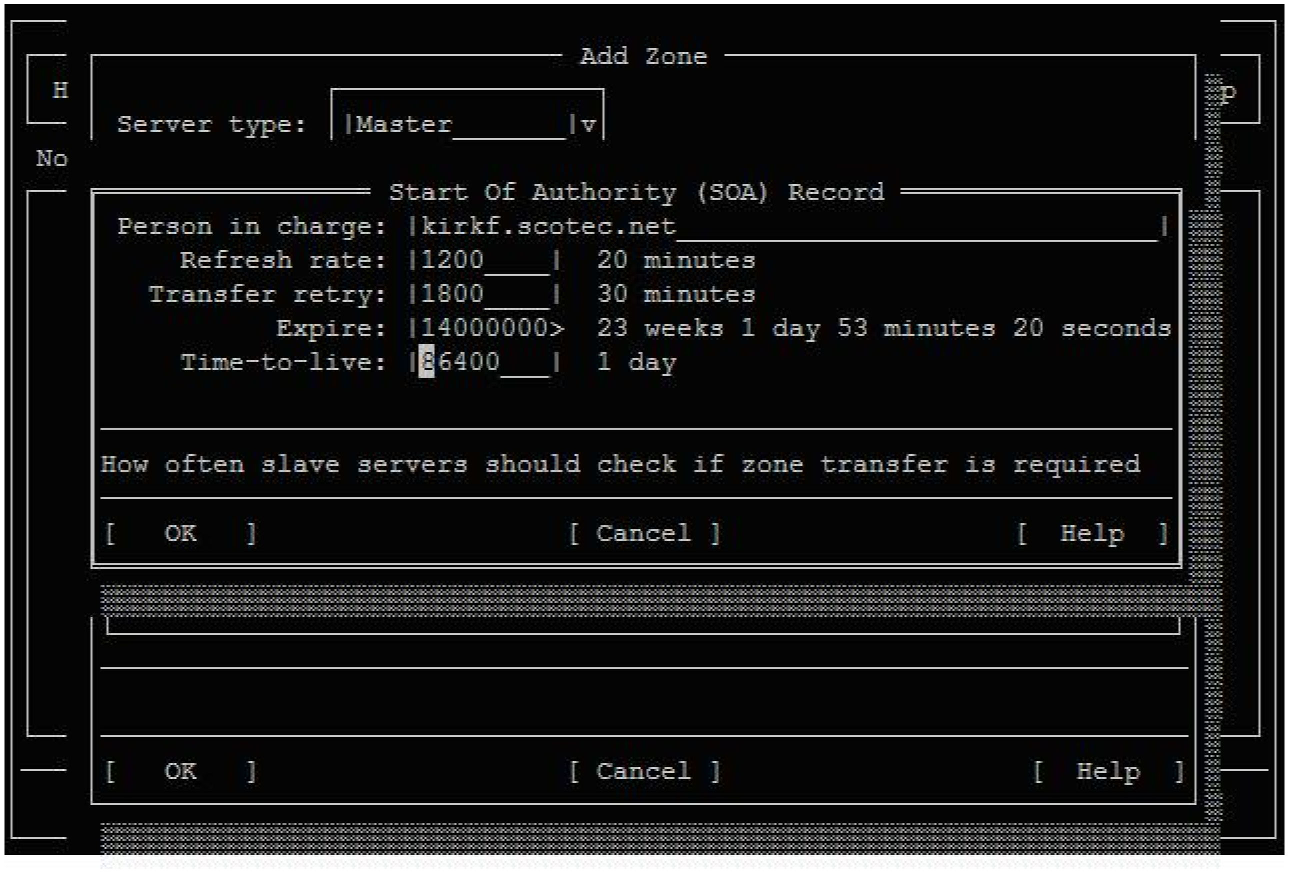
- Define the Domain and Network data



```
                         Add Zone
  Server type:  |Master_______|v

    Zone name: |scotec.net________|
     Networks: <92_|.<68_|.|3__|.|___|  [ Add      ]
                                        [(Delete)]

 [ Start of Authority... ][ Name servers... ]

 Optional records
 [ Host info...        ][ Owner...           ][ Requesters...      ]
 [ Mail exchangers... ][ Description...     ]

Add network to list

[   OK   ]                  [ Cancel ]                  [  Help  ]
```

Administering DNS with DNS Manager

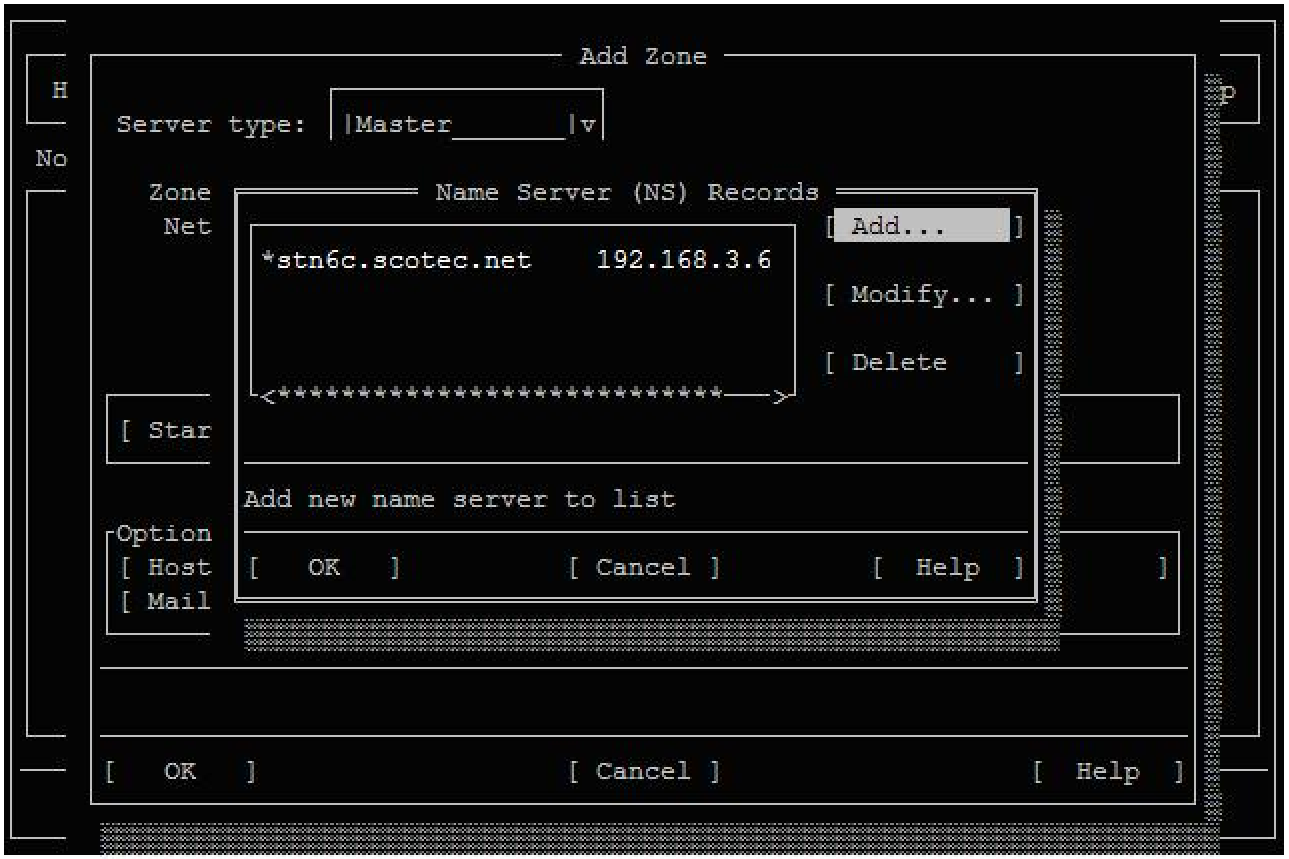
- Define the SOA data



```
                            Add Zone
 Server type:  |Master_______|v

          Start Of Authority (SOA) Record
 Person in charge: |kirkf.scotec.net______________________________|
     Refresh rate: |1200____|   20 minutes
   Transfer retry: |1800____|   30 minutes
           Expire: |14000000>   23 weeks 1 day 53 minutes 20 seconds
     Time-to-live: |86400___|   1 day

How often slave servers should check if zone transfer is required

[   OK   ]                    [ Cancel ]                  [  Help  ]

[   OK   ]                    [ Cancel ]                  [  Help  ]
```

Administering DNS with DNS Manager

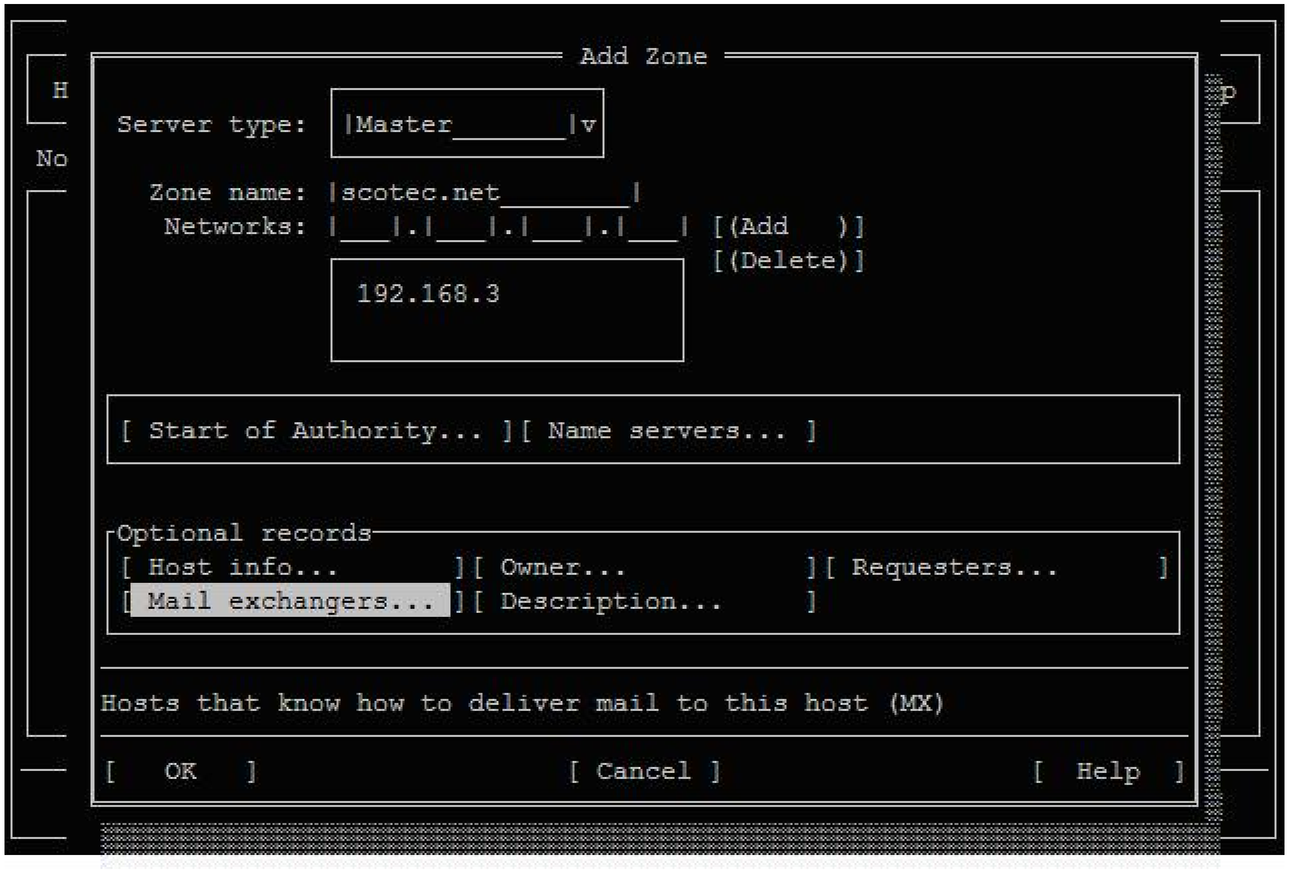
- Add nameserver records



```
                         Add Zone
 H
     Server type:  |Master________|v
 No
       Zone     Name Server (NS) Records
        Net                                  [ Add...     ]
             *stn6c.scotec.net    192.168.3.6
                                             [ Modify... ]

                                             [ Delete    ]
             <*****************************>
    [ Star

             Add new name server to list

    Option
    [ Host   [   OK   ]      [ Cancel ]       [  Help  ]
    [ Mail



  [   OK   ]                 [ Cancel ]                 [  Help  ]
```

Administering DNS with DNS Manager

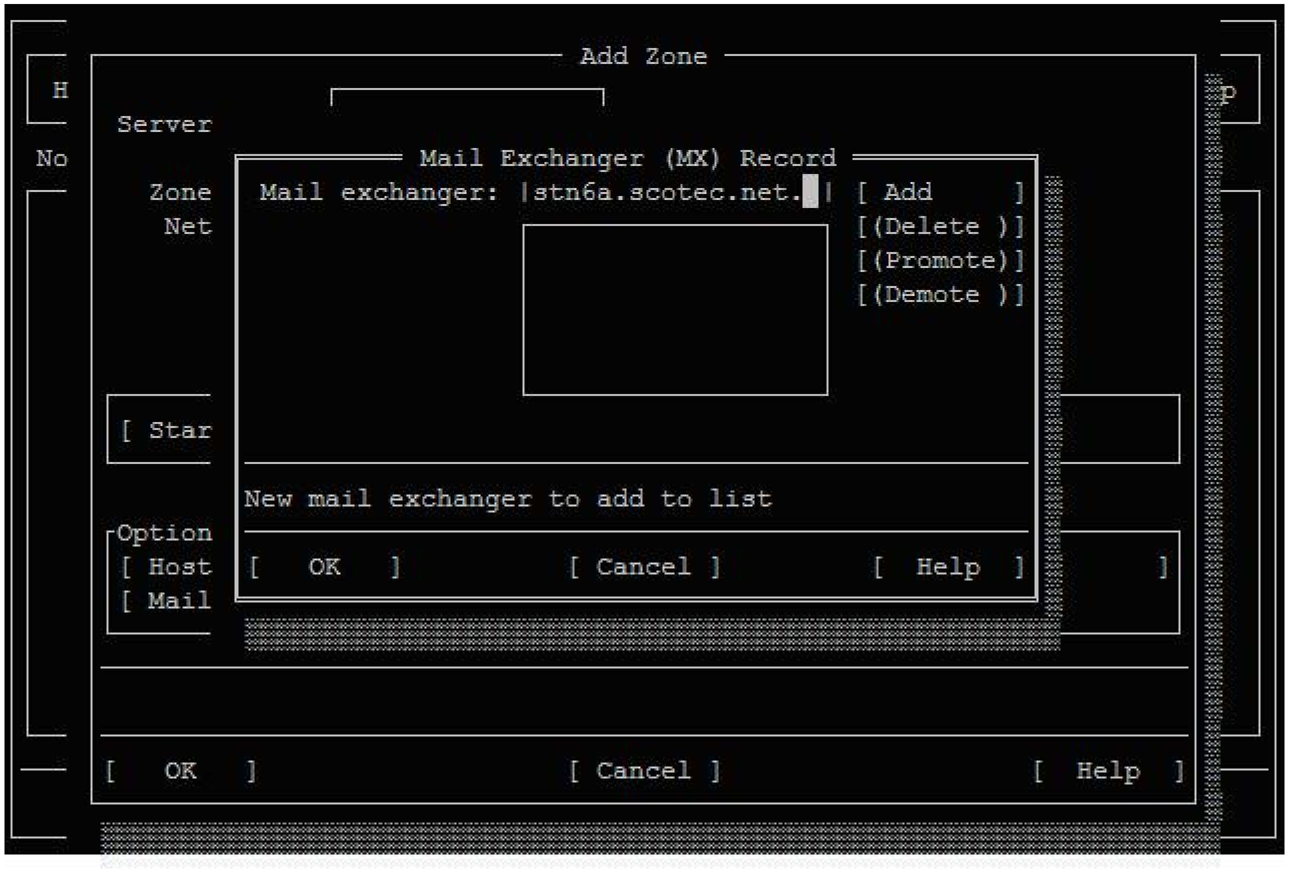
- Building our DNS with DNS Manager



```
                          Add Zone
 H
   Server type:  |Master_______|v
 No
     Zone name: |scotec.net________|
      Networks: |___|.|___|.|___|.|___|  [(Add    )]
                                          [(Delete)]
                 192.168.3

 [ Start of Authority... ][ Name servers... ]

 Optional records
 [ Host info...         ][ Owner...            ][ Requesters...      ]
 [ Mail exchangers...   ][ Description...      ]

Hosts that know how to deliver mail to this host (MX)

[   OK   ]                  [ Cancel ]                    [  Help  ]
```

Administering DNS with DNS Manager

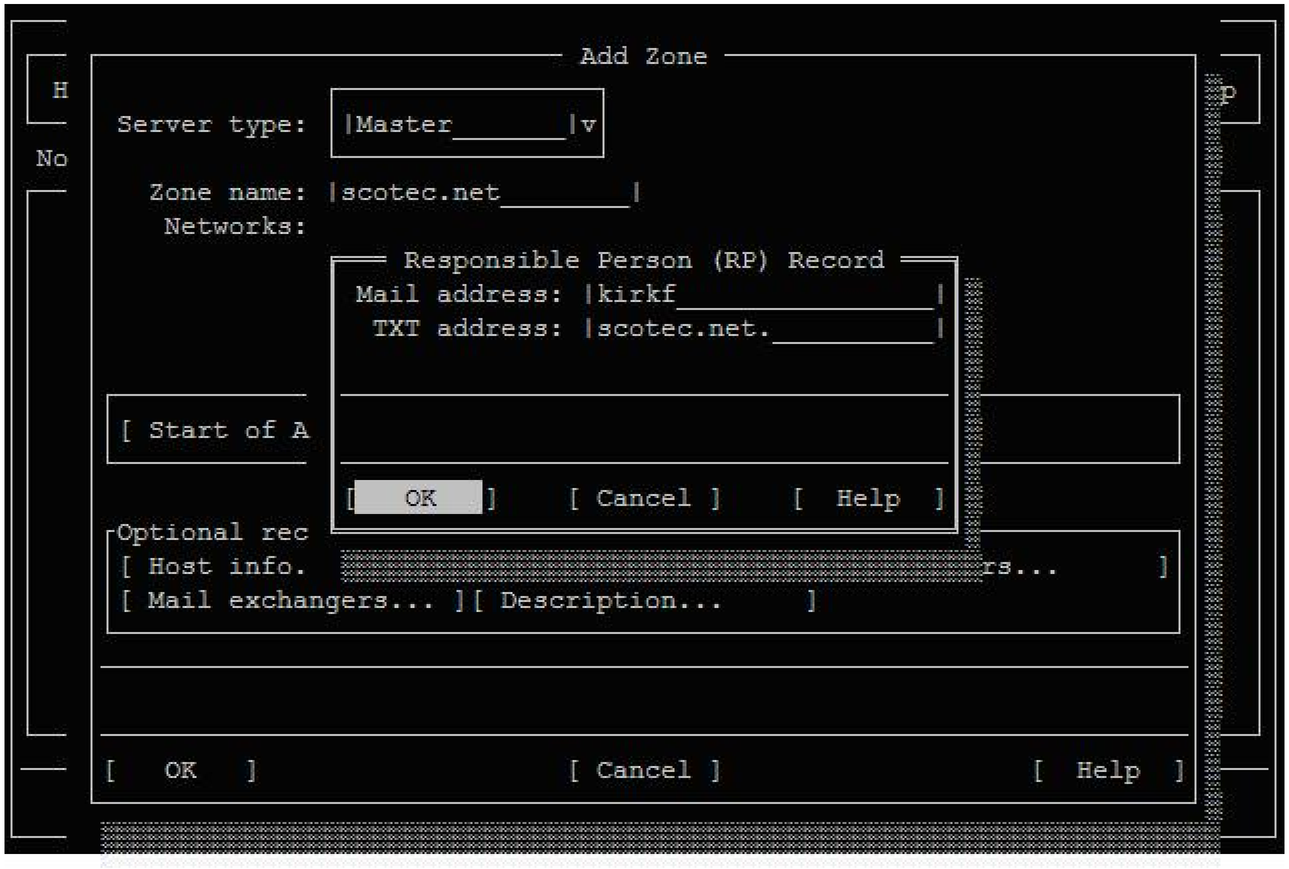
- Add domain MX records



```
                              Add Zone
 H                                                              p
     Server
 No               Mail Exchanger (MX) Record
       Zone    Mail exchanger: |stn6a.scotec.net. |  [ Add      ]
        Net                                          [(Delete )]
                                                     [(Promote)]
                                                     [(Demote )]

     [ Star

              New mail exchanger to add to list

     Option
     [ Host   [   OK   ]          [ Cancel ]          [  Help  ]          ]
     [ Mail

     [   OK   ]                   [ Cancel ]                 [  Help  ]
```

Administering DNS with DNS Manager

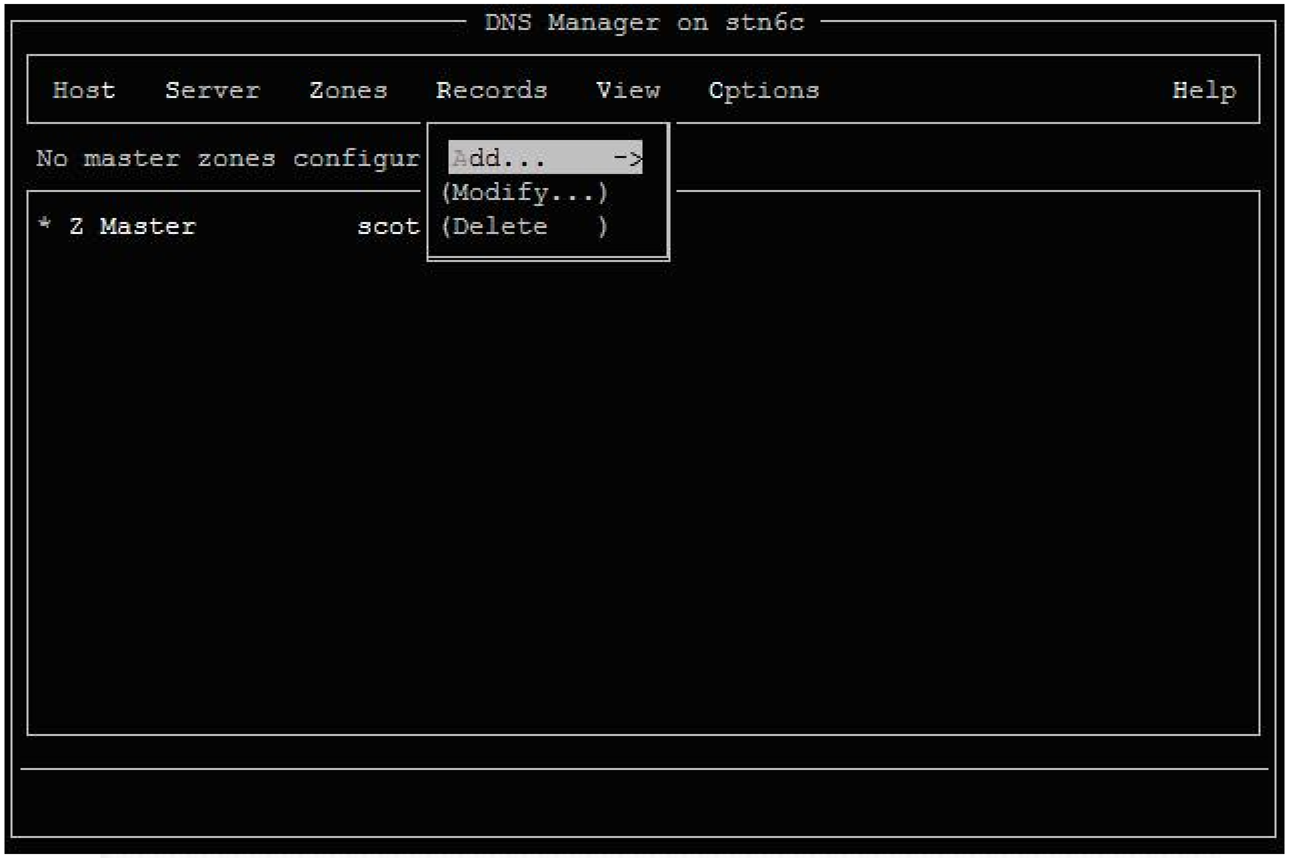
- Add a Domain contact



```
                         Add Zone
 H
   Server type:  |Master_______|v
 No
     Zone name: |scotec.net________|
      Networks:
                ═══ Responsible Person (RP) Record ═══
                  Mail address: |kirkf_______________|
                   TXT address: |scotec.net.__________|

   [ Start of A

                 [   OK    ]    [ Cancel ]    [  Help  ]
   Optional rec
   [ Host info.                                    rs...      ]
   [ Mail exchangers... ][ Description...     ]

   [   OK   ]                  [ Cancel ]              [  Help  ]
```

Administering DNS with DNS Manager

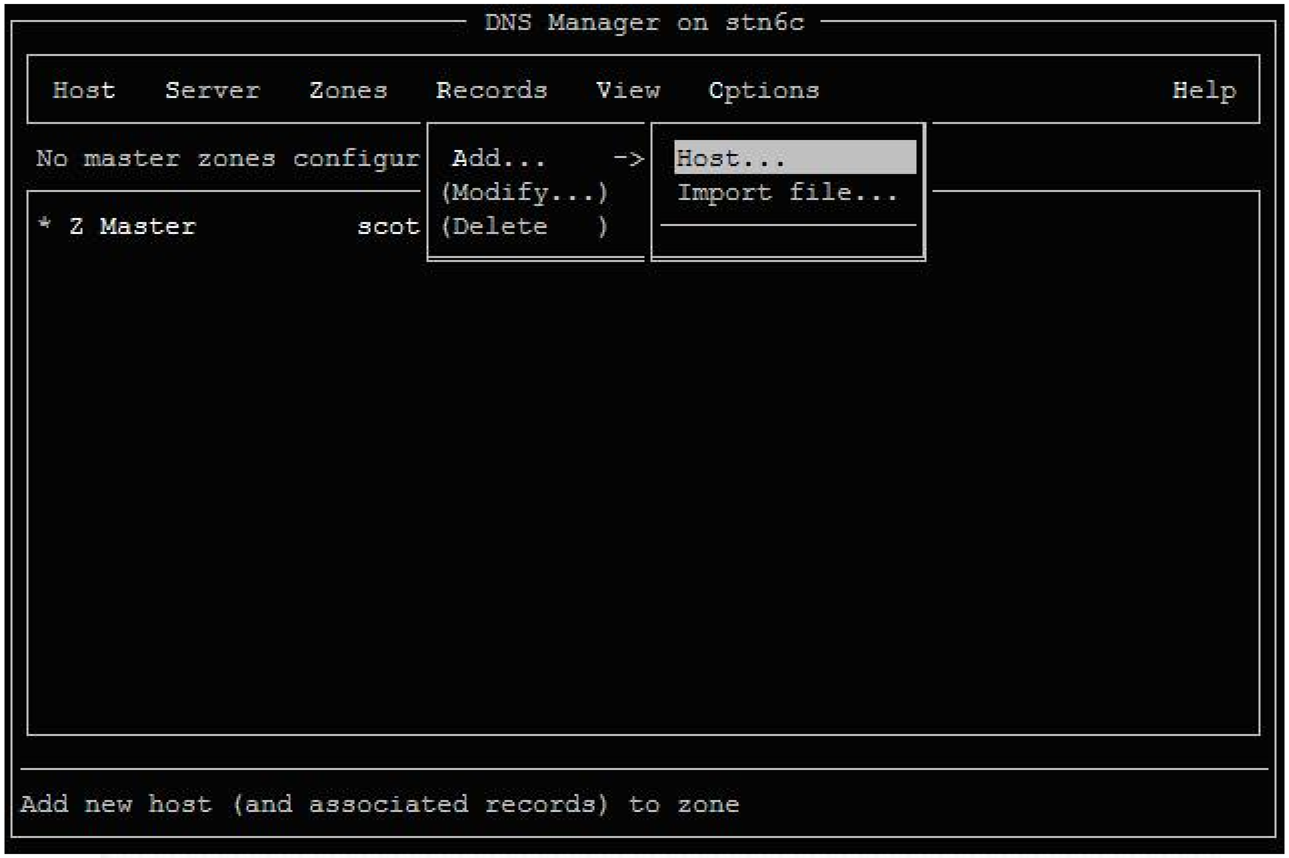
- Add records



```
                         DNS Manager on stn6c
 Host   Server   Zones   Records   View   Options                    Help
No master zones configur  Add...      ->
                          (Modify...)
* Z Master           scot (Delete    )
```

Administering DNS with DNS Manager

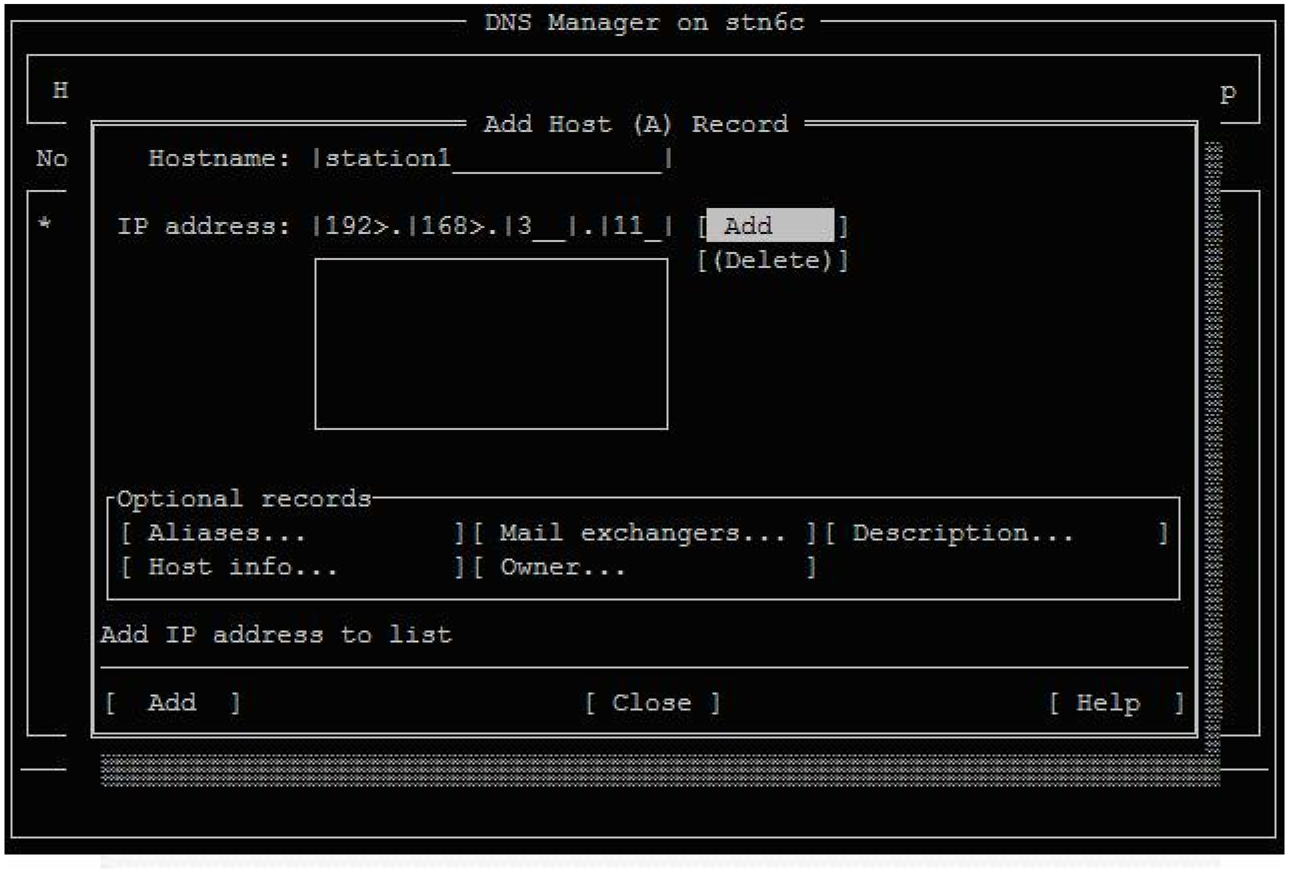
- You can add host records or import from a hosts format file



```
                        DNS Manager on stn6c
 Host    Server    Zones    Records    View    Options              Help
No master zones configur  Add...      ->  Host...
                         (Modify...)      Import file...
* Z Master          scot (Delete    )

Add new host (and associated records) to zone
```

Administering DNS with DNS Manager

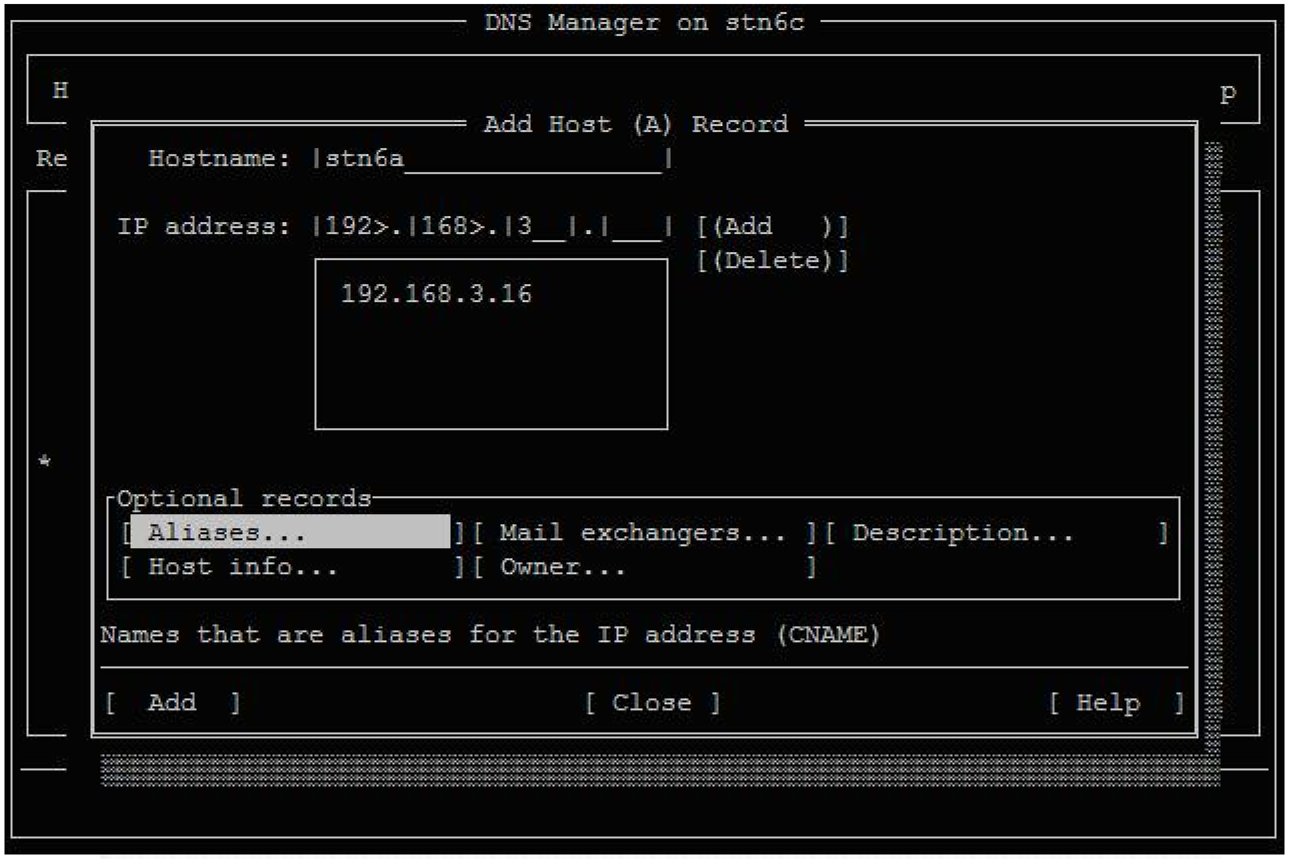
▪Add the hosts addresses and name



```
                     DNS Manager on stn6c
 H                                                              p
          Add Host (A) Record
No   Hostname: |station1______________|

*  IP address: |192>.|168>.|3__|.|11_| [ Add    ]
                                       [(Delete)]

   Optional records
   [ Aliases...        ][ Mail exchangers... ][ Description...    ]
   [ Host info...      ][ Owner...           ]

  Add IP address to list

  [  Add  ]                 [ Close ]                  [ Help  ]
```

Administering DNS with DNS Manager

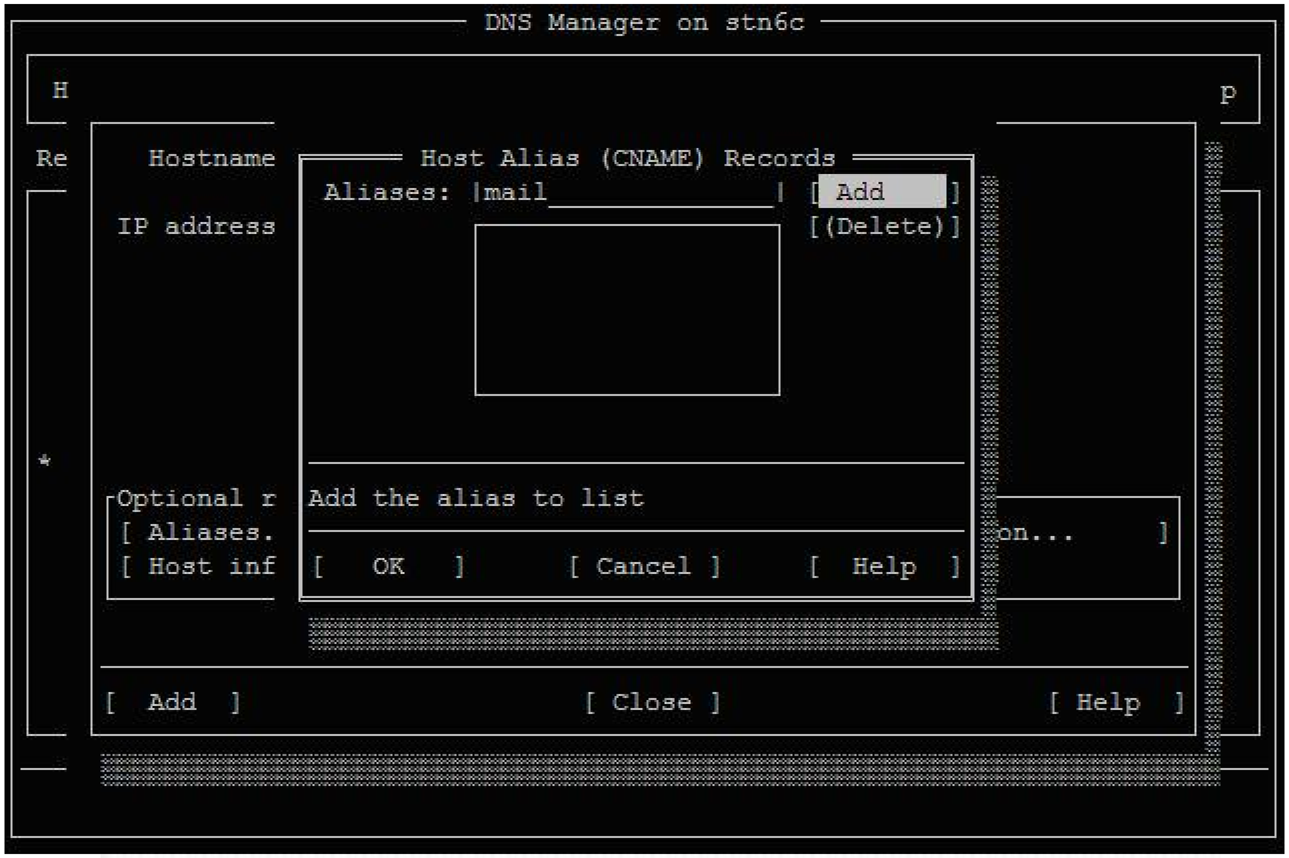
- You can add additional resource records for this host



```
                         DNS Manager on stn6c

 H                                                                    p
                          Add Host (A) Record
 Re      Hostname: |stn6a                |

       IP address: |192>.|168>.|3__|.|___| [(Add    )]
                                            [(Delete)]
                     192.168.3.16


 *
        Optional records
        [ Aliases...        ][ Mail exchangers... ][ Description...      ]
        [ Host info...      ][ Owner...           ]

      Names that are aliases for the IP address (CNAME)

      [  Add  ]                  [ Close ]                   [ Help  ]
```

Administering DNS with DNS Manager

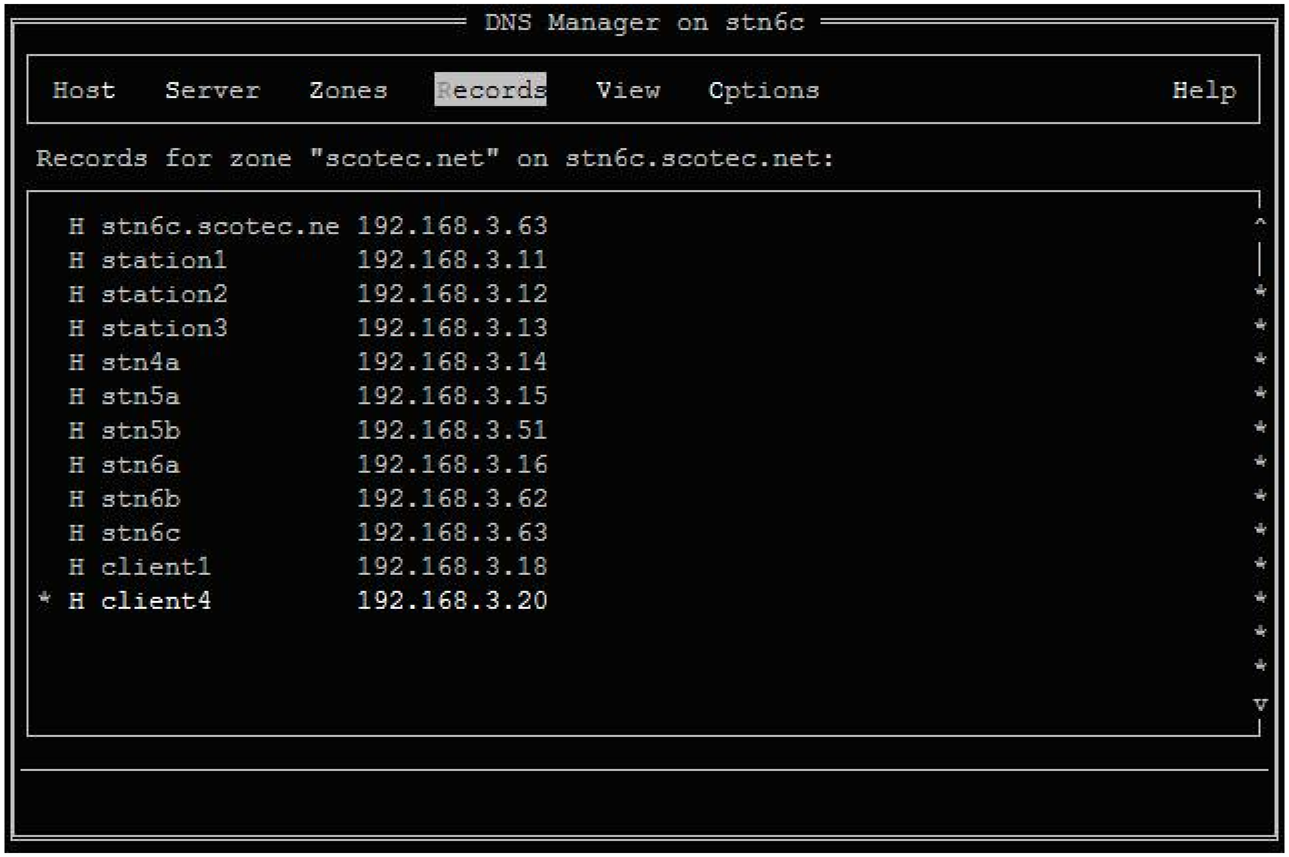
- Add any CNAME records for this host



```
─────────────────── DNS Manager on stn6c ───────────────────
 H                                                        p
 Re    Hostname  ═══ Host Alias (CNAME) Records ═══
                  Aliases: |mail_____________|  [ Add    ]
       IP address                               [(Delete)]

 *
      Optional r  Add the alias to list
      [ Aliases.                                       on...      ]
      [ Host inf  [   OK   ]     [ Cancel ]     [  Help  ]

      [  Add  ]              [ Close ]               [ Help  ]
```

Administering DNS with DNS Manager

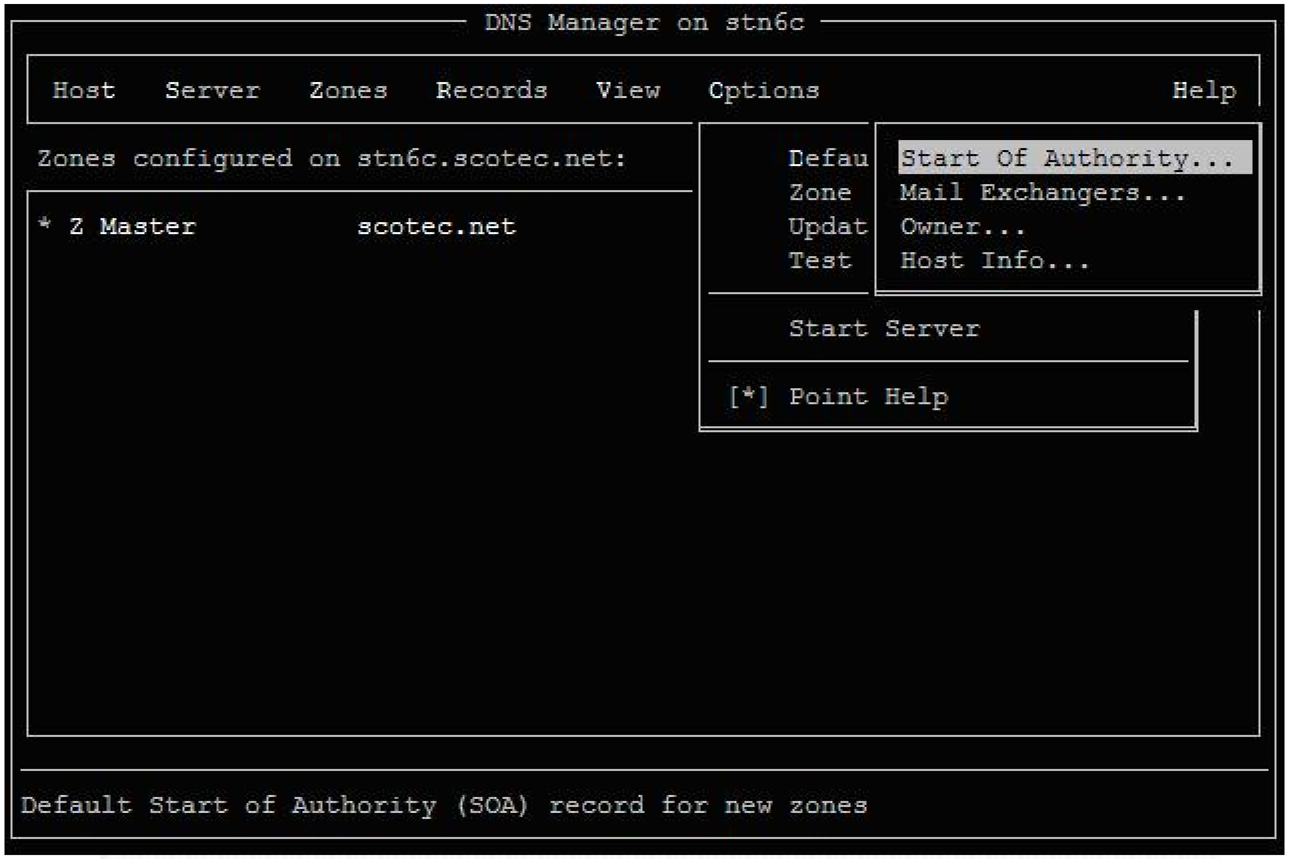
- The resulting table of hosts is available in the View menu



```
Records for zone "scotec.net" on stn6c.scotec.net:

  H stn6c.scotec.ne 192.168.3.63
  H station1        192.168.3.11
  H station2        192.168.3.12
  H station3        192.168.3.13
  H stn4a           192.168.3.14
  H stn5a           192.168.3.15
  H stn5b           192.168.3.51
  H stn6a           192.168.3.16
  H stn6b           192.168.3.62
  H stn6c           192.168.3.63
  H client1         192.168.3.18
* H client4         192.168.3.20
```

Administering DNS with DNS Manager

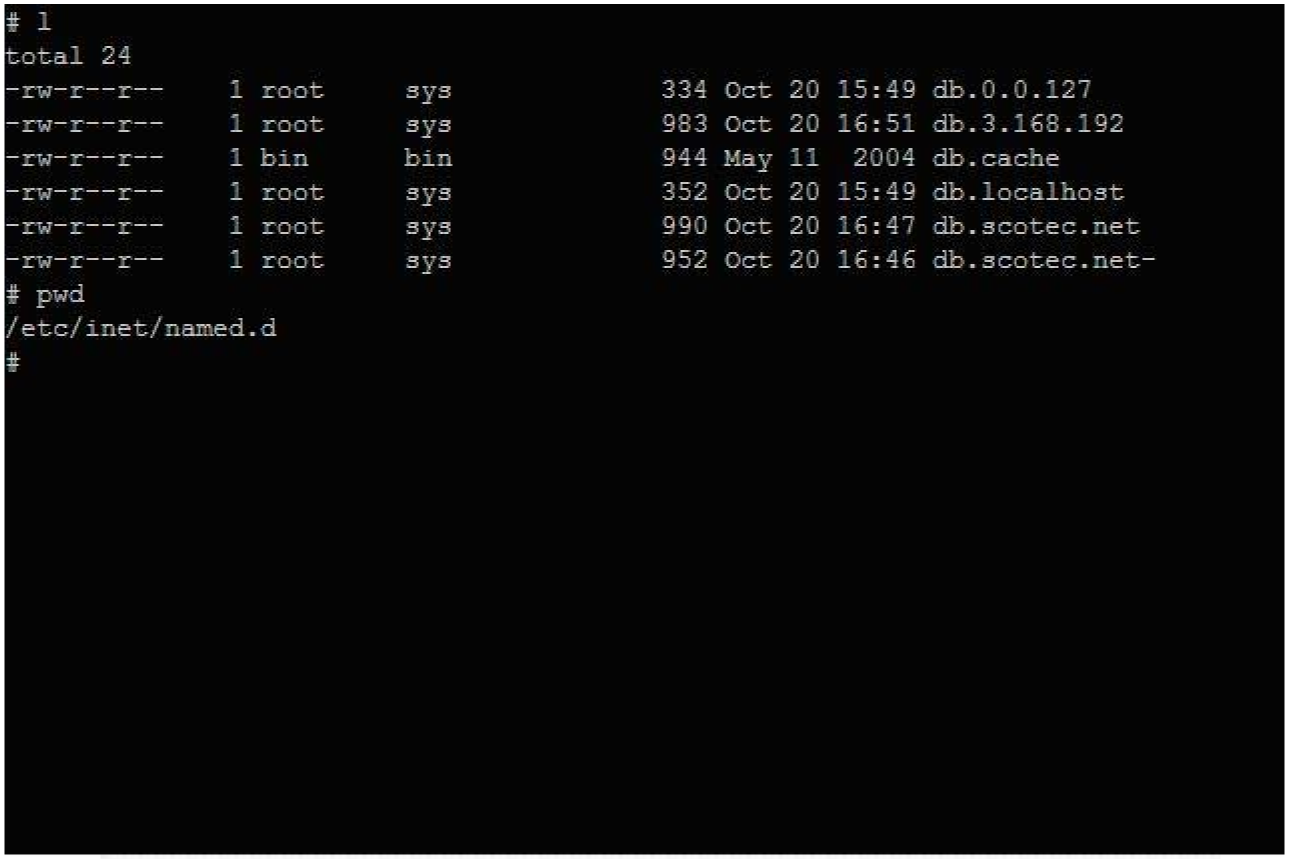
- You can also set server wide defaults before creating zone data



```
                         DNS Manager on stn6c

 Host   Server   Zones   Records   View   Options                    Help

Zones configured on stn6c.scotec.net:           Defau  Start Of Authority...
                                                Zone   Mail Exchangers...
* Z Master          scotec.net                  Updat  Owner...
                                                Test   Host Info...

                                                Start Server

                                            [*] Point Help

Default Start of Authority (SOA) record for new zones
```

Administering DNS with DNS Manager

- Start the server and the zone files are created in /etc/inet/named.d, or /etc/named.d on OSR6



```
# l
total 24
-rw-r--r--    1 root     sys            334 Oct 20 15:49 db.0.0.127
-rw-r--r--    1 root     sys            983 Oct 20 16:51 db.3.168.192
-rw-r--r--    1 bin      bin            944 May 11  2004 db.cache
-rw-r--r--    1 root     sys            352 Oct 20 15:49 db.localhost
-rw-r--r--    1 root     sys            990 Oct 20 16:47 db.scotec.net
-rw-r--r--    1 root     sys            952 Oct 20 16:46 db.scotec.net-
# pwd
/etc/inet/named.d
#
```

Name Service Clients

- **Name service clients**
- A machine running client software can be configured to handle resolution using DNS, */etc/hosts*, or NIS, or some combination of these three methods in a specified order.
- If DNS is used, the IP addresses of the DNS name servers are configured in the */etc/resolv.conf* file.
- A client can nominate itself (using the *localhost* or loopback address, 127.0.0.1) as a name server if it is configured as a DNS name server.
- The minimal configuration of the */etc.resolv.conf* file to allow a host to perform a DNS look up on itself is:

```
nameserver 127.0.0.1
```

Name Service Clients

- Remote DNS name service only should be configured if a computer has limited memory or CPU power.
  - In this case, all queries will be answered by a name server running on another computer on the network.
  - You will obtain better performance by configuring the local machine as a caching−only name server.
  - The resolver invoked by client programs does not cache replies because these programs tend to be short lived.
  - Because a name server caches replies, all clients can take advantage of the data in its cache.
  - Having a local name server avoids network traffic and propagation delays on subsequent queries to resolve the same IP addresses or host names.

The resolv.conf File

- The Resolver is configured in /etc/resolv.conf
- The are 6 possible directives
  - domain
  - hostresorder
  - search
  - nameserver
  - sortList
  - options
- If resolv.conf exists and points to at least one NameServer, DNS is used to resolve addresses
  - Otherwise hosts is used (or possibly NIS)

Configuring Your Resolver - Unix

The NameServer directive

```
nameserver 222.33.4.2
nameserver 222.33.4.7
```

- Each NameServer should be listed on a separate line
- By default you can specify a maximum of 3 NameServers
  - This is controlled by the kernel tunable MAXNS
- If you only have one NameServer,
  - the resolver will query it with a 5 second timeout
  - If it receives an ICMP port or host unreachable error, it doubles the 5 seconds and tries again
  - It will retry 4 times, 5 sec., 10 sec, 20 sec. for a total of 35 seconds then fall back to hosts
  - If it times out on even one query, it returns a null response and does not fall back. It will never fall back to hosts!!

Configuring Your Resolver - Unix

- OSR6 resolver issues
  - The hostresorder directive is a legacy directive from OpenServer 5 that is only recognised by OSR5 ABI binaries and not by SVR5 Binaries
  - Determine the type of application you want to use

    file /bin/ping returns

    /bin/ping: ELF 32-bit LSB executable 80386, dynamically linked, stripped, no debug (OSR5 ABI)
  - This indicates that the ping command will use the hostresorder directive.

    file /usr/lib/mozilla/mozilla-bin returns

    /usr/lib/mozilla/mozilla-bin: ELF 32-bit LSB executable 80386, dynamically linked, stripped, no debug (SVR5 ABI)

Configuring Your Resolver - Unix

- This is why some utilities will fail during an initial system build
  - Utilities such as brand, file manager, custom have components which try to resolve names
  - Some of these components are SVR5, i.e. fstyp
  - Setting resolv.conf before networking is configured or before there is a working DNS will stop these utilities working

Configuring Your Resolver - Unix

- Resolver will follow the behaviour set in /etc/netconfig
- There are 3 resolver libraries

```
/usr/lib/tcpip.so       (uses local /etc/hosts, /etc/services, etc.)
/usr/lib/resolv.so      (uses DNS)
/usr/lib/tcpip_nis.so   (uses NIS)
```

- The order of search is specified in /etc/netconfig in the line

```
tcp tpi_cots_ord v inet tcp /dev/tcp /usr/lib/tcpip.so,/usr/lib/resolv.so
```

- This says the tcp service should use hosts first then DNS
- If you need SVR5 ABI applications to use hosts before DNS you must modify the order in which libraries are called

The Resolver Libraries

- *tcpip.so*
  - contains the */etc/hosts* name−to−address mapping routines for the TCP/IP protocol suite
- *resolv.so*
  - contains the Domain Name Service (DNS) name−to−address mapping for the TCP/IP protocol suite
- *straddr.so*
  - contains the name−to−address mapping routines for any protocol that accepts strings as addresses.
  - The loopback driver is an example.

The tcpip.so library

- The routines in this dynamic library create addresses from the **hosts** TCP/IP package.
- The *hosts* file contains the machine's IP address as the first field followed by any number of machine names separated by white space. For example:

```
#IP address machine name(s) (optional comment)
#
192.11.108.01 bilbo
192.11.108.16 elvis # Located at Graceland
192.11.103.2 weeble wombat
```

- The *services* file contains three fields: ``service name'', ``port/protocol'' (with one of two protocol
- specifications either ``tcp'' or ``udp''), and ``aliases''. For example:

```
#service name port/protocol aliases
#
netstat 15/tcp
netstat 15/udp
time 37/tcp mail
time 37/udp mail
nntp 119/tcp usenet readnews untp
```

The tcpip.so library

- For an application to use this library to request the address of a service on a particular host, the host name must appear in the */etc/hosts* file and the service name must appear in the */etc/services* file.
- If one or the other does not appear, an error will be returned by the name−to−address mapping routines.

The Resolv.so Library

- The routines in this dynamic library create addresses similar to the *tcpip.so* file, except that it uses Domain Name Service instead of */etc/hosts* to provide similar features.

Troubleshooting DNS

- After configuration, DNS starts automatically if */etc/inet/named.conf* is present. When **named** starts up, it writes its process id to the file */etc/inet/named.pid*. This is useful to programs that want to send signals to **named**.
- If you encounter problems with **named**, first view the logfile */var/adm/log/osmlog* for any errors. If none are found, use the **ndc** command to send commands to **named**.
- This allows you to troubleshoot DNS without restarting the **named** process.
- Useful commands are:
- ***ndc dumpdb***
  - Dumps the current database and cache to */var/tmp/named_dump.db*. This gives you an indication to whether the database was loaded correctly.
- ***ndc /debug***
  - Toggles debugging on or off. Each following invocation of this command toggles debugging on or off. The output goes to */var/tmp/named.run*.

- ***ndc reload***
- Causes **named** to read *named.conf* and reload the database, overwriting cached data. This is useful
- when you make a change to a data file and you want **named**'s internal database to reflect the change.
- This command also has the effect of scheduling all slave servers for serial−number checks, which
- could lead to zone transfers ahead of the usual schedule. Normally, serial−number comparisons are
- done only at the intervals specified in the zone's SOA record.
- ***ndc reload zone1** ...*
- Reloads the database for the specified zones.

Troubleshooting DNS

- ***ndc trace***
- Increases the tracing level of all incoming queries. The trace is sent to */var/adm/log/osmlog* and
- provides a large amount of data.
- ***ndc /trace***
- Toggles tracing on or off.
- ***ndc notrace***
- Turns off tracing.
- You can also use **ndc** to control **named** daemons running on remote name servers provided that they have
- been updated to at least version 8.2.1 of BIND and that they have been configured to allow remote control.
- See the **ndc**(1Mtcp) manual page for more information.

Troubleshooting DNS

- **Using nslookup, dig, and host**
- **nslookup**, **dig**, and **host** are useful commands that allow you to perform DNS queries, and to test out your DNS configuration

Troubleshooting DNS

- **nslookup interactive commands**
- These sample commands are available from the **nslookup** shell:

  volga
- Return the IP address of *volga*.

  172.16.118.1
- Return the name matching the IP address you enter.

  set querytype=ns
- Set the query type to the Name Server record. Future queries of names and IP addresses return the NS record from that host.

  set querytype=a
- Restore the query type to the Address record.

  server server
- Make ***server*** the default server that is queried.

  nslookup interactive options

Troubleshooting DNS

- Here are the commonly used options of **nslookup**. For a complete list, see the manual page for **nslookup**.

  [no]recurse
- Sets the query type to recursive. When toggled to norecurse, **nslookup** performs iterative queries.

  querytype=type
- Sets the query type to the DNS data type specified. Common types include **a** (Address), **any** (any datatype), **mx** (Mail Exchanger), and **ns** (Name Server).

  retry=n
- Resends the query ***n*** times before giving up.

  root=root server
- Sets the root server to the server you enter.

  timeout=n
- The period of time **nslookup** waits for a response after the query is sent. This period doubles between each retry.
- You can save any of these options in a ***.nslookuprc*** file in your home directory. The format of this file, which is searched for each time you invoke **nslookup**, is one **set** command per line.

Troubleshooting DNS

- **Examples of using the dig command**
  - Obtain the latest list of root domain servers:

    dig . ns
  - Find out the name servers for a zone:

    dig @*server domain* ns
  - Request all records for a zone from an authoritative server:

    dig @*server domain* axfr
  - Look up the domain name corresponding to the IP address 172.16.118.1:

    dig −x 172.16.118.1
- **Examples of using the host command**
  - Use **host** to find all the host records for a zone:

    host −l *domain*
  - Use **host** to request all the records for a zone:

    host −l −v −t any *domain*
  - **NOTE:** These commands require a zone transfer which the server may disallow.

Content

Questions