

**NAME**

zkt-signer — Secure DNS zone signing tool

**SYNOPSIS**

```
zkt-signer [-L|--logfile file] [-V|--view view] [-c file] [-fhnr] [-v [-v]] -N named.conf [zone ...]
zkt-signer [-L|--logfile file] [-V|--view view] [-c file] [-fhnr] [-v [-v]] [-D directory] [zone ...]
zkt-signer [-L|--logfile file] [-V|--view view] [-c file] [-fhnr] [-v [-v]] -o origin [zonefile]
```

**DESCRIPTION**

The *zkt-signer* command is a wrapper around *dnssec-signzone(8)* and *dnssec-keygen(8)* to sign a zone and manage the necessary zone keys. It is able to increment the serial number before signing the zone and can trigger *named(8)* to reload the signed zone file. The command controls several secure zones and, if started in regular intervals via *cron(8)*, can do all that stuff automatically.

In the most useful usage scenario the command will be called with option **-N** to read the secure zones out of the given *named.conf* file. If you have a configuration file with views, you have to use option **-V** viewname or **--view** viewname to specify the name of the view. Alternately you could link the executable file to a second name like *zkt-signer-viewname* and use that command to specify the name of the view. All master zone statements will be scanned for filenames ending with ".signed". These zones will be checked if the necessary zone- and key signing keys are existent and fresh enough to be used in the signing process. If one or more out-dated keys are found, new keying material will be generated via the *dnssec-keygen(8)* command and the old keys will be marked as depreciated. So the command do anything needed for a zone key rollover as defined by [2].

If the resigning interval is reached or any new key must be announced, the serial number of the zone will be incremented and the *dnssec-signzone(8)* command will be evoked to sign the zone. After that, if the option **-r** is given, the *rndc(8)* command will be called to reload the zone on the nameserver.

In the second form of the command it is possible to specify a directory tree with the option **-D** *dir*. Every secure zone found in a subdirectory below *dir* will be signed. However, it is also possible to reduce the signing to those zones given as arguments.

In the last form of the command, the functionality is more or less the same as the *dnssec-signzone(8)* command. The parameter specifies the zone file name and the option **-o** takes the name of the zone.

If neither **-N** nor **-D** nor **-o** is given, then the default directory specified in the *dnssec.conf* file by the parameter *zonedir* will be used as top level directory.

**OPTIONS**

**-L** *file/dir*, **--logfile=***file/dir*

Specify the name of a log file or a directory where logfiles are created with a name like *zkt-YYYY-MM-DDThhmmssZ.log*. If the argument is not an absolute path name and a zone directory is specified in the config file, this will be prepended to the given name. This option is also settable in the *dnssec.conf* file via the parameter **LogFile**.

The default is no file logging, but error logging to syslog with facility **USER** at level **ERROR** is enabled by default. These parameters are settable via the config file parameter **SyslogFacility**, **SyslogLevel**, **LogFile** and **LogLevel**.

The additional parameter **VerboseLog** specifies the verbosity (0|1|2) of messages that will be logged with level **DEBUG** to file and syslog.

**-V** *view*, **--view=***view*

Try to read the default configuration out of a file named *dnssec-<view>.conf*. Instead of specifying the **-V** or **--view** option every time, it is also possible to create a hard- or softlink to the executable file with an additional name like *zkt-signer-<view>*.

- c file, --config=file**  
Read configuration values out of the specified file. Otherwise the default config file is read or build-in defaults will be used.
- O optstr, --config-option=optstr**  
Set any config file option via the commandline. Several config file options can be specified via the argument string but have to be delimited by semicolon (or newline).
- f, --force**  
Force a resigning of the zone, regardless if the resigning interval is reached or new keys must be announced.
- n, --noexec**  
Don't execute the *dnssec-signzone(8)* command. Currently this option is of very limited usage.
- r, --reload**  
Reload the zone via *rndc(8)* after successful signing. In a production environment it is recommended to use this option to be sure that a freshly signed zone will be immediately propagated. However, that's only feasible if named runs on the signing machine, which is not recommended.
- v, --verbose**  
Verbose mode (recommended). A second **-v** will be a little more verbose.
- h, --help**  
Print out the online help.

## SAMPLE USAGE

- ```
zkt-signer -N /var/named/named.conf -r -v -v
  Sign all secure zones found in the named.conf file and, if necessary, trigger a reload of the zone.
  Print some explanatory remarks on stdout.
```
- ```
zkt-signer -D zonedir/example.net. -f -v -v
  Force the signing of the zone found in the directory zonedir/example.net. Do not reload the zone.
```
- ```
zkt-signer -D zonedir -f -v -v example.net.
  Same as above.
```
- ```
zkt-signer -f -v -v example.net.
  Same as above if the dnssec.conf file contains the path of the parent directory of the example.net
  zone.
```
- ```
zkt-signer -f -v -v -o example.net. zone.db
  Same as above if we are in the directory containing the example.net files.
```
- ```
zkt-signer --config-option='ResignInterval 1d; Sigvalidity 28h; \
  ZSK_lifetime 2d;' -v -v -o example.net. zone.db
  Sign the example.net zone but override some config file values with parameters given on the
  commandline.
```

## Zone setup and initial preparation

Create a separate directory for every secure zone.

This is useful because there are many additional files needed to secure a zone. Besides the zone file (*zone.db*), there is a signed zone file (*zone.db.signed*), a minimum of four files containing the keying material, a file called *dnskey.db* with the current used keys, and the *dsset-* and *keyset-*files created by the *dnssec-signzone(8)* command. So in summary there is a minimum of nine files used per secure zone. For every additional key there are two extra files and every delegated subzone creates also two or three files.

Name the directory just like the zone.

That's only needed if you want to use the `zkt-signer` command in directory mode (`-D`). Then the name of the zone will be parsed out of the directory name.

Change the name of the zone file to `zone.db`

Otherwise you have to set the name via the `dnssec.conf` parameter `zonefile`, or you have to use the option `-o` to name the zone and specify the zone file as argument.

Add the name of the signed zonefile to the `named.conf` file

The filename is the name of the zone file with the extension `.signed`. Create an empty file with the name `zonefile.signed` in the zone directory.

Include the keyfile in the zone.

The name of the keyfile is settable by the `dnssec.conf` parameter `keyfile`. The default is `dnskey.db`

```

.
.
.
      IN NS          ns1.example.net.
      IN NS          ns2.example.net.
$INCLUDE dnskey.db
.
.
.

```

You can also run `zkt-conf(8)` in the secure zone directory to do this. Try

```
$ zkt-conf -w zone.db
```

Control the format of the SOA-Record

For automatic incrementation of the serial number, the SOA-Record must be formatted, so that the serial number is on a single line and left justified in a field of at least 10 spaces!

```
@ IN SOA ns1.example.net. hostmaster.example.net. (
      60          ; Serial          43200 ; Refresh
      1800       ; Retry           2W   ; Expire
      7200       ; Minimum

```

If you use BIND version 9.4 or later and use the unixtime format for the serial number (which is the default since ZKT-1.0) than this is not necessary. See also the parameter `Serialformat` in `dnssec.conf`.

Try to sign the zone

If the current working directory is the directory of the zone `example.net`, use the command

```
$ zkt-signer -D .. -v -v example.net
or
$ zkt-signer -o example.net.
```

to create the initial keying material and a signed zone file. Then try to load the file on the name server.

## ENVIRONMENT VARIABLES

**ZKT\_CONFFILE**

Specifies the name of the default global configuration files.

## FILES

*/var/named/dnssec.conf*

Built-in default global configuration file. The name of the default global config file is settable via the environment variable `ZKT_CONFFILE`. Use `zkt-conf(8)` with option `-w` or `dnssec-zkt(8)` with option `-Z` to create an initial config file.

*/var/named/dnssec-<view>.conf*

View specific global configuration file.

*./dnssec.conf*

Local configuration file. The file contains typically only the diff to the global site wide config file. Use for example

```
$ zkt-conf -w -l -O "key_ttl: 5d"
```

to create a local config file with a different key ttl time.

*dnskey.db*

The file contains the currently used key and zone signing keys. It will be created by *dnssec-signer(8)*. The name of the file is settable via the dnssec configuration file (parameter *keyfile*).

*zone.db*

This is the zone file. The name of the file is settable via the dnssec configuration file (parameter *zonefile*).

**BUGS**

The named.conf parser is a bit rudimental and not very well tested.

**AUTHORS**

The man page is written by Holger Zuleger and Mans Nilsson

**COPYRIGHT**

Copyright (c) 2005 – 2010 by Holger Zuleger. Licensed under the BSD Licence. There is NO warranty; not even for MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE.

**SEE ALSO**

*dnssec-keygen(8)*, *dnssec-signzone(8)*, *rndc(8)*, *named.conf(5)*, *zkt-conf(8)*, *zkt-ls(8)*, *zkt-keygen(8)*  
RFC4033, RFC4034, RFC4035

[1] DNSSEC HOWTO Tutorial by Olaf Kolkman, RIPE NCC  
([http://www.nlnetlabs.nl/dnssec\\_howto/](http://www.nlnetlabs.nl/dnssec_howto/))

[2] RFC4641 "DNSSEC Operational Practices" by Miek Gieben and Olaf Kolkman  
(<http://www.ietf.org/rfc/rfc4641.txt>)