Managing a lot of identical DNS zones with BIND

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

First publication of this article on 6 October 2007. Last update on of 16 October 2007

https://www.bortzmeyer.org/identical-domains-with-bind.html

A very common question on mailing lists of other fora dedicated to the DNS or specifically to BIND is 'How can I have a single zone file for all the domains I manage, since their content is more or less identical?'

There are several ways to do so, I describe them here, while developing my personal preference, file sharing, more in depth.

The solutions are :

— Use only one zone file, shared by all zones, with all the content in it being relative,
— Generate zone files from some sort of a database by a program,
— Use DNAME records.

The first solution is often my favorite. The idea is to have one zone file, with a content made mostly of **relative** domain names. That is using `@` instead of the zone name, not adding the zone name and the final dot, etc. BIND lets you use the at sign (@) to refer to the zone to which `named.conf` assigns the zone file. An example of such a zone file is :

```
$TTL 38400
@       IN      SOA     ns1.myorg.net. hostmaster.myorg.net. (
                        2007100601
                        10800
                        3600
                        604800
                        38400 )
        IN       NS     ns1.myorg.net.
        IN       NS     ns2.myorg.net.
        IN       MX  10   smtp.myorg.net.
www                     IN      A        192.0.2.34
imap    IN   A 192.0.2.45
; Add more at your leisure
```

The name of the zone never appears on the left-hand side of the resource records above. If this zone is loaded for `example.org`, `www.example.org` will resolve to 192.0.2.34 (because `www` is written without a final dot, the name of the zone will be automatically appended).

Once you have such a file, then use a symbolic link on the master name server, so that all files links to this one. For instance, if `named.conf` contains :

```
zone "example.org" {
    type master;
    file "example.org.db";
};
zone "example.net" {
   type master;
   file "example.net.db";
};
```

you just have to create the symbolic links to the reference file :

```
% ln -s reference-zone.db example.org.db
% ln -s reference-zone.db example.net.db
```

(Another way to have all domains use the same zone file is to name it in `named.conf` zone statements, that way :

```
zone "example.org" {
    type master;
    file "reference-zone.db";
};
zone "example.net" {
   type master;
   file "reference-zone.db";
};
```

Wether it is simpler for your co-workers or your successors is a matter of taste)

The O'Reilly book 'DNS & BIND Cookbook' has a chaper online, which is precisely the chapter dealing with this issue `<http://www.oreilly.com/catalog/dnsbindckbk/chapter/ch02.pdf>` ('Using a Single Data File for Multiple Zones').

The second solution, the generation of the zone files from a program which reads some sort of a database is a bit more complicated (you need someone able to write a twenty-lines program in Perl or Python) but is more powerful since it can even handle cases where the zone files are not exactly identical.

The 'database' does not have to be a relational database engine, text files with a format are databases, too, and so are XML or JSON files.

The third solution is to use the DNAME resource records described in RFC 6672[1]. I find DNAME complicated to understand and counter-intuitive and not supported by all DNS implementations (but note that only the authoritative name servers of the zone have to support them).

1. Pour voir le RFC de numéro NNN, `https://www.ietf.org/rfc/rfcNNN.txt`, par exemple `https://www.ietf.org/rfc/rfc6672.txt`

The main advantage of DNAME records is of memory use : it gives a significant memory benefit (with file sharing, my favorite approach, the memory used by BIND is proportional to the number of zones multiplied by the size of the zone ; with DNAME, most zones use very little memory).

Here is an example. We assume that the parent zone does not allow you to insert DNAME so you have to do everything yourself. We want `example.net` to be an equivalent of `example.org`. The zone file for `example.net` will look like :

```
; DNAME only handles subdomains, we need to duplicate the MX, the NS,
; etc, of the apex.
@       IN      SOA     ns1.myorg.net. hostmaster.myorg.net. (
                        2007100601
                        10800
                        3600
                        604800
                        38400 )
        IN       NS     ns1.myorg.net.
        IN       NS     ns2.myorg.net.
        IN       MX  10   smtp.myorg.net.

;
        IN       DNAME example.org.
; Add nothing more, all domain names *under* example.net will be
; "created" by the DNAME record above.
```

Remember that DNAME records only work for the domain names **underneath** the domain you manage (the RFC says "a new DNS Resource Record called "DNAME", which provides the capability to map an entire **subtree** of the DNS name space to another domain). So, `www.example.net` will be handled by the DNAME record (and rewritten as `www.example.org`) but we need to put manually the MX and NS records of the domain `example.net`.