

## DNS Exercise - Part II

=====

In this exercise, you will \*delegate\* a subdomain of your own domain.

In order to keep things simple, it will work like this: each machine will delegate a subdomain to the next host along (which will be the master) and the next one after that (which will be the slave).

Example:

- \* Let's say you are host103.dns and have domain `utopia.dns` already set up
- \* You will pick a subdomain, let's say `elmo.utopia.dns`
- \* You will delegate this subdomain to host104.dns and host105.dns (host105.dns is the master and host106.dns is the slave. In practice, when you are delegating it doesn't really matter which is master, because all authoritative nameservers appear the same to the outside world)
- \* Because you are a conscientious domain owner, you won't add the delegation to host105.dns and host106.dns until they have correctly set up their authoritative nameservice for the domain, and you've tested it.

Now, because this pattern is repeated by everyone else in the class, it also means that:

- \* You will receive delegation for a domain from host102.dns (for which you will be the master)
- \* You will receive delegation for another domain from host101.dns (for which you will be slave, with host102.dns as the master)

So you will be doing three different jobs! You will have to:

1. set yourself up as master for the domain delegated from host102.dns
2. set yourself up as slave for the domain delegated from host101.dns
3. and delegate a subdomain of yours to host104.dns and host105.dns

This means that a lot will be going on at once - so please follow the worksheet carefully!

-----

### Exercise parameters

-----

To start, please write the following missing information down on a piece of paper, numbered as (1) to (5). If it's not clear to you what needs to be done, please ask.

- (1) My machine is:      host\_\_\_\_\_.dns
- (2) I control domain:      \_\_\_\_\_.dns  
(this is the domain you set up in the previous exercise, for which your machine is the master)
- (3) I am going to delegate this subdomain:

\_\_\_\_\_.\_\_\_\_\_ .dns  
(2)

and I am going to delegate it to:

(4) host\_\_\_\_\_ .dns (= myhost+1) [master]

(5) host\_\_\_\_\_ .dns (= myhost+2) [slave]

Wrap around to host101 and/or host101 if you run past the highest-numbered host in the class (whoever has the highest number)

Once you have done this, copy fields (1)-(5) from the worksheet for the machine numbered ONE BELOW YOU into fields (6)-(10) (write them down on paper). If you are host101, then the machine "below" you is the highest-numbered host in use in the class.

(6) Their machine is: host\_\_\_\_\_ .dns (= myhost-1)

(7) They control domain: \_\_\_\_\_ .dns

(8) They are going to delegate this subdomain:

\_\_\_\_\_.\_\_\_\_\_ .dns  
(7)

and they are going to delegate it to:

(9) host\_\_\_\_\_ .dns (= myhost) [master] \*\*

(10) host\_\_\_\_\_ .dns (= myhost+1) [slave]

Next, copy fields (1)-(5) from the machine TWO BELOW YOU into fields (11)-(15) (on paper)

(11) Their machine is: host\_\_\_\_\_ .dns (= myhost-2)

(12) They control domain: \_\_\_\_\_ .dns

(13) They are going to delegate this subdomain:

\_\_\_\_\_.\_\_\_\_\_ .dns  
(12)

and they are going to delegate it to:

(14) host\_\_\_\_\_ .dns (= myhost-1) [master]

(15) host\_\_\_\_\_ .dns (= myhost) [slave] \*\*

-----  
Step 1: Set up as master for domain (8)  
-----

You are going to be master for the domain given in (8). So the first step is to create a zonefile for this domain:

```
# vi /etc/namedb/master/_____._____.dns
```

Or if you prefer PICO:

```
# pico /etc/namedb/master/_____._____.dns
(8)
```

... and insert the following contents in the file (don't include the sections with numbers like '(9)' - these are just notes.

```
$TTL 10m
@      IN      SOA      host_____.dns. yourname.example.com. (
                                (9)
                                2010070101
                                10m
                                10m
                                4w
                                10m )

      IN      NS       host_____.dns.
                                (9)
      IN      NS       host_____.dns.
                                (10)

www    IN      A       10.20.XXX.1    ; replace with your own IP
```

Replace "yourname.example.com." with your modified E-mail address as in the previous exercise, and use the current YYYYMMDD00 as the serial number.

Now validate the zonefile you have created:

```
# named-checkzone _____._____.dns
/etc/namedb/master/_____._____.dns
(8)
(8)
```

If this reports any errors, then fix them.

Next, edit `/etc/namedb/named.conf` to configure bind as master for that zone using the zonefile you have created:

(Remember, if you don't know "vi", use "pico")

```
# vi /etc/namedb/named.conf
```

... add this entry:

```
zone "_____._____.dns" {
    (8)
    type master;
    file "master/_____._____.dns";
    (8)
```

```
        allow-transfer { 10.20.XXX.1; };  
};
```

Replace 10.20.XXX.1 with the IP address of machine (10), which is going to be slave for this zone.

Then validate your modified configuration file:

```
# named-checkconf
```

Again, if this reports any errors then fix them. Now get your nameserver to reload its conf file and your new zone:

```
# rndc reload  
# tail /var/log/messages
```

Once again, check for any errors and fix them. Finally, test that your machine is giving out authoritative answers:

```
# dig +noredc @10.20.XXX.1 _____.(13)_____.(13).dns. soa
```

replacing 10.20.XXX.1 with your own IP address. Check that you get a SOA response with the expected serial number, and the AA flag is present.

Good - you are half way to getting delegation for this domain (it won't be done until your slave is set up properly)

Step 2: Set up as slave for domain (13)

-----  
The host below you has set themselves up as master for the domain you wrote in (13), and will expect you to be the slave.

So now edit `/etc/namedb/named.conf` to enable yourself as slave for this domain:

```
# pico /etc/namedb/named.conf  
or  
# vi /etc/namedb/named.conf
```

... add this entry:

```
zone "_____(13)_____.(13).dns" {  
    type slave;  
    file "slave/_____(13)_____.(13).dns";  
    masters { 10.20.XXX.1; };  
};
```

Replace 10.20.XXX.1 with the IP address of the master, the machine listed in space (14).

Now validate your modified configuration file:

```
# named-checkconf
```

If this reports any errors then fix them. Now get your nameserver to reload its conf file:

```
# rndc reload
# tail /var/log/messages
```

Once again, check for any errors and fix them.

If the machine below you has already set themselves up as master, then the zone transfer should take place within a few seconds. You can check this has happened by looking in `/var/log/messages` again, and checking whether the slave zone file has been created:

```
# ls /etc/namedb/slave/
```

If not, then either the machine below you has not finished setting themselves up as master for the domain, or else they have not permitted access to your IP address to allow you to copy the zone. You can check using these commands:

```
# dig +nored @10.20.XXX.1 _____ .dns. soa
                        (14)                (13)
```

```
# dig @10.20.XXX.1 _____ .dns. axfr
                        (14)                (13)
```

The first should show you the SOA record with the correct serial number; the second should show you the entire contents of their zone file. If these are OK, then the zone transfer should take place within a few minutes.

Step 3: Test before delegation of domain (3)

-----

You are now about to delegate the domain you chose in (3) to the machines listed in (4) and (5); here you are acting in the role of a domain registry.

However, before you perform this delegation, you should check that they are both set up correctly, especially that they are both authoritative for the domain in question. Otherwise, you would be creating a lame delegation, which is not good.

Test the master using the following command:

```
# dig +nored @host_____ .dns. _____ .dns. soa
                        (4)                (3)
```

Check:

- \* Is the response authoritative? (Flag AA)
- \* Does the SOA record list the correct host as the master?
- \* Are the nameserver (NS) records in the Authority section correct? There should be two NS records, one giving the hostname of the master (4) and

- one the hostname of the slave (5)
- \* Make a note of the zone serial number

And then test the slave:

```
# dig +noredc @host____.dns. _____._____.dns. soa
                        (5)                        (3)
```

Check:

- \* Is the response authoritative? (Flag AA)
- \* Does the zone serial number match that given by the master?
- \* Are the nameserver (NS) records in the Authority section correct?

If any of these checks fail, explain what the problem was to the owners of those machines. Don't proceed until they have fixed the problems - and make sure you have *\*re-tested\** the servers to ensure the problems really have been fixed.

#### Step 4: Delegate domain (3)

-----

This is the point at which you delegate the subdomain (3); all queries for this subdomain will be referred to the servers (4) and (5).

Edit the zonefile for your domain (2):

```
# pico /etc/namedb/master/_____.dns
or
# vi /etc/namedb/master/_____.dns
                        (2)
```

... add these RRs (records)

```
_____ IN      NS      host____.dns.
(3)      IN      NS      host____.dns.
                        (4)
                        (5)
```

Note: in the space marked (3) you just put the *\*subdomain\** you have chosen, e.g.

```
elmo      IN      NS      host104.dns.
          IN      NS      host105.dns.
```

This is because the domain origin is added automatically (e.g. if the zonefile is for ``utopia.dns`` then ``elmo`` becomes ``elmo.utopia.dns``)

You must also `_increment_` the serial number in the SOA record at the top of the zone file; this must be done after every zone file change of course.

Save your changes, then validate your modified zone file:

```
# named-checkzone _____.dns /etc/namedb/master/_____.dns
                        (2)                        (2)
```

If it's OK then reload:

```
# rndc reload
```

```
# tail /var/log/messages
```

That's it! Now all you need to do is to test the new subdomain by doing a normal recursive lookup for a resource record within it, for example:

```
# dig www._____._____.dns.  
          (3)
```

This test should work from anywhere in the class. The query will be first referred to your nameservers, and then you will give out a referral to the nameservers (4) and (5) which hold the data for this zone.

Step 5: Check you have received delegation for domain (8)

-----

Once you have got this far, you can check that you have received delegation for the domain (8) which you are master for. That is, make sure your slave is functioning correctly and has retrieved a copy of your zonefile; and talk to the domain owner on machine (6) to request delegation. Work along with them to ensure that any problems are ironed out. Once you have delegation, test that your new domain works correctly.

Additional steps

-----

If at any time you are being held up waiting for someone else to complete their part, then help them out.

If you have completed everything successfully, then here are some additional things you can do.

- \* Add some more resource records to the zone file for domain (8), which you control. Remember to increment the serial number.

Check that your slave has copied your modified zone file. Question: how can you check that the slave has updated: (a) given console access onto the slave machine itself, and (b) without any console access to that machine?

Check that these new resource records work, by resolving them from some other machine (one which is neither master nor slave for the zone)

- \* Find someone else who has also finished. Ask them to act as a third nameserver (second slave) for your domain, for increased resilience. Note that you'll have to change the NS records within the zone, and you'll have to change the delegation from above to be consistent.
- \* Perform the 'dig +nored' test starting from the root servers, for `www.\_\_\_\_\_.\_\_\_\_\_.dns` within your subdomain