Notes

This reference guide provides the information needed for a system administrator familiar with BIND and DNS to add DNSSEC to servers and clients. The BIND DNSSEC Guide, also published by ISC, has full details, explanations, and backstories. See

```
https://isc.org/downloads/bind/dnssec/
```

Not every registrar supports DNSSEC, and the means of getting DS record information to your registrar varies. Read your registrar's documentation.

Usage: We use `example.com` as the domain name in all examples. Substitute your own. The words "zone" and "domain" are synonyms for our purposes and we intend no difference in their meaning.

```
Shell commands are shown with a
$ prefix:
$ cd working/keys
```

A line ending with "\" means the next line is a continuation of it:

```
$ chmod g+r \
  *.private
```

OS variations

The directories, UIDs, and GIDs used by BIND will vary somewhat from OS to OS. We use ***red italics*** to note places where the specifics of your OS might vary from ours. Substitute the UID and GID used by BIND on your installation.

Configuration

Overall configuration

DNSSEC uses cryptography. Verify compliance with your national or corporate rules for use of cryptography.

Server requirements

BIND 9.10 or higher installed.

Registrar or parent zone supports DNSSEC via DS records.

Server's network environment supports TCP and large UDP for DNS ports (port 53) including EDNS0.

If server is virtual it must have an adequate randomness (entropy) source. See

```
https://wikipedia.org/wiki//dev/random
```

System requirements

Directory holding master zone files must be `rw` for group ***bind***.

BIND configuration

Into named.conf options{}:

```
key-directory "keys";
dnssec-validation auto;
```

Recommendations

The computer on which you edit and sign zones (signing master) ought not be its own resolver. Let that server be authoritative-only and use some other server as a resolver.

Keep that signing master server as secure and locked down as possible and use it for DNS-related services only.

Testing

BIND tools for DNSSEC

```
$ delv @server [type] example.com
```

Downloadable DNSSEC tools

```
https://www.dnssec-tools.org
```

Web tools to check client resolver

```
http://www.dnssec-failed.org
http://dnssectest.sidnlabs.nl
```

Web tools to check zones and signing

```
http://dnssec-debugger.\
  verisignlabs.com
http://dnsviz.net
https://www.zonemaster.fr
http://zonecheck.org
```

Additional resources

ISC documentation

ISC's Knowledge Base is

```
https://kb.isc.org
```

ISC's primary documentation of DNSSEC and its usage in BIND is the BIND DNSSEC Guide; see

```
https://isc.org/downloads/bind/dnssec/
```

The full BIND 9 Administrator's Reference Manual is in the cur/doc directory at

```
ftp://ftp.isc.org/isc/bind9/ \
  cur/9.10/doc/arm/Bv9ARM.html
```

The Knowledge Base article about it is

```
https://kb.isc.org/article/AA-01031/
```

Websites

```
https://wikipedia.org/wiki/DNSSEC
```

DNSSEC for BIND
Quick Reference Guide

for Unix-like systems

BIND 9.10 • 2015

DNSSEC provides a means by which a client of the DNS can be confident that replies to its queries are authentic in origin and content. It does not encrypt queries or replies.

Specifically, DNSSEC protocols enable a client to validate a response from a server using data from trusted sources external to that server. The queried server and its parent zone must both contain DNSSEC data, and the client's resolver must have software that can use that data to perform the validation.

To benefit fully from the information in this reference guide you should read all of it before you get started. Also you should be somewhat familiar with configuring and operating BIND on a Unix or Unix-like system.

This Guide is available in PDF format in the ISC Knowledge Base at https://kb.isc.org/ and in HTML format at www.isc.org

www.isc.org

Internet Systems Consortium

Creating and using keys

Preparation

There is no "official" place to store BIND DNSSEC keys. There aren't even popular customs. But current and future automation of BIND DNSSEC processing assumes certain locations and names, so we specify directories per `auto-dnssec`.

Directories and protections

BIND needs to be able to add keys to key directories, read the private keys that it generates, and write signed zone files into the directory holding the unsigned files. Do this by ensuring that key directories and master zone directories are in group bind with group-read/write permission, and that private keys are in group bind with group read permission.

```
$ cd working
$ chgrp bind keys master
$ chmod g+w keys master
```

Put your keys into a separate directory for each domain. Keys for example.com should go in ***working***/`keys/example.com` and that directory should be group-writeable by group "***bind***".

Generating keys

```
$ cd working/keys
$ mkdir example.com
$ chgrp bind example.com
$ cd example.com
$ dnssec-keygen -a RSASHA256 \
     -b 1024 example.com
$ dnssec-keygen -a RSASHA256 \
     -b 2048 -f KSK example.com
$ chgrp bind *
$ chmod g+r *.private
```

Signing a domain

Preparation

Create keys for this domain using our instructions.

BIND configuration

A zone that exists but is not signed will have an entry in the BIND configuration. To that entry, add

```
key-directory "keys/example.com";
inline-signing yes;
auto-dnssec maintain;
```

Now tell BIND to reload that zone:

```
$ rndc reload example.com
```

Verifying signature exists

Depending on the speed of your server and the amount of entropy available to BIND, it might take several minutes to complete the zone signing. The simplest way to verify that the signing has completed is to look at a zone transfer of the newly-signed zone and ensure that the last record is signed:

```
$ dig @localhost AXFR example.com
```

Notifying the parent

After signing has completed, the final step in signing a domain is to register the signatures with its parent. Make a DS record from the zone-signing key:

```
$ cd working/keys
$ dnssec-dsfromkey \
    `grep -l zone-signing *key`
```

That will display two DS records, created with different algorithms. Use the longer one (it will have " 8 2 " in it) and submit it to your registrar. Follow your registrar's instructions.

Ongoing maintenance

Overview

DNSSEC-signed domains need to change keys occasionally. When keys change, everything must be re-signed. We suggest updating Zone Signing Keys yearly, and Key Signing Keys as often as you update your SSL certificates.

ZSK rollover

To use new ZSK keys beginning 1 January of year *yyyy*, on 1 December, set the old key to expire, create its successor, and make it readable by the running `bind`:

```
$ cd working/keys/example.com
$ dnssec-settime \
  -I yyyy0101 -D yyyy0201 \
  basename-of-ZSK
$ dnssec-keygen \
  -S basename-of-ZSK
$ chgrp bind *
$ chmod g+r *.private
```

The basename of a key is the filename with the suffix ("`.key`" or "`.private`") removed. You can tell which key is the ZSK by looking inside it with a text editor.

KSK rollover

The KSK rollover process is very similar, but requires interaction with the registrar. Mark the existing KSK for expiration, create its successor and make it readable by userid `bind`. Make a new DS record and upload it to the registrar alongside the existing DS record. A month after the rollover date, log on to the registrar account and delete the old DS record.

The KSK rollover process is described in section 7.2.2 of the BIND DNSSEC Guide.

NSEC and NSEC3

Overview

DNSSEC validates data by providing verifiable cryptographic signatures of that data. That technique can't be used to validate the absence of data, because there is nothing to sign. NSEC and NSEC3 records provide something to be signed to authenticate the absence of data.

The difference between NSEC and NSEC3 is briefly explained in Wikipedia; search for "NSEC3". To better understand the difference and make an informed choice, we suggest you visit `www.internetsociety.org` and type "nsec3" into its search box.

`in-addr` zones, and those permitting open zone transfers, will not benefit from NSEC3 and should use NSEC.

Converting to and from NSEC3

Our BIND domain-signing instructions produce NSEC records. To convert a signed zone to NSEC3, do this:

```
$ rndc signing -nsec3param \
    1 0 10 auto example.com
```

To convert from NSEC3 back to NSEC, do this:

```
$ rndc signing -nsec3param \
    none example.com
```