

## SECTION 9    Configure a DNS Server Using BIND

This section contains some changes concerning the commands `dnssec-keygen` and `nsupdate`.

### Objectives

1. Create a Key for Zone Transfer
2. Configure Dynamic DNS

## Objective 1 Create a Key for Zone Transfer

To create a key, use the **dnssec-keygen** command. The file name of the key is printed on the screen:

```
da51:/var/lib/named # dnssec-keygen -a HMAC-MD5 -b 128 -n HOST
zonetransfer
Kzonetransfer.+157+01389
```

The options are explained in the following table:

**Table 9-1**

Option	Description
-a HMAC-MD5	The encryption procedure used (here HMAC-MD5)
-b 128	The length of the key (in the example, 128 bits)
-n HOST	The type of key
zonetransfer	Name of key

If you use this command, two files are created in the current directory:

```
da51:/var/lib/named # ls -l K*
-rw----- 1 root root 56 Feb 21 10:39 Kzonetransfer.+157+01389.key
-rw----- 1 root root 81 Feb 21 10:39 Kzonetransfer.+157+01389.private
```

- The \*.key file contains a DNS KEY record that can be included in a zone file using the **include** statement.
- The \*.private file contains algorithm specific fields. For security reasons, this file does not have general read permission.



---

The numbers at the end of the filename will be slightly different when you run this command.

---

Both files contain the same key:

```
da51:/var/lib/named # cat Kzonetransfer.+157+01389.key
zonetransfer. IN KEY 512 3 157 KhDVspogFonWKv58rFXOWw==
da51:/var/lib/named # cat Kzonetransfer.+157+01389.private
Private-key-format: v1.2
Algorithm: 157 (HMAC_MD5)
Key: KhDVspogFonWKv58rFXOWw==
```

This key has to be included in the configuration file `/etc/named.conf` on both the master server and the slave server.

## Objective 2    Configure Dynamic DNS

If the number of zones and hosts increases, it is inconvenient to edit the zone files manually.

You can modify the resource record sources of the name server dynamically without editing and reloading files. This is called dynamic DNS. Dynamic DNS can also be used by external services like DHCP.

To allow dynamic changes, add the following line in the zone definition.

```
allow-update { 127.0.0.1; };
```

In this example, only the loopback address is used, but it is also possible to add IP addresses of other hosts.

To edit the DNS records dynamically, use the command **nsupdate**:

```
da51:~ # nsupdate  
>
```



The security tool AppArmor is installed and enabled on SUSE Linux Enterprise Server 10 by default. There is an AppArmor profile for BIND that prohibits **nsupdate** to change the BIND zone information.

AppArmor is covered in the course *SUSE Linux Enterprise Server 10 Security* (Course 3075) in detail.

Stop the AppArmor daemon by entering **rcapparmor stop** to use **nsupdate** as described in the following.

---



Before using **nsupdate**, make sure that all zone files have the correct access permissions. If they are not set properly, use the following commands:

```
chgrp -R named /var/lib/named  
chmod -R g+w /var/lib/named
```

---

Dynamic updates are stored in a journal file for the zone. This file is automatically generated by the server when the first dynamic update is performed. The name of the journal file is created by appending the extension `.jnl` to the name of the corresponding zone file. The journal file is in a binary format and should not be edited manually.

The contents of the journal file are written to the zone file every 15 minutes. When the name server is shut down, the contents of the journal file are written to the zone file, too.



Dynamic updates will change the layout of your zone files when the data is written to them. You should either use an editor or **nsupdate** to modify your zone information instead of using both tools.

---

The most important **nsupdate** options are

- **server *server* [*port*]**. Sends all dynamic update requests to the name server *server*.  
  
If no *port* number is specified, the default DNS port number 53 is used.
- **update delete *reference* [*ttl*] [*class*] [*type* [*value...*]]**. Deletes any resource records named *reference*.  
  
If the record type and value are provided, only matching resource records will be removed.  
  
The Internet class (IN) is assumed if *class* is not supplied.  
  
*ttl* is ignored. It is only allowed for compatibility.
- **update add *reference* *ttl* [*class*] *type* *value...***. Adds a new resource record with the specified *ttl* (in seconds), *class* and *value*.
- **send**. Sends the current message. This is necessary to actually execute the command. This is equivalent to entering a blank line.



---

All commands are described in the man page of `nsupdate`: **man 8 nsupdate**.

---

The following is an example of using **nsupdate**:

```
da51:~ # nsupdate
> server 127.0.0.1
> update add da13.digitalairlines.com 86400 A 10.0.0.13
>
> update delete da13.digitalairlines.com A
>
> update add 13.0.0.10.in-addr.arpa 86400 PTR da13.digitalairlines.com
>
```

This will generate messages like the following in `/var/log/messages`:

```
May 20 11:13:58 da51 named[5161]: client 127.0.0.1#32781: updating zone
'digitalairlines.com/IN': adding an RR
May 20 11:13:58 da51 named[5161]: journal file
master/digitalairlines.com.zone.jnl does not exist, creating it
May 20 11:13:58 da51 named[5161]: zone digitalairlines.com/IN: sending
notifies (serial 2005051903)
May 20 11:21:50 da51 named[5161]: client 127.0.0.1#32783: updating zone
'0.0.10.in-addr.arpa/IN': adding an RR
May 20 11:21:50 da51 named[5161]: journal file master/10.0.0.zone.jnl does
not exist, creating it
May 20 11:21:50 da51 named[5161]: zone 0.0.10.in-addr.arpa/IN: sending
notifies (serial 2005051902)
```

The journal files will be created automatically:

```
da51:/var/lib/named # dir master/
total 16
drwxrwxr-x  2 root  named 200 May 20 11:21 .
drwxrwxr-x  9 root  named 408 May 20 10:40 ..
-rw-rw-r--  1 root  named 463 May 19 12:06 10.0.0.zone
-rw-r--r--  1 named named 814 May 20 11:21 10.0.0.zone.jnl
-rw-rw-r--  1 root  named 440 May 19 14:51 digitalairlines.com.zone
-rw-r--r--  1 named named 794 May 20 11:13 digitalairlines.com.zone.jnl
```

Press **Ctrl + D** or enter **quit** to quit `nsupdate`.

In the following table, the most important record types are listed:

**Table 9-2**

<b>Record Type</b>	<b>Meaning</b>	<b>Value</b>
SOA	Start of Authority	Parameter for the domain
NS	DNS server	Name of a DNS server for this domain
MX	Mail exchanger	Name and priority of a mail server for this domain
A	Address	IP address of the computer
PTR	Pointer	Name of the computer
CNAME	Canonical name	Alias name for the computer

Alternatively, you can specify a file containing the needed commands. An ASCII file (called **updates**) with the following content delivers the same result as the interactive DNS update as shown above:

```
da51:~ # cat updates
update delete da7.digitalairlines.com
update add da8.digitalairlines.com 84600 A 10.0.0.8
da51:~ # nsupdate -v -k Kdhcp-dns.+157+23165.key updates
da51:~ #
```

## Summary

Objective	Summary
1. Create a Key for Zone Transfer	<p>Generate the key by using the <b>dnssec-keygen</b> command.</p> <p>The *.key file contains a DNS KEY record that can be included in a zone file using the include statement.</p> <p>The *.private file contains algorithm specific fields. For security reasons, this file does not have general read permission.</p> <p>This key has to be included in the configuration file /etc/named.conf on both the master server and the slave server.</p>
2. Configure Dynamic DNS	<p>You can to modify the resource records of BIND dynamically and without editing and reloading files.</p> <p>Dynamic DNS can also be used by external services like DHCP.</p> <p>To manipulate the DNS records dynamically, use the <b>nsupdate</b> command.</p>

---