BIND TRANSFER SECURITY

We're going to limit zone transfer of your zones so that only
your secondary/slave nameservers are allowed to request copies
of the zones.

Note: if the instructor group (for example, group 0) is providing slave
(secondary) service for your domain, then the "partner" referred below
is the instructor responsible for group 0.

ACL based security

To start with, we'll enable IP based ACLs -- on the AUTH1 host:

1. Start by editing /etc/namedb/named.conf, and in the "options" section,
   let's define who is allowed to transfer your zone.

   ```
   allow-transfer { 127.0.0.1; ::1; YOUR_OWN_IP; myslaves; };
   ```

   ... replace "YOUR_OWN_IP" with the IP of your machine :)

   Now we need to define the ACL "myslaves". To do so, AFTER the options
   section (find the '};' symbol at the end of the section), add something
   similar to this:

   (If the slave for your "MYTLD" domain is auth1.grp25, for example)

```
acl myslaves { 10.20.25.1; 10.20.X.2; }; // ACL with IP of Group25 master
```

   This means "myslaves is an ACL consisting of the IP 10.20.25.1,
   and your NSD secondary 10.20.25.2.

   NOTE: remember to enter the correct values! You must write the IP
   of the machine who is your secondary in the class - remember !

2. Restart named

   ```
   $ sudo service named restart
   ```

3. Make sure that you didn't break the zone transfer, by asking your
   slave partner to run a zone transfer against YOUR machine.

   From their server:

   ```
   $ dig @auth1.grpX.dns.nsrc.org MYTLD axfr
   ```

   Make sure that it still works.

4. Now try and ask someone else in the class whose server is NOT in the
   ACL to try the same axfr command as above.

   Q: Do they succeed ?

   Q: What do you see in the logs in /etc/namedb/log/general ?
   What do you see in the logs in /etc/namedb/log/transfers ?

TSIG KEY based security

Instead of using IP addresses, we'll now be using cryptographic keys to authenticate zone transfer -- this uses TSIG, a mechanism by which the communication between the master and slave server will be authenticated using this key.

1. Run:

```
          $ cd /tmp/
          $ sudo dnssec-keygen -a HMAC-MD5 -b 128 -n HOST mydomain.key
```

You will see something similar to this:

```
Kmydomain.key.+157+32373   (the last number will change)
```

Two files have been created:

```
          $ ls -l K*

Kmydomain.key.+157+32373.key
Kmydomain.key.+157+32373.private
```

2. View the contents of the private key

```
          $ cat Kmydomain.key.+157+32373.private
```

You will see something similar to:

```
Private-key-format: v1.2
Algorithm: 157 (HMAC_MD5)
Key: tHTRSKKrmyGmPnzNCf2IRA==
Bits: AAA=
```

... the "Key:" is the important bit here, so copy "tHTRSKKrmyGmPnzNCf2IRA==", but of course not the one above, the one in YOUR file :)

We will use this in the next steps.

3. Modify your named.conf

```
          $ cd /etc/namedb/
```

Edit the file, and change the allow-transfer statement, so that it looks like this:

```
options {
          ...
          allow-transfer { 127.0.0.1; ::1; };  // myslaves is removed!
          ...
};
```

Note: We have removed "myslaves"

Now, after the options (or at the bottom of the file), add a new declaration for the key

```
key "mydomain-key" {
        algorithm hmac-md5;
        secret "tHTRSKKrmyGmPnzNCf2IRA=="; // Your REAL key goes here!

};
```

Change the definition for your zone:

```
zone "MYTLD" {
         type master;
         file "/etc/namedb/master/mytld";

         allow-transfer { key mydomain-key; };  // <-- Add this!
};
```

As you can see above, we've added an "allow-transfer" statement allowing transfer of the zone for holders of the "mydomain-key".

Note: the allow-transfer is now placed INSIDE the zone definition, and not globally inside the options section -- BIND can control zone transfer either globally, or by zone. We could have chosen to allow transfers GLOBALLY (for all zones), by leaving the allow-transfer statement in the main "options" section.

4. Restart named

```
$ sudo service named restart
```

5. Try and make a zone transfer from ANOTER machine -- ask your neighbors to do:

```
$ dig @10.20.XX.1 MYTLD axfr
```

Look at /etc/namedb/log/general and /etc/namedb/log/transfers

Q: What do you notice ?

6. Then, ask them to try again with the key:

```
$ dig @10.20.XX.1 axfr mydomain -y mydomain-key:tHTRSKKrmyGmPnzNCf2IRA==
```

Q: what happens now ?

Check the logs again, especially /etc/namedb/log/transfers

7. On your partner's SLAVE host (your secondary - again, this may be the instructor if they are providing secondary service for your domain).

Start by asking your partner to delete their copy of your zone:

- Have them remove the zone from /etc/namedb/slave/MYTLD -- remember, this is on the machine of your SLAVE partner:

```
$ sudo rm /etc/namedb/slave/MYTLD
```

- Ask them to restart named

```
$ sudo service named restart
```

Check with them that the zone is gone AND that their server wasn't able to reload it.

Q: What do you see in the MASTER (auth1) logs (transfers and general) ?

Q: What do you see in the SLAVE logs (transfers and general) ?

8. Still on the SLAVE (if the instructor is providing secondary service, they will perform this part)

Find the statement for the zone:

```
zone "MYTLD" {
        type slave;
        masters { 10.20.XX.1; };
        file "slave/mydomain.dns";
};
```

... and add the key, and a statement to tell which key to use when talking to "10.20.XX.1" (the master):

```
key mydomain-key {
        algorithm hmac-md5;
        secret "tHTRSKKrmyGmPnzNCf2IRA==";
};
server 10.20.XX.1 {           // here you put the IP of YOUR master
        keys { mydomain-key; };
};
```

9. Restart named

```
$ sudo service named restart
```

On the SLAVE server:

Q: Is the zone "MYTLD" back in the slave/ directory ?

Q: What do you see in the MASTER (auth1) logs (transfers and general) ?

Q: What do you see in the SLAVE logs (transfers and general) ?

Can you see a general benefit from using keys instead of IP ACLs ?

Optional section if you are running a secondary yourself:

10. Now, do the same for your NSD "auth2" server

... since you have disabled IP ACLs, your AUTH NSD server is not able to get the zone!

Read the NSD manual page (man nsd.conf) if you are in doubt about how to specify the key format in NSD for zone transfers. Update update the "zone:" definition for MYTLD, so that it now uses a KEY instead of NOKEY to transfer the zone from your MASTER (auth1).

After, you will need to run "nsdc restart". Does the zone get transferred ? Remember to check the logs on the MASTER (auth1) as well!

Optional section if you are using Swatch to monitor the logs:

11. If you set up Swatch in a previous exercise, make it complain if it
   sees a forbidden zone transfer:

```
          Edit /usr/local/etc/swatch.conf, and add a new section -- remember
          to use TAB for the space at the beginning of the lines:
```

```
- - - - - - - - - - - - - - -  cut below - - - - - - - - - - - - - -


watchfor /client ([0-9.:]+)\D\d+: zone transfer '(.*)\/.XFR\/IN' denied$/
          mail=sysadm,subject=Denied AXFR for zone '$2' from $1
          threshold type=limit,count=1,seconds=600

- - - - - - - - - - - - - - -  cut above - - - - - - - - - - - - - -
```

12. Kill Swatch

```
          $ ps ax | grep swatch

          Find the process ID (the number on the left), and run kill on it:

          $ sudo kill PID_OF_SWATCH

          Restart swatch (switch to root temporarily using the sudo -s command)

          $ sudo -s
          # /usr/local/bin/swatch -c /usr/local/etc/swatch.conf --tail-file=/etc/
namedb/log/general --daemon
          # exit
          $
```

Note: Why are we telling swatch to look at the "general" log file ?

If you remember from the previous logging lab, we configured BIND to log the security category into the general channel definition. Therefore, we need to monitor the /etc/namedb/log/general file.

13. Re-run the zone transfer as in step 4 (from another machine) and see if
   you receive a mail to the arm user when that happens.:

```
          $ mutt -f /var/mail/sysadm
```

Try again 2 more times to do AXFR within a minute.

Q: How many mails did you receive? Why?