OpenBSD as a domain name server

Author: Daniele Mazzocchio
Last update: May 13, 2007
Latest version: http://www.kernel-panic.it/openbsd/dns/

Table of Contents

1. Introduction

So our network is growing rapidly, with our fresh new redundant firewalls, mail server, proxy cache and so on. Now our mind is filled up with IP addresses and our fingers are getting tired of typing all those numbers and dots. It's definitely time to set up a domain name server! The following is the list of the pieces of software we will use:

OpenBSD
: the secure by default operating system, with "*only two remote holes in the default install, in more than 10 years!*";

Bind (Berkeley Internet Name Daemon)
: an "*openly redistributable reference implementation of the major components of the Domain Name System*".

OpenBSD is certainly a well-suited platform for running a domain name server: first and foremost, the default install always includes the latest (patched) release of Bind, saving us the bother of compiling and installing it; secondly, OpenBSD is renowned for its security, and domain name server security is at the base of the whole network security; lastly, OpenBSD is very stable, reliable, fast and easy-to-administer ...just how a domain name server should be!

Anyway, most of the topics we will cover aren't OpenBSD-specific: Bind supports several platforms, thus making its configuration easy to port acrossdifferent operating systems with minimal changes.

Bind is very powerful, flexible and feature-rich, and this can sometimes turn its configuration into a tricky task. Therefore, we will proceed step-by-step, starting with a very basic configuration and then building upon it, gradually introducing the most interesting and popular features of Bind. We will address common issues such as redundancy, security and DHCP and NAT handling.

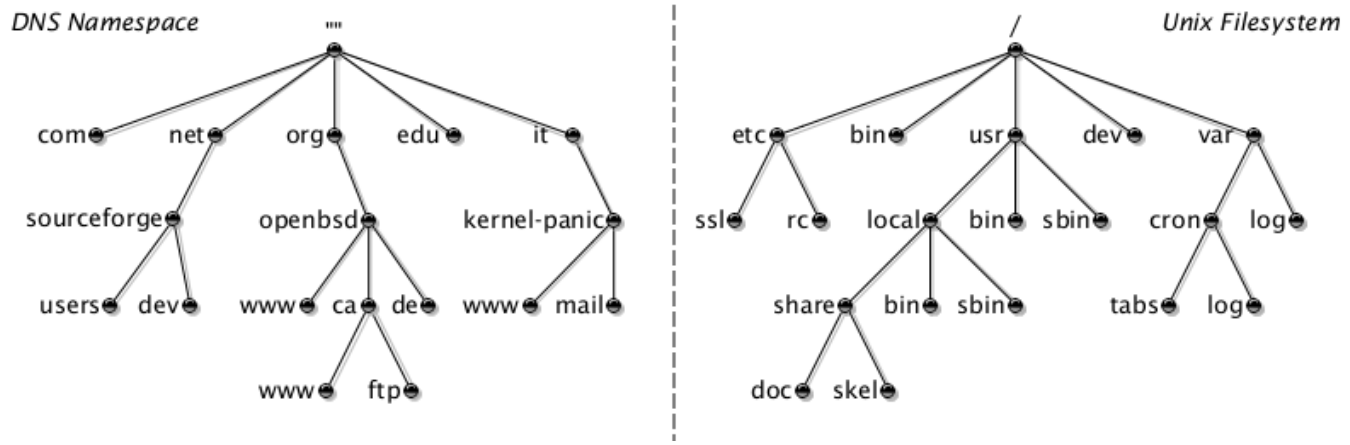
A basic working knowledge of OpenBSD is assumed, since we won't dwell on the installation and base configuration of the operating system.

2. The Domain Name System

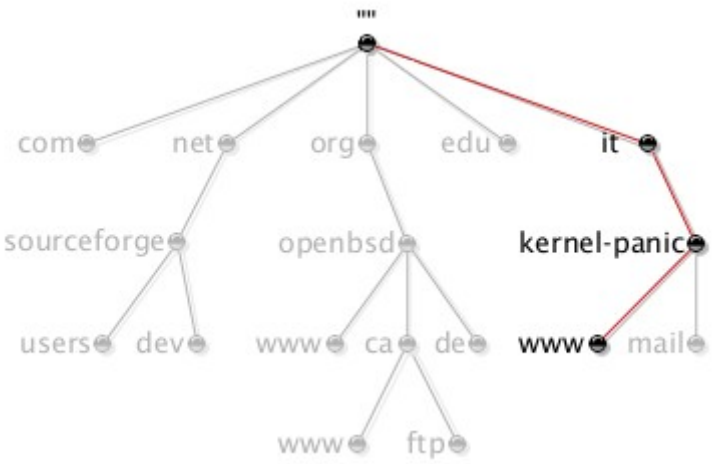
DNS configuration is much easier if you have a good understanding of its fundamentals. Hence, before hurrying to edit our zone data files, let's take a brief look at the overall structure of the Domain Name System and its inner mechanisms.

2.1 A few definitions

The Domain Name System is a distributed database of resource records (see [RFC1034]), associating many types of information (e.g. IP address, mail exchanger, etc.) with domain names. Similarly to the Unix file system, the structure of this database is a hierarchical inverted tree, with the root at the top. The whole tree is called the Domain Name Space.



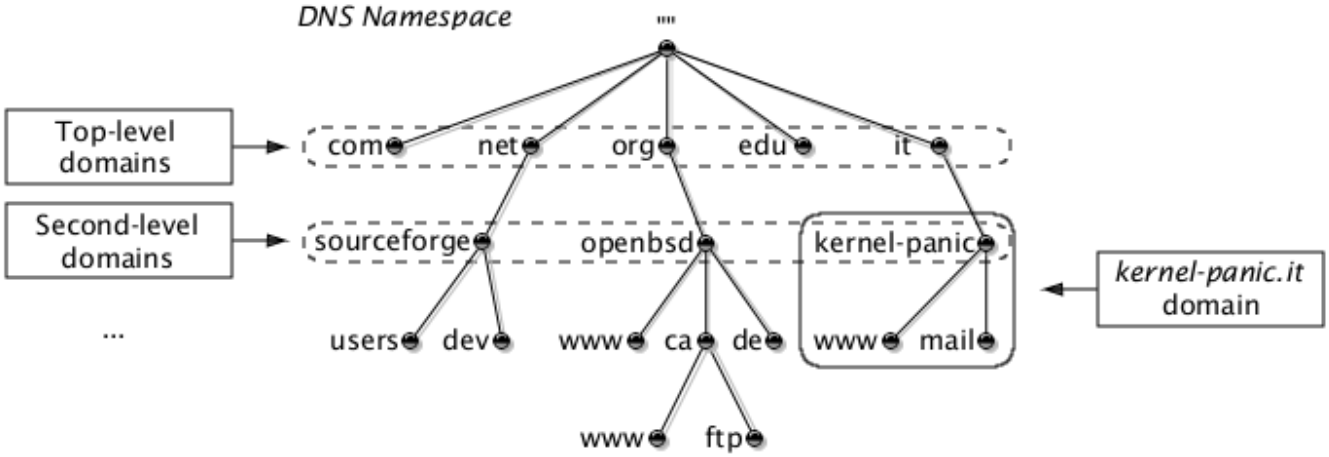
Each node in the Domain Name Space has a text label (the root node has a special zero-length label, "") and is uniquely identified by its domain name, i.e. the list of the labels on the path from the node to the root, separated by dots (Unix paths, on the contrary, start from the root and are separated by slashes).



For instance, the domain name highlighted in the above picture is made up of the sequence "www", "kernel-panic", "it" and the root's null label, and is therefore written as "www.kernel-panic.it.". Since the root node is usually written as a single dot, domain names ending with a trailing dot are considered absolute (similarly to Unix absolute pathnames, starting with a leading slash). An absolute domain name is also referred to as a fully qualified domain name (FQDN). Domain names with no trailing dot are considered relative to another domain, usually to the root itself. A relative domain name is also referred to as a partially qualified domain name (PQDN).

A domain is a subtree of the domain name space and takes the domain name of its top node. Each domain may have its own subtrees, called subdomains. Domains may also be referred to by level: a top-level (or

first-level) domain is a child of the root; a second-level domain is a child of a first-level domain; and so on.



The hierarchical structure of the domain name system allows for the decentralization of its administration; in fact, an organization administering a domain can delegate, i.e. assign responsability for, a subdomain to a different organization and only maintain information about the non-delegated part of the domain (called a zone).

Programs that store information about a zone are called domain name servers and are said to have authority for that zone. There are two types of name servers:

- primary master name servers, which read the data for the zone from a local file (called zone data file);
- secondary master name servers (or slaves), which get data from another name server that is authoritative for the zone (called master server), through a zone transfer; usually, but not necessarily, the master server is the zone's primary master.

Having two types of name servers makes administration easier, by providing a single point of configuration, while allowing for redundancy, load sharing and responsiveness by having multiple authoritative name servers for a zone.

2.2 The name resolution process

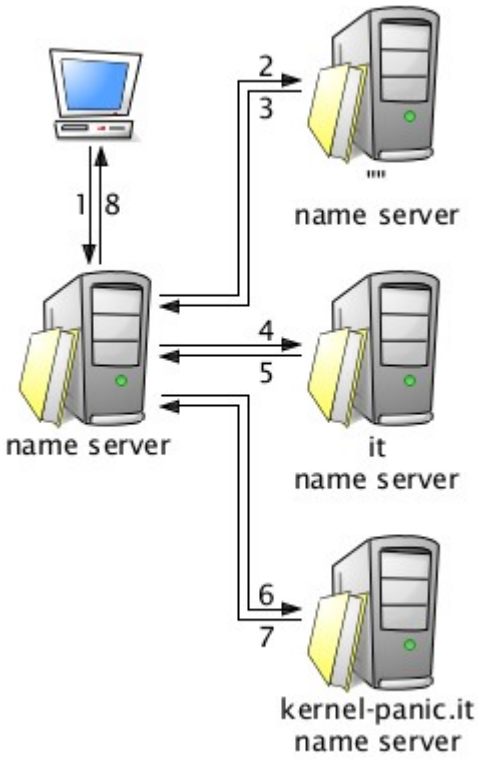
Clients that access name servers are called resolvers. In Bind, the resolver is just a library that must be linked by applications requiring name service. When an application needs information from the domain name space, it uses the resolver to perform a query against a DNS server (usually the corporate or the ISP's server). If authoritative for the queried zone, the DNS server will reply immediately; otherwise, it will search through the domain name space to find the requested data. This process is called name resolution.

There are two types of DNS queries:

- iterative (or nonrecursive), which simply ask a DNS server the best answer it *already* knows;
- recursive, which ask the DNS server to fully answer the query, or give an error.

Usually resolvers perform recursive queries, placing the burden of resolution on the queried name server; DNS servers, instead, perform sequences of iterative queries, following any referrals, until they receive the answer they are looking for.

Let's see how it all works by going through an example. Suppose you want to visit the "`www.kernel-panic.it`" web site; you type the URL in your browser, press "`Enter`" and this is what happens next:



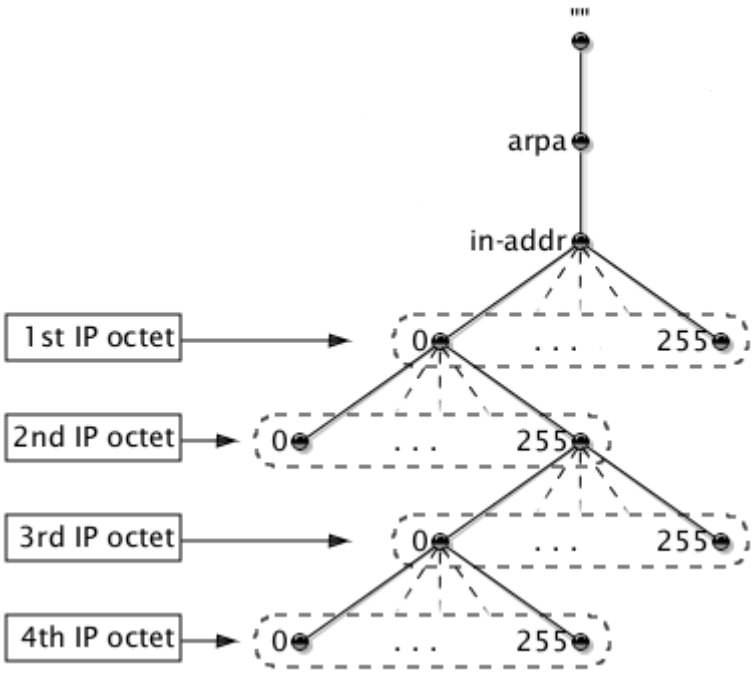
1. the resolver performs a recursive query against your corporate DNS server, expecting the IP address of the "`www.kernel-panic.it`" web server (or an error) in return;
2. since the corporate DNS server isn't authoritative for the queried zone, it will send an iterative query for the address of the "`www.kernel-panic.it`" domain name to a root name server, i.e. one of the 13 worldwide DNS servers which know the name servers authoritative for each of the top-level zones;
3. the queried root name server won't probably know the full answer, but it will certainly know which name servers are authoritative for the "`it`" zone. Therefore, it will refer your corporate DNS server to those name servers;
4. your DNS server will choose one of the referred name servers and send it the same iterative query for the "`www.kernel-panic.it`" domain name;
5. the queried "`it`" name server won't probably know the full answer and therefore will refer your corporate DNS server to the list of name servers authoritative for the "`kernel-panic.it`" zone;
6. your DNS server has finally discovered the authoritative name servers for the "`kernel-panic.it`" zone and will send the same query to one of them;
7. the queried name server will return the address of the "`www.kernel-panic.it`" domain name;
8. your corporate name server is finally able to return the information to the resolver.

As you can see, the resolution process may involve quite a few steps; but after each step, the name server learns a new piece of information about the domain name space. For instance, in the previous example, the corporate DNS server has learned which servers are authoritative for the "`it`" and "`kernel-panic.it`" zones. So what happens now if you want to connect to the "`ftp.kernel-panic.it`" machine? Your corporate name server already knows the authoritative servers for the "`kernel-panic.it`" zone; therefore it will send the query directly to one of them and get the answer in a single step, thus speeding up the resolution process. Storing learned data for future reference is called caching. Since version 4.9, Bind also keeps track of non-existing domains (negative caching), thus preventing the repeating of failed queries.

2.3 Reverse name resolution

Reverse name resolution is the process of mapping an IP address back to a FQDN. Though this may seem to require an exhaustive search of the whole domain name space, it is, in matter of fact, as simple as name resolution because the developers of DNS have created a special "`in-addr.arpa`" domain that uses the dotted-octed representation of IP addresses as labels.

In other words, the `in-addr.arpa` domain has (or could have, to be more precise) up to 256 third-level subdomains (numbered from 0 to 255), corresponding to the possible values in the first octet of an IP address; each of those 256 subdomains could have, in turn, up to 256 fourth-level subdomains, also numbered from 0 to 255, corresponding to the values of the second octet, and so on.



Therefore, to look up the FQDN associated with an IP address, the resolver simply has to query the name server for the `PTR` record (see below) of the corresponding node in the `in-addr.arpa` domain. For example, to get the domain name for the `62.149.140.23` IP address, the resolver will query the DNS server for the `PTR` record of the "`23.140.149.62.in-addr.arpa`" domain name.

As you can see, IP addresses appear reversed in the `in-addr.arpa` domain name. This is due to a basic difference between IP addresses and domain names: IP addresses get more specific from left to right, while domain names get more specific from right to left. Hence, naming nodes in the `in-addr.arpa` domain in this (seemingly odd) way actually allows IP addresses to correctly reflect the hierarchical structure of the domain name system.

2.4 Resource records

Each node in the domain name space has a set of resource information (which may be empty) associated to it, composed of separate resource records (RRs). This information is contained in text form within the zone data files, while queries and zone transfers represent it in binary form. A resource record is made up of five fields:

Name
: The domain name the resource record refers to

Type
: The type of the resource record (see below)

TTL
: The time to live of the RR, i.e. how long resolvers should keep it in cache before considering it outdated

Class
: The type of network or software the record applies to; currently valid classes are Internet (IN), CHAOSnet (CH) and Hesiod (HS). We will discuss only the Internet class, which applies to all TCP/IP-based internets and is by far the most widespread

RDATA
: The actual resource data associated with the domain name

The main DNS record types are the following (see [RFC1035]):

`A` (Address)
: A 32-bit host IP address

`AAAA` (IPv6 Address)
: A host address in IPv6 format

`CNAME` (Canonical Name)
: Specifies an alias for a domain name, i.e. a different FQDN that can be used to refer to the same host

`KEY`
: The server's public key for TSIG and DNSSEC

`MX` (Mail eXchanger)
: Specifies a list of mail servers to which to send mail for that domain name

`NS` (Name Server)
: the authoritative name server for the domain

`PTR` (Pointer)
: A pointer to another location in the domain name space; it is mostly used to associate a domain name with an IP address in the "`in-addr.arpa`" domain for reverse name resolution

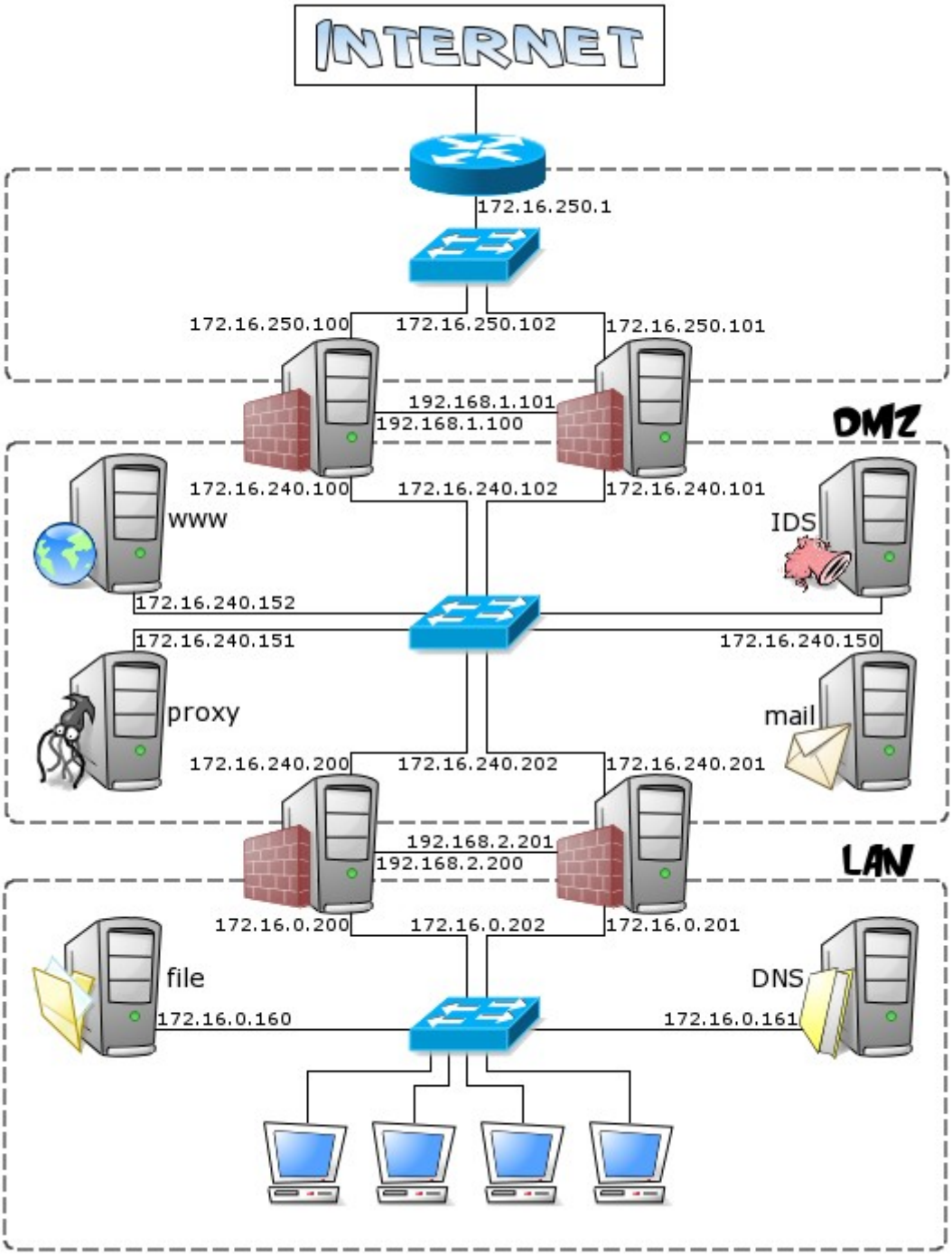
`SOA` (Start Of Authority)
: Identifies the start of a zone of authority

`TXT` (Text)
: a text string containing arbitrary data (up to 255 bytes) associated with a name

3. Base configuration

Now that we have a working knowledge of the Domain Name System architecture, it's time to move from theory to practice and set up our first domain name server.

This is the overall layout of the network in which our name server(s) will be placed.



It is a very simple network, made up of:

- a DMZ (172.16.240.0/24), containing the servers that access the internet (e.g. mail, web and proxy servers);
- a LAN (172.16.0.0/24), containing clients and servers not accessible from the public internet (e.g. file server, DHCP server);
- a router, in a small subnet (172.16.250.0/24), connecting the DMZ to the internet.

All of our systems will belong to the "`kernel-panic.it`" zone and our first DNS server will be the primary master name server for that zone; it will reside in the DMZ and answer internal queries for internet and DMZ servers' names.

3.1 The main configuration file

Bind configuration takes place in the `named.conf(5)` file, which is, by default, located in `/var/named/etc/`. You can, however, specify an alternate path using the `-c` flag of the `named(8)` command.

The configuration syntax is rather simple: it is a series of statements enclosed in curly braces and terminated with a semi-colon. Statements contain a variable number of semi-colon terminated clauses, in keyword/value form. Supported comment styles are:

- C style (`/* Multiline comment */`);
- C++ style (`// Inline comment`);
- Shell style (`# Inline comment`);

The "`options`" statement sets up global options to be used by Bind. The "`directory`" clause specifies the directory against which subsequent relative paths should be resolved. The default values are retained for unspecified clauses. E.g.:

```
options {
    # Bind runs chrooted to "/var/named/", hence "/" actually is "/var/named/"
    directory       "/";
};
```

The "`zone`" statements tell Bind what zones it is authoritative for; for each zone, the "`type`" clause specifies whether the server is a master or a slave for it and the "`file`" clause specifies the path to the corresponding zone data file. E.g.:

```
zone "kernel-panic.it" {
    type            master;
    file            "master/db.kernel-panic.it";
};
```

The names of the zone data files are free-form, but it's highly recommended to follow a reasonable naming convention to make maintenance easier. For instance, zone data files are often called `db.`*`domain`*.

In order to allow for reverse name resolution, we also need to create zone data files for each network:

```
zone "240.16.172.in-addr.arpa" {
    type            master;
    file            "master/db.172.16.240";
};

zone "250.16.172.in-addr.arpa" {
    type            master;
    file            "master/db.172.16.250";
};

zone "3.2.1.in-addr.arpa" {
    type            master;
    file            "master/db.1.2.3";
};
```

The name server will also need to map the loopback address to a name. Therefore, we will have to create specific zone data files for the "`localhost`" zone and the `127.0.0.0/8` network:

```
zone "localhost" {
    type            master;
    file            "master/db.localhost";
};

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

[RFC1912] also recommends that the "`255.in-addr.arpa`" and "`0.in-addr.arpa`" zones always be present in nameserver configurations "*to either provide nameservice for "special" addresses, or to help eliminate accidental queries for broadcast or local address to be sent off to the root nameservers*":

```
zone "255.in-addr.arpa" {
    type            master;
    file            "master/db.255";
};

zone "0.in-addr.arpa" {
    type            master;
    file            "master/db.0";
};
```

Finally, if the name server must be able to resolve internet names, we have to give it the list of the root name servers, which is specified using a `hint` zone.

```
zone "." {
    type            hint;
    file            "master/root.hint";
};
```

You can find a copy of the `root.hint` file in the `/var/named/standard` directory.

3.2 The zone data files

Zone data files contain information about the zones for which the server is authoritative, and, according to Bind configuration, they will be placed in the `/var/named/master/` directory.

Usually, the first line of a zone data file sets the default TTL for the zone, i.e. how long other DNS servers and applications are allowed to cache the record.

```
$TTL   3h
```

A zone data file may contain multiple `$TTL` statements: each applies to all subsequent records (that don't have an explicit TTL) until a new `$TTL` statement. You may want to tweak this value to find a good trade-off between bandwidth usage and data freshness.

The next entry in a zone data file is the `SOA` record, which indicates that the name server is authoritative for this zone.

```
@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour
```

Let's examine it in detail. The "`@`" symbol represents the zone the server is authoritative for; well, to be more precise, it represents the origin of the data in the zone data file, which, by default, is the same as the

zone's domain name. The origin is appended to all names in the zone data file that don't end with a trailing dot and can be modified with the `$ORIGIN` statement.

`IN` is the class of the record (Internet). `SOA` is the record type. "`dns1.kernel-panic.it.`" is the name of the primary master name server for this zone and "`danix.kernel-panic.it.`" is the mail address of the zone administrator, with the "`@`" replaced with a dot (therefore, the actual address would be "`danix@kernel-panic.it`").

Now we come to the numbers enclosed within parentheses (parentheses simply allow the record to span across multiple lines) (note that comments start with a semicolon and finish at the end of the line). The serial number is a progressive number that must be increased each time zone data is updated, otherwise slave name servers won't notice data has changed (according to [RFC1912], the recommended format for the serial number is "*YYYYMMDDnn*", where "`nn`" is the revision number). The refresh value sets how often slave name servers should check that their zone data is up to date. If the master is unreachable, the retry and expire values tell slaves at what interval to attempt to connect again and after how long to stop giving out data about the zone. The last value is the time to live for negative responses from the name servers authoritative for the zone.

Next, every zone data file has one or more `NS` records, specifying the name servers authoritative for the zone.

```
kernel-panic.it.    IN NS       dns1.kernel-panic.it.
kernel-panic.it.    IN NS       dns2.kernel-panic.it.
```

The first field of a resource record is its name and must start on the first column; it can be left blank if it is the same as the preceding one. Therefore, the above `NS` records can be shortened as:

```
                    IN NS       dns1.kernel-panic.it.
                    IN NS       dns2.kernel-panic.it.
```

The `MX` record allows you to specify the host that will manage mail for the domain name; this record has an extra parameter, a 16-bit number indicating the mail exchanger's priority (the lower the number, the higher the priority).

```
                    IN MX    0  mail.kernel-panic.it.
                    IN MX   10  mail.provider.com.
```

The next record, "`A`", is specific to forward-mapping zone data files, since it associates domain names with their IP address.

```
mail                IN A        172.16.240.150
proxy               IN A        172.16.240.151
www1                IN A        172.16.240.152
www2                IN A        172.16.240.153
dns1                IN A        172.16.240.154
dns2                IN A        172.16.240.155

mickey              IN A        172.16.0.200
                    IN A        172.16.240.200
minnie              IN A        172.16.0.201
                    IN A        172.16.240.201

router              IN A        172.16.250.1
                    IN A        1.2.3.4
[...]
```

The `CNAME` record maps an alias to its canonical name; in other words, it defines a domain name pointing to another node of the domain name space.

```
antivirus           IN CNAME    mail
cache               IN CNAME    proxy
```

Ok, we're done with forward-mapping; let's have a look at the reverse-mapping zone data files. The beginning is exactly the same: you set the default TTL and insert the `SOA` and `NS` records that we've seen before. Next come the `PTR` records, which map addresses to host names; well, to be more precise, they map names in the `in-addr.arpa` domain to domain names. Again, the origin is automatically appended to all domain names that don't end with a trailing dot, allowing you to specify only the last octet(s) of the IP addresses.

`/var/named/master/db.172.16.240`

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it danix.kernel-panic.it (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

                    IN NS       dns1.kernel-panic.it.
                    IN NS       dns2.kernel-panic.it.

100                 IN PTR      donald.kernel-panic.it.
101                 IN PTR      daisy.kernel-panic.it.
102                 IN PTR      fw-ext.kernel-panic.it.
150                 IN PTR      mail.kernel-panic.it.
151                 IN PTR      proxy.kernel-panic.it.
152                 IN PTR      www1.kernel-panic.it.
153                 IN PTR      www2.kernel-panic.it.
154                 IN PTR      dns1.kernel-panic.it.
155                 IN PTR      dns2.kernel-panic.it.
200                 IN PTR      mickey.kernel-panic.it.
201                 IN PTR      minnie.kernel-panic.it.
202                 IN PTR      fw-int.kernel-panic.it.
```

To recap, here are the complete zone data files.

3.3 Starting Bind

Running Bind is as simple as typing "`named`". The first time, you may want to run it with the `-g` flag, which runs the server in the foreground and forces all logging to stderr.

```
# named -g
Starting privilege seperation
20-Jun-2006 23:02:37.703 starting BIND 9.3.1 -g
20-Jun-2006 23:02:37.708 loading configuration from '/etc/named.conf'
[...]
20-Jun-2006 23:02:37.718 running
```

You will probably be warned that the name server couldn't find the `/etc/rndc.key` file: don't worry about this yet, we will discuss `rndc(8)` in a moment. In case `named(8)` complains about syntax errors, you can use the `named-checkconf(8)` and `named-checkzone(8)` commands to check the syntax of the Bind configuration file and the zone data files respectively.

If everything looks alright, you can test your fresh new name server with `nslookup(1)` or `dig(1)`.

```
$ nslookup mail.kernel-panic.it 127.0.0.1
Server:         127.0.0.1
Address:        127.0.0.1#53

Name:   mail.kernel-panic.it
Address: 172.16.240.150

$
```

To start Bind on system boot, simply add the following line to the /etc/rc.conf.local(8) file:

/etc/rc.conf.local

```
named_flags=""
```

3.4 rndc(8)

The rndc(8) utility allows you to communicate with the name server and send it authenticated commands over a TCP connection. It reads its configuration from the rndc.conf(5) file (by default in /var/named/etc/), which has a syntax similar to named.conf(5). The following is a sample configuration file to connect to the serverat localhost:

/var/named/etc/rndc.conf

```
options {
    default-server  localhost;
    default-port    953;
    default-key     "rndc-key";
};

server localhost {
    key             "rndc-key";
};

key "rndc-key" {
    algorithm       hmac-md5;
    secret          "jIpKqniOSfP7Nr5GTTyDkw==";
};
```

To make the name server accept rndc(8) connections, just add the following lines to your named.conf(5) file (adjusting the allow list as needed):

/var/named/etc/named.conf

```
key "rndc-key" {
    algorithm       hmac-md5;
    secret          "jIpKqniOSfP7Nr5GTTyDkw==";
};

controls {
    inet            127.0.0.1 port 953
                    allow { 127.0.0.1; }
                    keys { "rndc-key"; };
};
```

If you like things simple, you can generate the rndc(8) configuration file automatically, by using the rndc-confgen(8) utility.

3.5 Adding a slave name server

Now that your primary master name server runs fine, you may want to set up a slave name server to allow for redundancy and load sharing. Bind configuration doesn't change much:

/var/named/etc/named.conf

```
options {
    directory       "/";
};

zone "kernel-panic.it" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.kernel-panic.it";
};

zone "240.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.172.16.240";
};

zone "250.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.172.16.250";
};

zone "3.2.1.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.1.2.3";
};

# Loopback address
zone "localhost" {
    type            master;
    file            "master/db.localhost";
};

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

# Special zones
zone "255.in-addr.arpa" {
    type            master;
    file            "master/db.255";
};

zone "0.in-addr.arpa" {
    type            master;
    file            "master/db.0";
};

# Root zone
zone "." {
    type            hint;
    file            "master/db.cache";
};
```

For all the zones the slave name server is authoritative for (except for the loopback address and the "special" zones) the `type` field is now `slave`. We also had to add the `masters` clause to tell Bind the address of the primary master name server(s). The file name you provide in a zone with a `slave` type, is the file where Bind will store data transferred from the master. In this way, should the master name server be unreachable at startup, Bind will still have a local copy of the data.

4. Further Bind configuration

So we have a couple of name servers, doing a good job and allowing us to address our DMZ servers by name now. Their setup is rather simple, but can be largely enough in many situations. Anyway, Bind can do much more and solve many of the potential problems you may have to face. Let's see some the most common ones.

4.1 Views and split namespace

Our name servers are configured to return the private addresses of our DMZ servers, i.e. the addresses on the 172.16.240.0/24 network. However, some of those servers (e.g. mail and web servers) must be accessed from the internet, using a public IP address (that of the NAT device). Therefore, the name server should return different answers depending on the origin of the query: it should return the private addresses if queried from the internal network and the public address if queried from the outside.

This is called a split namespace: the real namespace is only available to the internal systems, while hosts on the internet can only see its reduced and translated version (called shadow namespace). Bind achieves this through one of its greatest features: views. Let's see them in action with a brief example.

First we need to define the group of hosts that should access the servers by their private address. We do this by defining an acl, which is simply a statement that associates a name with a group of hosts.

/var/named/et/named.conf

```
acl "internal" {
    127/8; 172.16.240/24; 172.16.0.0/24;
};
```

Next we add the views to named.conf(5) and specify different zone data files for each view.

/var/named/et/named.conf

```
view "internal" {
    # This view applies to machines in the 'internal' acl
    match-clients   { "internal" };
    # Allow 'internal' machines to query for internet names
    recursion       yes;

    zone "kernel-panic.it" {
        type        master;
        file        "master/db.kernel-panic.it"
    };

    zone "240.16.172.in-addr.arpa" {
        type        master;
        file        "master/db.172.16.240"
    };

    zone "250.16.172.in-addr.arpa" {
        type        master;
        file        "master/db.172.16.250"
    };

    zone "3.2.1.in-addr.arpa" {
        type        master;
        file        "master/db.1.2.3"
    };

    # Loopback address
    zone "localhost" {
```

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

    # Special zones
    zone "255.in-addr.arpa" {
        type        master;
        file        "master/db.255"
        };
    zone "0.in-addr.arpa" {
        type        master;
        file        "master/db.0"
        };

    # Root zone
    zone "." {
        type        hint;
        file        "master/db.cache"
    };
};

view "internet" {
    # This view applies to all the other machines
    match-clients   { any; };
    # Do not allow external machines to query for internet names
    recursion       no;

    zone "kernel-panic.it" {
        type        master;
        file        "master/db.kernel-panic.it.shadow"
    };

    zone "3.2.1.in-addr.arpa" {
        type        master;
        file        "master/db.1.2.3.shadow"
    };
};
```

The following are the shadow zone data files:

/var/named/master/db.kernel-panic.it.shadow

```
$TTL 1d

@ IN SOA dns.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

                    IN NS       dns.kernel-panic.it.
                    IN NS       dns.provider.com.

                    IN MX    0  mail.kernel-panic.it.
                    IN MX   10  mail.provider.com.

                    IN A        1.2.3.4
```

```
www                 IN CNAME    kernel-panic.it.
mail                IN CNAME    kernel-panic.it.
dns                 IN CNAME    kernel-panic.it.

*                   IN MX    0  mail.kernel-panic.it.
                    IN MX   10  mail.provider.com.
```

/var/named/master/db.1.2.3.shadow

```
$TTL 1d

@ IN SOA dns.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

                    IN NS       dns.kernel-panic.it.
                    IN NS       dns.provider.com.

4                   IN PTR      kernel-panic.it.
```

As you may have noticed, to increase DNS service availability, we have set up an additional name server hosted by our ISP.

4.2 Delegation

So far, we have taken into account only our DMZ servers: now is time for our LAN servers to enter the scene. Let's see how they relate to the other hosts:

- machines on the internet shouldn't know anything about our internal network and private servers; therefore, we won't have to modify the shadow files;
- LAN machines should only resolve names internal to our network (LAN and DMZ); surfing the web is made possible through the proxy server (in the DMZ) which is able to resolve internet names;
- DMZ servers should resolve both internal and external names.

The simplest solution would obviously be to once again take advantage of views and add the internal servers to the zone data files configured in the "`internal`" view (see above). The more interesting and scalable solution, however, is to create a new zone, "`lan.kernel-panic.it`" and delegate it to a couple of name servers (master and slave) that we will place in the LAN.

On the parent side, we simply need to add the appropriate `NS` records and the corresponding `A` records:

/var/named/master/db.kernel-panic.it

```
[...]
lan                 IN NS       dns1.lan.kernel-panic.it.
                    IN NS       dns2.lan.kernel-panic.it.

dns1.lan.kernel-panic.it.   IN A        172.16.0.161
dns2.lan.kernel-panic.it.   IN A        172.16.0.162
[...]
```

Delegated name servers will simply have to create the appropriate configuration and zone data files the usual way. You can find the complete files here.

4.3 Dynamic updates and notify

And what about our DHCP-enabled clients? Can Bind map names to dynamic IP addresses? Of course the

answer is "yes"! Bind supports dynamic update (see [RFC2136]), which enables the DHCP server to automatically add/delete/modify resource records whenever changes occur. Configuration is very simple:

/var/named/etc/named.conf

```
zone "lan.kernel-panic.it" {
    type            master;
    file            "master/db.lan.kernel-panic.it";
    allow-update    { 172.16.0.163; };
    notify          yes;
};
```

The `allow-update` clause specifies the list of IP addresses allowed to update the zone (usually just the DHCP server). It may also accept an ACL name or a TSIG key (see below for further details). For example:

/var/named/etc/named.conf

```
key dhcp-dns1.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "+io/5nabnVFgC4Tx+UAkgg==";
};

zone "lan.kernel-panic.it" {
    type            master;
    file            "master/db.lan.kernel-panic.it";
    allow-update    { key dhcp-dns1.lan.kernel-panic.it.; };
    notify          yes;
};
```

The `notify` clause tells Bind to send a `NOTIFY` announcement to all of the slave name servers for that zone to inform them that the zone data has changed. This allows Bind to minimize the delay in synchronization between master and slave name servers. Dynamic update and DNS NOTIFY work great together, beacuse Bind 9 automatically increments the zone's serial number after each update and this increment automatically triggers zone change notification.

Alternatively to `allow-update`, Bind 9 also supports the `update-policy` clause, which allows for a stricter control over which keys are allowed to update which records in a specific zone. For example:

/var/named/etc/named.conf

```
zone "lan.kernel-panic.it" {
    type            master;
    file            "master/db.lan.kernel-panic.it";
    update-policy   { grant dhcp-dns1.lan.kernel-panic.it. subdomain lan.kernel-
panic.it. A; };
    notify          yes;
};
```

Please refer to the official documentation for a detailed explanation of the `update-policy`'s syntax.

4.4 TSIG and security

So far, our only concern was having everything running smooth, without caring much about security. But we must keep in mind that part of our name servers will be exposed to the internet and, therefore, we can't ignore security issues.

The most basic security measures are implemented by default on OpenBSD: Bind runs as the unprivileged user "`named`" and chrooted inside the `/var/named` directory. This will make it much harder for attackers to exploit newly-discovered vulnerabilities.

Another important security measure is to configure Bind not to reveal its version number, just to make

attackers' lives a little more complicated.

/var/named/etc/named.conf

```
options {
    version         "Go hack yourself!";
};
```

We have already seen how views and acls can help in dealing with NAT and firewalls, but they are also a great security feature, since they allow you to select which hosts should access which information. For example, using the `recursion` substatement, you can specify which hosts are allowed to perform recursive queries against your name servers. This allows you to prevent some of the most common spoofing attacks (see [DNS&BIND]).

/var/named/etc/named.conf

```
acl "dmz" {
    127/8; 172.16.240/24;
};

view "dmz" {
    match-clients   { "dmz" };
    recursion       yes;
};

view "internet" {
    match-clients   { any; };
    recursion       no;
};
```

Needless to say, if your name server only answers queries from other name servers or for domains it is authoritative for (such as our LAN servers), you could completely turn off recursion.

/var/named/etc/named.conf

```
options {
    version         "Go hack yourself!";
    recursion       no;
};
```

Besides recursion, Bind also allows you to restrict queries and zone transfers using the `allow-query` and `allow-transfer` clauses respectively. These clauses apply to a specific zone, if used within a `zone` statement, or globally, if used within the `options` statement. E.g.:

/var/named/etc/named.conf

```
acl "dmz" { 127/8; 172.16.240/24; };

options {
    # Restrict zone transfers to our internal name servers
    allow-transfer  { 172.16.0.161; 172.16.0.162; };
};

zone "kernel-panic.it" {
    type            master;
    file            "master/db.kernel-panic.it"
    # Restrict queries to DMZ servers
    allow-query     { "dmz" };
};
```

Using acls and address match lists to restrict zone transfers is better than nothing, but using transaction

signatures, or TSIG (see [RFC2845]), is considerably better. TSIG allows name servers to authenticate DNS messages, using shared secrets (TSIG keys) and a one-way hash function (HMAC-MD5).

TSIG configuration is very simple. The first step is to create the shared key(s): the easiest way is using the `dnssec-keygen(8)` program, which creates two files, both containing the key generated.

```
# dnssec-keygen -a HMAC-MD5 -b 128 -n HOST rndc-key
Krndc-key.+157+32572
# ls
Krndc-key.+157+32572.key             Krndc-key.+157+32572.private
# cat Krndc-key.+157+32572.key
rndc-key. IN KEY 512 3 157 p2L9cNndDtTTHn6GzGHOEg==
# cat Krndc-key.+157+32572.private
Private-key-format: v1.2
Algorithm: 157 (HMAC_MD5)
Key: p2L9cNndDtTTHn6GzGHOEg==
```

The next step is to configure both name servers with the shared key:

/var/named/etc/named.conf

```
key dns1-dns2.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "p2L9cNndDtTTHn6GzGHOEg==";
};
```

Though it may look like a domain name, the argument to the `key` statement (`dns1-dns2.kernel-panic.it.`) is actually the name of the key. As suggested by the RFC, it is made up of the names of the two hosts that use it. The RFC also recommends that you use different keys for each pair of hosts.

Now that the keys are in place, we can use the `server` statement's `key` clause to tell the slave name server to sign all zone transfer requests and queries sent to its master server:

/var/named/etc/named.conf

```
server 172.16.240.154
    keys            { dns1-dns2.kernel-panic.it.; };
};
```

Similarly, on the master name server, we can restrict zone tranfers to those signed with a specific key:

/var/named/etc/named.conf

```
zone "kernel-panic.it" {
    type            master;
    file            "master/db.kernel-panic.it";
    allow-transfer  { key dns1-dns2.kernel-panic.it.; };
};
```

5. Appendix A

5.1 First draft of the configuration and zone data files

Our (modest) initial goal was to set up a couple of name servers, with a very basic configuration, and get them to do their job, without caring about security or advanced features like delegation, dynamic update or views. Since we have only seen the configuration and zone data files in pieces, you may find it useful to have a look at them in their entirety.

5.1.1 DMZ primary master

/var/named/etc/named.conf

```
options {
    directory       "/";
};

zone "kernel-panic.it" {
    type            master;
    file            "master/db.kernel-panic.it";
};

zone "240.16.172.in-addr.arpa" {
    type            master;
    file            "master/db.172.16.240";
};

zone "250.16.172.in-addr.arpa" {
    type            master;
    file            "master/db.172.16.250";
};

zone "3.2.1.in-addr.arpa" {
    type            master;
    file            "master/db.1.2.3";
};

# Loopback address
zone "localhost" {
    type            master;
    file            "master/db.localhost";
};

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

# Special zones
zone "255.in-addr.arpa" {
    type            master;
    file            "master/db.255";
};

zone "0.in-addr.arpa" {
    type            master;
    file            "master/db.0";
};

# Root zone
```

```
zone "." {
    type            hint;
    file            "master/root.hint";
};
```

/var/named/master/db.kernel-panic.it

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Mail exchangers
                IN MX   0       mail.kernel-panic.it.
                IN MX   10      mail.provider.com.

; Addresses for the canonical names
mail            IN A            172.16.240.150
proxy           IN A            172.16.240.151
www1            IN A            172.16.240.152
www2            IN A            172.16.240.153
dns1            IN A            172.16.240.154
dns2            IN A            172.16.240.155

mickey          IN A            172.16.0.200
                IN A            172.16.240.200
minnie          IN A            172.16.0.201
                IN A            172.16.240.201
donald          IN A            172.16.240.100
                IN A            172.16.250.100
daisy           IN A            172.16.240.101
                IN A            172.16.250.101
fw-int          IN A            172.16.0.202
                IN A            172.16.240.202
fw-ext          IN A            172.16.240.102
                IN A            172.16.250.102

router          IN A            172.16.250.1
                IN A            1.2.3.4

; Aliases
mk              IN CNAME        mickey
mn              IN CNAME        minnie
dn              IN CNAME        donald
ds              IN CNAME        daisy
fw1             IN CNAME        fw-int
fw2             IN CNAME        fw-ext

; Interface specific names
mk-lan          IN A            172.16.0.200
mk-dmz          IN A            172.16.240.200
mn-lan          IN A            172.16.0.201
mn-dmz          IN A            172.16.240.201
dn-dmz          IN A            172.16.240.100
dn-ext          IN A            172.16.250.100
```

```
ds-dmz          IN A            172.16.240.101
ds-ext          IN A            172.16.250.101
fw1-lan         IN A            172.16.0.202
fw1-dmz         IN A            172.16.240.202
fw2-dmz         IN A            172.16.240.102
fw2-ext         IN A            172.16.250.102
router-int      IN A            172.16.250.1
router-ext      IN A            1.2.3.4
```

/var/named/master/db.172.16.240

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
100             IN PTR          donald.kernel-panic.it.
101             IN PTR          daisy.kernel-panic.it.
102             IN PTR          fw-ext.kernel-panic.it.
150             IN PTR          mail.kernel-panic.it.
151             IN PTR          proxy.kernel-panic.it.
152             IN PTR          www1.kernel-panic.it.
153             IN PTR          www2.kernel-panic.it.
154             IN PTR          dns1.kernel-panic.it.
155             IN PTR          dns2.kernel-panic.it.
200             IN PTR          mickey.kernel-panic.it.
201             IN PTR          minnie.kernel-panic.it.
202             IN PTR          fw-int.kernel-panic.it.
```

/var/named/master/db.172.16.250

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          router.kernel-panic.it.
100             IN PTR          donald.kernel-panic.it.
101             IN PTR          daisy.kernel-panic.it.
102             IN PTR          fw-ext.kernel-panic.it.
```

/var/named/master/db.1.2.3

```
$TTL 3h
```

```
@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
4               IN PTR          router.kernel-panic.it.
```

/var/named/master/db.localhost

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses for the canonical names
                IN A            127.0.0.1
```

/var/named/master/db.127.0.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          localhost.
```

/var/named/master/db.255

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour
```

```
; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

/var/named/master/db.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

/var/named/master/root.hint

```
;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     192.228.79.201

; 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

; operated by VeriSign, Inc.
.                        3600000      NS    J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET.      3600000      A     192.58.128.30

; operated by RIPE NCC
```

```
.                        3600000      NS    K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET.      3600000      A     193.0.14.129

; operated by ICANN
.                        3600000      NS    L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET.      3600000      A     198.32.64.12

; operated by WIDE
.                        3600000      NS    M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET.      3600000      A     202.12.27.33
```

5.1.2 DMZ secondary master

/var/named/etc/named.conf

```
options {
    directory       "/";
};

zone "kernel-panic.it" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.kernel-panic.it";
};

zone "240.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.172.16.240";
};

zone "250.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.172.16.250";
};

zone "3.2.1.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.1.2.3";
};

# Loopback address
zone "localhost" {
    type            master;
    file            "master/db.localhost";
};

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

# Special zones
zone "255.in-addr.arpa" {
    type            master;
    file            "master/db.255";
};

zone "0.in-addr.arpa" {
    type            master;
```

```
    file            "master/db.0";
};

# Root zone
zone "." {
    type            hint;
    file            "master/root.hint";
};
```

/var/named/master/db.localhost

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses for the canonical names
                IN A            127.0.0.1
```

/var/named/master/db.127.0.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          localhost.
```

/var/named/master/db.255

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

/var/named/master/db.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

/var/named/master/root.hint

```
;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     192.228.79.201

; 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

; operated by VeriSign, Inc.
.                        3600000      NS    J.ROOT-SERVERS.NET.
J.ROOT-SERVERS.NET.      3600000      A     192.58.128.30

; operated by RIPE NCC
.                        3600000      NS    K.ROOT-SERVERS.NET.
K.ROOT-SERVERS.NET.      3600000      A     193.0.14.129

; operated by ICANN
.                        3600000      NS    L.ROOT-SERVERS.NET.
L.ROOT-SERVERS.NET.      3600000      A     198.32.64.12
```

```
; operated by WIDE
.                       3600000      NS    M.ROOT-SERVERS.NET.
M.ROOT-SERVERS.NET.     3600000      A     202.12.27.33
```

5.2 Final version of the configuration and zone data files

Once we had our name servers working, we decided to get into the serious stuff and configure some of Bind's most useful features, like delegation, views, dynamic update and TSIG. Below are the complete final configuration and zone data files.

5.2.1 DMZ primary master

/var/named/etc/named.conf

```
/*********************************************************************
 * This is the primary master name server for the "kernel-panic.it" zone.     *
 * It accepts queries from both external and DMZ hosts, but uses different    *
 * namespaces. It accepts zone transfer requests only from the ISP's name     *
 * servers, the DMZ secondary master and the LAN name servers.                *
 *********************************************************************/

/* TSIG keys ***********************************************************/
key dns1-dns2.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "7U86ip+B+SRYirLGm4lxfg==";
};

key dns1-dns1.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "bvVFyHOWV/YjIdBbpAJZWQ==";
};

key dns1-dns2.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "1sMX8Xs5zEhpekJDyyNTDA==";
};

/* ACLs ****************************************************************/
acl "dmz" {
    127/8; 172.16.240/24;
};

acl "isp-ns" {
    1.2.3.5; 1.2.3.6;
};

acl "dmz-slaves" {
    key dns1-dns2.kernel-panic.it.;
};

acl "lan-slaves" {
    key dns1-dns1.lan.kernel-panic.it.;
    key dns1-dns2.lan.kernel-panic.it.;
};

/* rndc configuration **************************************************/
key "rndc-key" {
    algorithm       hmac-md5;
    secret          "Hp3cRzIhGLuzdPw53M2pHw==";
};
```

```
controls {
    inet            127.0.0.1 port 953
                    allow { 127.0.0.1; }
                    keys { "rndc-key"; };
};

/* Options *****************************************************************/
options {
    directory       "/";
    version         "Go hack yourself!";
};

/* Authoritative zones *****************************************************/
view "dmz" {
    match-clients   { "dmz"; };
    allow-transfer  { "dmz-slaves"; "lan-slaves"; };
    recursion       yes;

    zone "kernel-panic.it" {
        type        master;
        file        "master/db.kernel-panic.it";
    };

    zone "240.16.172.in-addr.arpa" {
        type        master;
        file        "master/db.172.16.240";
    };

    zone "250.16.172.in-addr.arpa" {
        type        master;
        file        "master/db.172.16.250";
    };

    zone "3.2.1.in-addr.arpa" {
        type        master;
        file        "master/db.1.2.3";
    };

    # Loopback address
    zone "localhost" {
        type        master;
        file        "master/db.localhost";
    };

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

    # Special zones
    zone "255.in-addr.arpa" {
        type        master;
        file        "master/db.255";
    };

    zone "0.in-addr.arpa" {
        type        master;
        file        "master/db.0";
    };

    # Root zone
    zone "." {
        type        hint;
```

```
        file        "master/root.hint";
    };
};

view "internet" {
    match-clients   { any; };
    allow-transfer  { "isp-ns"; };
    recursion       no;

    zone "kernel-panic.it" {
        type        master;
        file        "master/db.kernel-panic.it.shadow";
    };

    zone "3.2.1.in-addr.arpa" {
        type        master;
        file        "master/db.1.2.3.shadow";
    };
};
```

/var/named/etc/rndc.conf

```
options {
    default-server  localhost;
    default-port    953;
    default-key     "rndc-key";
};

server localhost {
    key             "rndc-key";
};

key "rndc-key" {
    algorithm       hmac-md5;
    secret          "Hp3cRzIhGLuzdPw53M2pHw==";
};
```

/var/named/master/db.kernel-panic.it

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Mail exchangers
                IN MX   0       mail.kernel-panic.it.
                IN MX   10      mail.provider.com.

; Delegated zone
lan             IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.

dns1.lan        IN A            172.16.0.161
dns2.lan        IN A            172.16.0.162
```

```
; Addresses for the canonical names
mail            IN A            172.16.240.150
proxy           IN A            172.16.240.151
www1            IN A            172.16.240.152
www2            IN A            172.16.240.153
dns1            IN A            172.16.240.154
dns2            IN A            172.16.240.155

mickey          IN A            172.16.0.200
                IN A            172.16.240.200
minnie          IN A            172.16.0.201
                IN A            172.16.240.201
donald          IN A            172.16.240.100
                IN A            172.16.250.100
daisy           IN A            172.16.240.101
                IN A            172.16.250.101
fw-int          IN A            172.16.0.202
                IN A            172.16.240.202
fw-ext          IN A            172.16.240.102
                IN A            172.16.250.102

router          IN A            172.16.250.1
                IN A            1.2.3.4

; Aliases
mk              IN CNAME        mickey
mn              IN CNAME        minnie
dn              IN CNAME        donald
ds              IN CNAME        daisy
fw1             IN CNAME        fw-int
fw2             IN CNAME        fw-ext

; Interface specific names
mk-lan          IN A            172.16.0.200
mk-dmz          IN A            172.16.240.200
mn-lan          IN A            172.16.0.201
mn-dmz          IN A            172.16.240.201
dn-dmz          IN A            172.16.240.100
dn-ext          IN A            172.16.250.100
ds-dmz          IN A            172.16.240.101
ds-ext          IN A            172.16.250.101
fw1-lan         IN A            172.16.0.202
fw1-dmz         IN A            172.16.240.202
fw2-dmz         IN A            172.16.240.102
fw2-ext         IN A            172.16.250.102
router-int      IN A            172.16.250.1
router-ext      IN A            1.2.3.4
```

/var/named/master/db.172.16.240

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

```
; Addresses (pointing to canonical names)
100             IN PTR          donald.kernel-panic.it.
101             IN PTR          daisy.kernel-panic.it.
102             IN PTR          fw-ext.kernel-panic.it.
150             IN PTR          mail.kernel-panic.it.
151             IN PTR          proxy.kernel-panic.it.
152             IN PTR          www1.kernel-panic.it.
153             IN PTR          www2.kernel-panic.it.
154             IN PTR          dns1.kernel-panic.it.
155             IN PTR          dns2.kernel-panic.it.
200             IN PTR          mickey.kernel-panic.it.
201             IN PTR          minnie.kernel-panic.it.
202             IN PTR          fw-int.kernel-panic.it.
```

/var/named/master/db.172.16.250

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          router.kernel-panic.it.
100             IN PTR          donald.kernel-panic.it.
101             IN PTR          daisy.kernel-panic.it.
102             IN PTR          fw-ext.kernel-panic.it.
```

/var/named/master/db.1.2.3

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Mail exchangers
                IN MX   0       mail.kernel-panic.it.
                IN MX   10      mail.provider.com.

; Addresses (pointing to canonical names)
4               IN PTR          router.kernel-panic.it.
```

/var/named/master/db.localhost

```
$TTL 3h
```

```
@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses for the canonical names
                IN A            127.0.0.1
```

/var/named/master/db.127.0.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          localhost.
```

/var/named/master/db.255

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

/var/named/master/db.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

/var/named/master/db.kernel-panic.it.shadow

```
$TTL 1d

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns.kernel-panic.it.
                IN NS           dns.provider.com.

; Mail exchangers
                IN MX   0       mail.kernel-panic.it.
                IN MX   10      mail.provider.com.

; Addresses for the canonical names
                IN A            1.2.3.4

; Aliases
www             IN CNAME        kernel-panic.it.
mail            IN CNAME        kernel-panic.it.
dns             IN CNAME        kernel-panic.it.

; Deault mail exchangers
*               IN MX   0       mail.kernel-panic.it.
                IN MX   10      mail.provider.com.
```

/var/named/master/db.1.2.3.shadow

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns.provider.com.

; Addresses (pointing to canonical names)
4               IN PTR          kernel-panic.it.
```

5.2.2 DMZ secondary master

/var/named/etc/named.conf

```
/*************************************************************************
 * This is the secondary master name server for the "kernel-panic.it" zone.   *
 * It accepts queries only from DMZ hosts and zone transfer requests from the *
 * ISP's name servers and the LAN name servers.                               *
 *************************************************************************/

/* TSIG keys ***************************************************************/
key dns1-dns2.kernel-panic.it. {
    algorithm       hmac-md5;
```

```
    secret          "7U86ip+B+SRYirLGm4lxfg==";
};

key dns2-dns1.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "uyUkoNVWKxah/Zr+Xcd8vQ==";
};

key dns2-dns2.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "Y2hqf7mCvqnQf8UFOJ2CyA==";
};

server 172.16.240.154 {
    keys            { dns1-dns2.kernel-panic.it.; };
};

/* ACLs ********************************************************************/
acl "dmz" {
    127/8; 172.16.240/24;
};

acl "isp-ns" {
    1.2.3.5; 1.2.3.6;
};

acl "lan-slaves" {
    key dns2-dns1.lan.kernel-panic.it.;
    key dns2-dns2.lan.kernel-panic.it.;
};

/* rndc configuration ******************************************************/
key "rndc-key" {
    algorithm       hmac-md5;
    secret          "3F5oVjZ2fRE/7x2NPy8rZA==";
};

controls {
    inet            127.0.0.1 port 953
                    allow { 127.0.0.1; }
                    keys { "rndc-key"; };
};

/* Options *****************************************************************/
options {
    directory       "/";
    version         "Go hack yourself!";
    allow-query     { "dmz"; };
    allow-transfer  { "isp-ns"; "lan-slaves"; };
    recursion       yes;
};

/* Authoritative zones *****************************************************/
zone "kernel-panic.it" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.kernel-panic.it";
};

zone "240.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.172.16.240";
```

```
};

zone "250.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.172.16.250";
};

zone "3.2.1.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; };
    file            "slave/bak.1.2.3";
};

# Loopback address
zone "localhost" {
    type            master;
    file            "master/db.localhost";
};

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

# Special zones
zone "255.in-addr.arpa" {
    type            master;
    file            "master/db.255";
};

zone "0.in-addr.arpa" {
    type            master;
    file            "master/db.0";
};

# Root zone
zone "." {
    type            hint;
    file            "master/root.hint";
};
```

/var/named/etc/rndc.conf

```
options {
    default-server  localhost;
    default-port    953;
    default-key     "rndc-key";
};

server localhost {
    key             "rndc-key";
};

key "rndc-key" {
    algorithm       hmac-md5;
    secret          "3F5oVjZ2fRE/7x2NPy8rZA==";
};
```

/var/named/master/db.localhost

```
$TTL 3h
```

```
@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses for the canonical names
                IN A            127.0.0.1
```

/var/named/master/db.127.0.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          localhost.
```

/var/named/master/db.255

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

/var/named/master/db.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.kernel-panic.it.
                IN NS           dns2.kernel-panic.it.
```

5.2.3 LAN primary master

/var/named/etc/named.conf

```
/*****************************************************************************
 * This is the primary master name server for the "lan.kernel-panic.it" zone  *
 * and a secondary master name server for the "kernel-panic.it" zone.         *
 * It accepts queries from internal hosts and zone transfers requests only    *
 * from the LAN secondary master. The DHCP server can dynamically update      *
 * clients resource records.                                                  *
 *****************************************************************************/

/* TSIG keys ******************************************************************/
key dns1-dns1.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "bvVFyHOWV/YjIdBbpAJZWQ==";
};

key dns2-dns1.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "uyUkoNVWKxah/Zr+Xcd8vQ==";
};

key dns1.lan-dns2.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "Cn0Xj2v6u7CGNeRSIfS1JQ==";
};

key dns1.lan-dhcp.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "9+MU2qJwwl9nk7ptG84kpQ==";
};

server 172.16.240.154 {
    keys            { dns1-dns1.lan.kernel-panic.it.; };
};

server 172.16.240.155 {
    keys            { dns2-dns1.lan.kernel-panic.it.; };
};

/* ACLs ***********************************************************************/
acl "dmz" {
    172.16.240/24;
};

acl "lan" {
    127/8; 172.16.0/24;
};

acl "lan-slaves" {
    key dns1.lan-dns2.lan.kernel-panic.it.;
};

/* rndc configuration *********************************************************/
key "rndc-key" {
    algorithm       hmac-md5;
    secret          "D6P3H5E+cWyeuSVEMZH5+Q==";
};

controls {
    inet            127.0.0.1 port 953
                    allow { 127.0.0.1; }
                    keys { "rndc-key"; };
```

```
};

/* Options *********************************************************************/
options {
    directory       "/";
    version         "Go hack yourself!";
    allow-query     { "dmz"; "lan"; };
    allow-transfer  { "lan-slaves"; };
    recursion       no;
};

/* Authoritative zones *********************************************************/
zone "lan.kernel-panic.it" {
    type            master;
    file            "master/db.lan.kernel-panic.it";
    update-policy   { grant dns1.lan-dhcp.lan.kernel-panic.it.
                            subdomain lan.kernel-panic.it. A; };
    notify          yes;
};

zone "0.16.172.in-addr.arpa" {
    type            master;
    file            "master/db.172.16.0";
};

zone "kernel-panic.it" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.kernel-panic.it";
};

zone "240.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.172.16.240";
};

zone "250.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.172.16.250";
};

zone "3.2.1.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.1.2.3";
};

# Loopback address
zone "localhost" {
    type            master;
    file            "master/db.localhost";
};

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

# Special zones
zone "255.in-addr.arpa" {
    type            master;
```

```
    file            "master/db.255";
};

zone "0.in-addr.arpa" {
    type            master;
    file            "master/db.0";
};
```

/var/named/etc/rndc.conf

```
options {
    default-server  localhost;
    default-port    953;
    default-key     "rndc-key";
};

server localhost {
    key             "rndc-key";
};

key "rndc-key" {
    algorithm       hmac-md5;
    secret          "D6P3H5E+cWyeuSVEMZH5+Q==";
};
```

/var/named/master/db.lan.kernel-panic.it

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.

; Mail exchangers
                IN MX   0       mail.kernel-panic.it.
                IN MX   10      mail.provider.com.

; Addresses for the canonical names
file            IN A            172.16.0.160
dns1            IN A            172.16.0.161
dns2            IN A            172.16.0.162
dhcp            IN A            172.16.0.163
```

/var/named/master/db.172.16.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
```

```
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.

; Addresses (pointing to canonical names)
160             IN PTR          file.lan.kernel-panic.it.
161             IN PTR          dns1.lan.kernel-panic.it.
162             IN PTR          dns2.lan.kernel-panic.it.
163             IN PTR          dhcp.lan.kernel-panic.it.
200             IN PTR          mickey.kernel-panic.it.
201             IN PTR          minnie.kernel-panic.it.
202             IN PTR          fw-int.kernel-panic.it.
```

/var/named/master/db.localhost

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.

; Addresses for the canonical names
                IN A            127.0.0.1
```

/var/named/master/db.127.0.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          localhost.
```

/var/named/master/db.255

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.
```

/var/named/master/db.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.
```

5.2.4 LAN secondary master

/var/named/etc/named.conf

```
/*********************************************************************
 * This is a secondary master name server for the "lan.kernel-panic.it" and   *
 * "kernel-panic.it" zones. It accepts queries only from internal hosts.      *
 *********************************************************************/

/* TSIG keys ***********************************************************/
key dns1-dns2.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "1sMX8Xs5zEhpekJDyyNTDA==";
};

key dns2-dns2.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "Y2hqf7mCvqnQf8UFOJ2CyA==";
};

key dns1.lan-dns2.lan.kernel-panic.it. {
    algorithm       hmac-md5;
    secret          "Cn0Xj2v6u7CGNeRSIfS1JQ==";
};

server 172.16.240.154 {
    keys            { dns1-dns2.lan.kernel-panic.it.; };
};

server 172.16.240.155 {
    keys            { dns2-dns2.lan.kernel-panic.it.; };
};

server 172.16.0.161 {
    keys            { dns1.lan-dns2.lan.kernel-panic.it.; };
};

/* ACLs ****************************************************************/
acl "dmz" {
    172.16.240/24;
};

acl "lan" {
    127/8; 172.16.0/24;
};

/* rndc configuration **************************************************/
key "rndc-key" {
    algorithm       hmac-md5;
```

```
    secret          "vb5zPXhAfsJx+5zl4cC5Xg==";
};

controls {
    inet            127.0.0.1 port 953
                    allow { 127.0.0.1; }
                    keys { "rndc-key"; };
};

/* Options *****************************************************************/
options {
    directory       "/";
    version         "Go hack yourself!";
    allow-query     { "dmz"; "lan"; };
    allow-transfer  { none; };
    recursion       no;
};

/* Authoritative zones *****************************************************/
zone "lan.kernel-panic.it" {
    type            slave;
    masters         { 172.16.0.161; };
    file            "slave/bak.lan.kernel-panic.it";
};

zone "0.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.0.161; };
    file            "slave/bak.172.16.0";
};

zone "kernel-panic.it" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.kernel-panic.it";
};

zone "240.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.172.16.240";
};

zone "250.16.172.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.172.16.250";
};

zone "3.2.1.in-addr.arpa" {
    type            slave;
    masters         { 172.16.240.154; 172.16.240.155; };
    file            "slave/bak.1.2.3";
};

# Loopback address
zone "localhost" {
    type            master;
    file            "master/db.localhost";
};

zone "0.0.127.in-addr.arpa" {
    type            master;
```

```
    file            "master/db.127.0.0";
};

# Special zones
zone "255.in-addr.arpa" {
    type            master;
    file            "master/db.255";
};

zone "0.in-addr.arpa" {
    type            master;
    file            "master/db.0";
};
```

/var/named/etc/rndc.conf

```
options {
    default-server  localhost;
    default-port    953;
    default-key     "rndc-key";
};

server localhost {
    key             "rndc-key";
};

key "rndc-key" {
    algorithm       hmac-md5;
    secret          "vb5zPXhAfsJx+5zl4cC5Xg==";
};
```

/var/named/master/db.localhost

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.

; Addresses for the canonical names
                IN A            127.0.0.1
```

/var/named/master/db.127.0.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
```

```
                IN NS           dns2.lan.kernel-panic.it.

; Addresses (pointing to canonical names)
1               IN PTR          localhost.
```

/var/named/master/db.255

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.
```

/var/named/master/db.0

```
$TTL 3h

@ IN SOA dns1.kernel-panic.it. danix.kernel-panic.it. (
    2007020601 ; serial
    3h         ; refresh after 3 hours
    1h         ; retry after 1 hour
    1w         ; expire after 1 week
    1h )       ; negative caching TTL of 1 hour

; Name servers
                IN NS           dns1.lan.kernel-panic.it.
                IN NS           dns2.lan.kernel-panic.it.
```

6. Appendix B

6.1 References

- [RFC1034] - RFC 1034 - Domain names - concepts and facilities
- [RFC1035] - RFC 1035 - Domain names - implementation and specification
- [RFC1912] - RFC 1912 - Common DNS Operational and Configuration Errors
- [RFC2136] - RFC 2136 - Dynamic Updates in the Domain Name System (DNS UPDATE)
- [RFC2845] - RFC 2845 - Secret Key Transaction Authentication for DNS (TSIG)
- [DNS&BIND] - *DNS and BIND, Fifth Edition*, Paul Albitz and Cricket Liu, O'Reilly, 2006

6.2 Bibliography

- BIND 9 Administrator Reference Manual
- *Pro DNS and BIND*, Ron Aitchison, Apress, 2005
- BIND for the Small LAN
- DNS Resource Record (RR) Types & DNS Parameters (IANA)
- DNS Spoofing techniques