OpenDNSSEC Key Rollover Guide

W. Matthijs Mekking

October 16, 2015

Abstract

OpenDNSSEC is a policy-based zone signer that automates the process of keeping track of DNSSEC [2], [4], [3] keys and the signing of zones. The goal of the project is to make DNSSEC easy to deploy. A previous guide [5] helps you set up OpenDNSSEC and provide simple steps for adding and removing zones. This document provides guidelines for doing key rollovers.

Contents

1 ZSK rollovers

Key rollovers in OpenDNSSEC are automated. This means you do not have to do anything to perform scheduled rollovers. In the Key And Signing Policy (KASP) file **kasp.xml** there is set a lifetime for keys and rollovers will happen periodically and automatically:

```
<ZSK>
    ...
    <Lifetime>P90D</Lifetime>
    ...
</ZSK>
```

Above excerpt shows the default lifetime of Zone Signing Keys (ZSKs).

1.1 Manual key rollover

If you want to do a manual rollover that is possible too. This simple command will start a new rollover in a timely manner for the ZSK of example.com:

```
$ ods-ksmutil key rollover --zone example.com --keytype ZSK
```

You can also roll all the keys for zones which share the same policy. This command for example will roll all KSKs for zones that have the default policy:

```
$ ods-ksmutil key rollover --policy default --keytype KSK
```

2 KSK rollovers

KSK rollovers are slightly different than ZSK rollovers because they need interaction with the parent zone operator.

First, the DNSKEY record for the new key is added to the zone. Your key will be in publish mode for some hours so that others, especially validating resolvers, will know about this key. After a suitable interval, the new DS record can be submitted to the parent; at the same time the old DS record can also be removed from the parent.

When it is time for this to happen a message with log-level ”info” will be sent to syslog looking something like:

```
Jun  5 07:52:14 server ods-enforcerd: WARNING: New KSK has reached the  \
  ready state; please submit the DS for example and use ods-ksmutil key
  ds-seen when the DS appears in the DNS.
```

The key is in the ready state, as can also be witnessed with **key list**:

```
$ ods-ksmutil key list
Keys:
Zone:                           Keytype:      State:    Date of next transition:
example.com                     ZSK           active    2015-06-23 17:24:00
example.com                     KSK           active    2015-06-04 12:52:09
example.com                     KSK           ready     waiting for ds-seen
```

Note that if at this point you do nothing, your zones will remain secure! The current KSK will remain active until you interact with OpenDNSSEC. The successor KSK will be in your zone but it will only become active if you issue the **ds-seen** command. But first make sure that the successor DS record is submitted.

2.1 Action: Submit the DS

You can export the keys that are ready to be submitted to the parent with:

```
$ ods-ksmutil key export -z example.com -e READY [-d]
```

The **key export** command by default exports KSKs. We want only the keys that are in the ready state, indicated with the **-e** option. The **-d** gives us the derived DS records. If your parent expects a DNSKEY record, leave out the **-d** option.

This step can be (semi-)automated by configuring a command in the **DelegationSignerSubmitCommand** tag (in **conf.xml**), for example with the simple mail script provided in the OpenDNSSEC git repository [1]:

```
<DelegationSignerSubmitCommand>
    /usr/bin/simple-dnskey-mailer.sh
</DelegationSignerSubmitCommand>
```

This will mail you when new keys need to be submitted to the parent. Of course you can use your own, more sophisticated scripts here. OpenDNSSEC feeds your program or script on stdin with the current set of DNSKEYs that we want to have in the parent as DS RRs.

2.2 Action: Seen the DS

When the DS records have been seen in DNS then this can be communicated to OpenDNSSEC with the **ds-seen** command:

```
$ ods-ksmutil key ds-seen -z example.com -x 42112
```

With **-x** you provide the keytag of the new DNSKEY (this keytag is also used in the corresponding DS record).

You are done. If you list the keys again you will see that the new KSK has become active and the predecessor is retired:

```
$ ods-ksmutil key list
Keys:
Zone:                           Keytype:      State:    Date of next transition:
example.com                     ZSK           active    2015-06-23 17:24:00
example.com                     KSK           retire    2015-06-08 16:15:11
example.com                     KSK           active    2020-06-06 11:28:32
```

The date of next transition will tell you when the retired KSK will automatically be removed from your zone.

2.3 Retire KSKs

If the DS records were not swapped, i.e. the old DS was left in the parent when the new one was added, then the **–no-retire** flag should be added to the **ds-seen** command above. Then, at some later time, the old key can be retired with the command:

```
$ ods-ksmutil key ksk-retire -z example.com -x 42112
```

This will retire the specific key (provided the key is active, and the action will not leave the zone without any active keys).

3 Feedback

If you have feedback or still run into trouble, you can subscribe and mail to `opendnssec-user@lists.opendnssec.org`.

4 Acknowledgements

The OpenDNSSEC team provided most of the text in this guide. This text is also available in a more verbose version on the OpenDNSSEC Wiki [6]. Furthermore, I would like to express my gratitude to Jan Žorž for proof-reading this guide.

References

[1] *OpenDNSSEC Source Code on GitHub*. `https://github.com/opendnssec`.

[2] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. *DNS Security Introduction and Requirements*. RFC 4033 (Proposed Standard), March 2005. `http://www.ietf.org/rfc/rfc4033.txt`.

[3] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. *Protocol Modifications for the DNS Security Extensions*. RFC 4035 (Proposed Standard), March 2005. `http://www.ietf.org/rfc/rfc4035.txt`, (Updated by RFC 4470).

[4] R. Arends, R. Austein, M. Larson, D. Massey, and S. Rose. *Resource Records for the DNS Security Extensions*. RFC 4034 (Proposed Standard), March 2005. `http://www.ietf.org/rfc/rfc4034.txt`, (Updated by RFC 4470).

[5] W. Matthijs Mekking. *OpenDNSSEC Initial Deployment Guide*, November 2014. `https://pletterpet.nl/static/publications/opendnssec-start-guide.pdf`.

[6] OpenDNSSEC Documentation. `https://wiki.opendnssec.org/`.