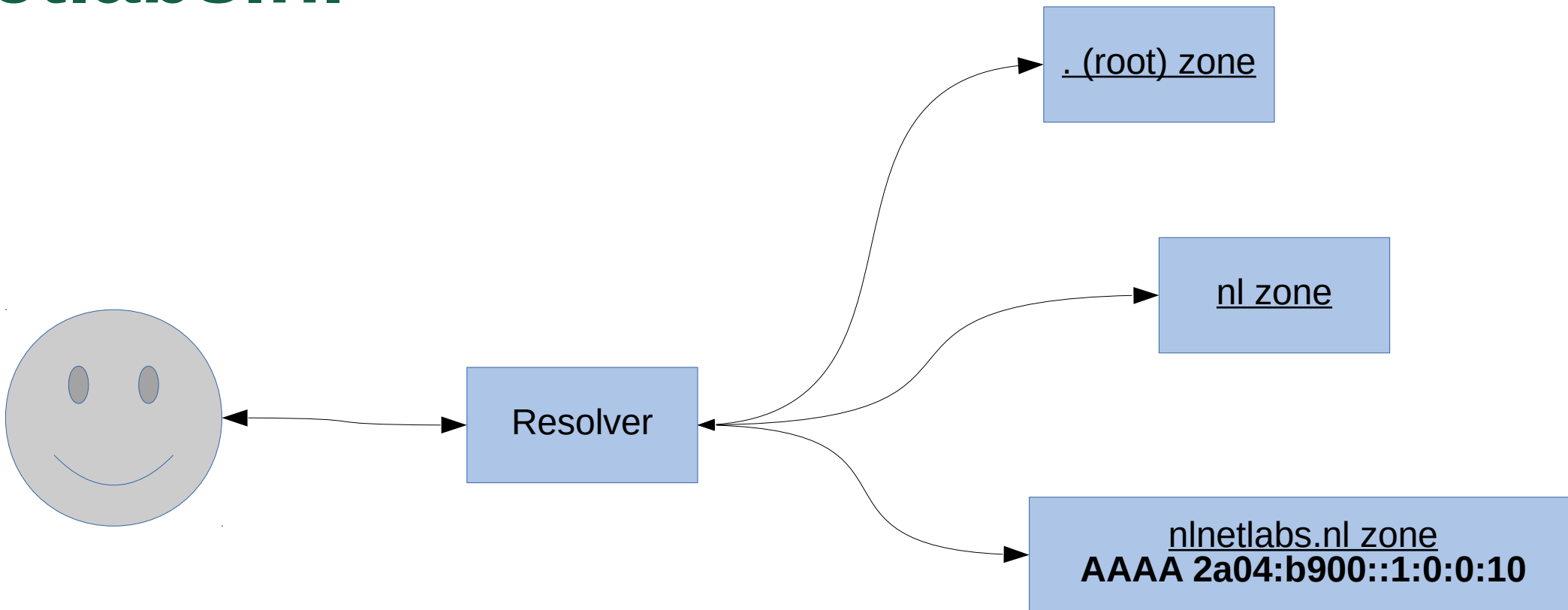


# DNSSEC: Rollin', Rollin', Rollin'

Ralph Dolmans  
ralph@nlnetlabs.nl

Martin Hoffmann  
martin@nlnetlabs.nl

# DNS – resolving IPv6 address for nlnetlabs.nl



# Answer with DNSSEC signature

```
:: QUESTION SECTION:
```

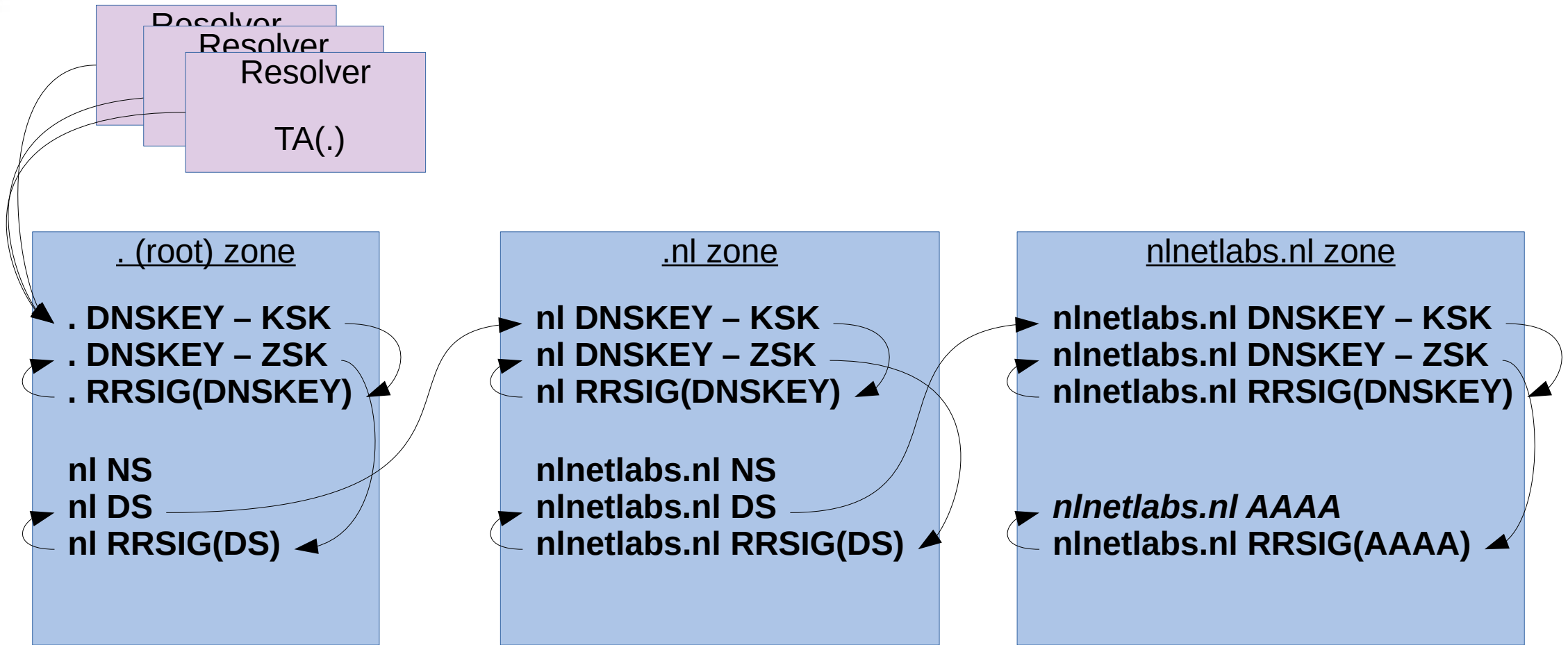
```
;nlnetlabs.nl. IN AAAA
```

```
:: ANSWER SECTION:
```

```
nlnetlabs.nl. 10200 IN AAAA 2a04:b900::1:0:0:10
```

```
nlnetlabs.nl. 10200 IN RRSIG AAAA 8 2 10200 20180529005003 20180501005003  
42393 nlnetlabs.nl. HcAuIC0d5eCYZYwsoEDymzQOBRR5SmhJUZww6n[...]
```

# DNSSEC authentication chain



# Root KSK roll

- Original timeline:
  - 26<sup>th</sup> October 2016 – new key created
    - Keytag 20326
  - 11<sup>th</sup> July 2017 – new key published in root zone
    - Start in-band update mechanism
  - 11<sup>th</sup> October 2017 – use new key to sign root DNSKEY rrset
    - Validation will fail for users that only have KSK2010 as TA

# Root KSK roll

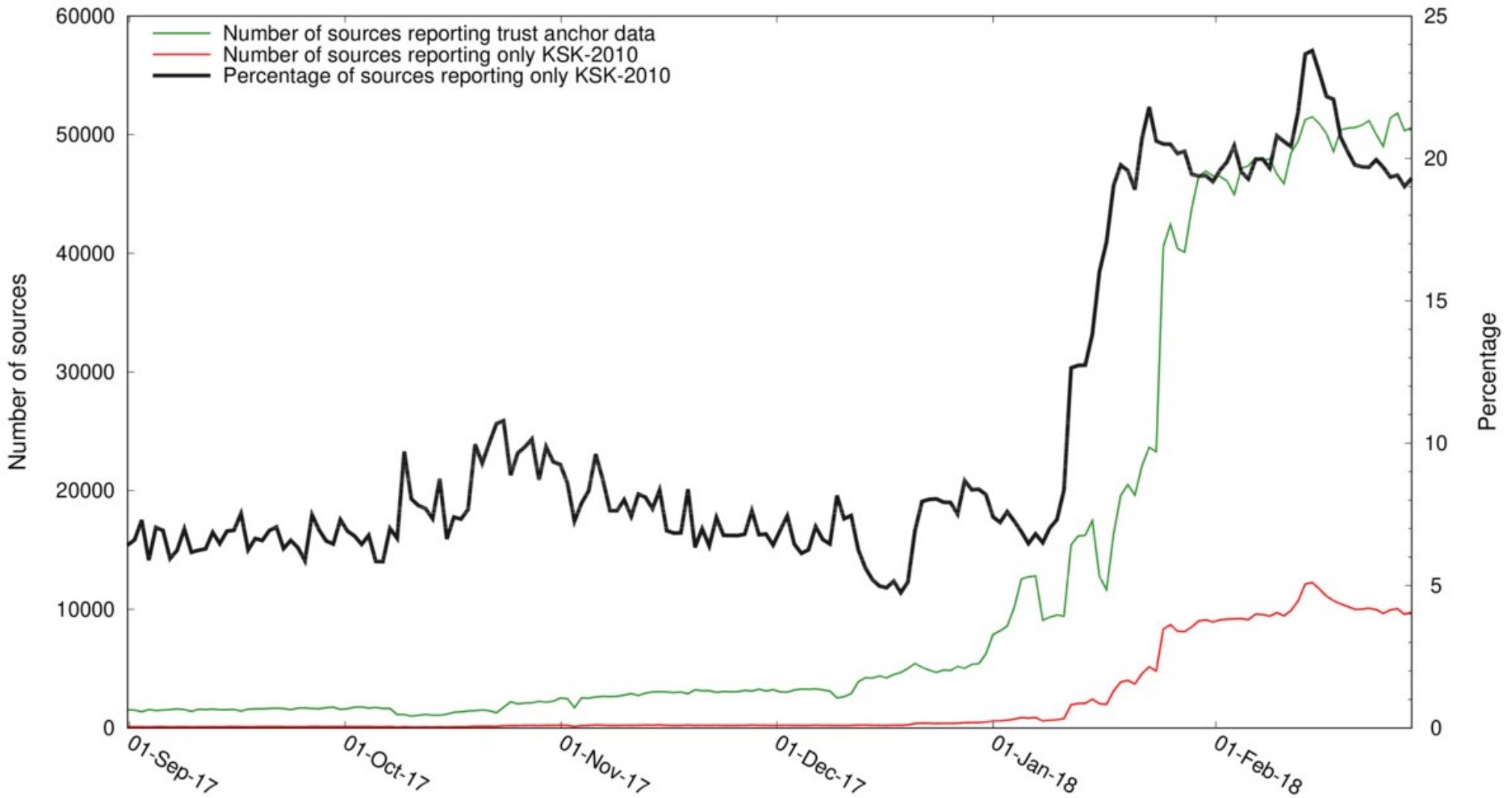
- 172800 IN DNSKEY 257 3 8 (  
AwEAAaz/tAm8yTn4Mfeh5eyl96WSVexTBAvkMgJzkKTO  
iW1vklbzxeF3+/4RgWOq7HrxRixHIFlExOLAJr5emLvN  
7SWXgnLh4+B5xQlNVz8Og8kvArMtNROxVQuCaSnIDdD5  
LKyWbRd2n9WGe2R8PzgCmr3EgVLrjyBxWezF0jLHwVN8  
efS3rCj/EWgvlWgb9tarpVUDK/b58Da+sqqls3eNbu7  
pr+eoZG+SrDK6nWeL3c6H5Apxz7LjVc1uTlIdslXxuOLY  
A4/ilBmSVIzuDWfdRUfhHdY6+cn8HFRm+2hM8AnXGXws  
9555KrUB5qihylGa8subX2Nn6UwNR1AkUTV74bU=  
) ; KSK; alg = RSASHA256; key id = **20326**

# Updating DNSSEC Trust Anchors

- Manually
- Outside DNS protocol (e.g. software update)
- In DNS protocol → RFC5011 (“Automated Updates of DNS Security (DNSSEC) Trust Anchors”)
  - Using trust in existing TA to start trusting another key
  - New root TA in resolvers on 11<sup>th</sup> August 2017

<http://root-trust-anchor-reports.research.icann.org/rfc8145-by-day-all.png>

RFC8145 Trust Anchor Reports for All Root Servers







# Root key sentinel (draft-ietf-dnsop-kskroll-sentinel)

- User impact on root key roll
  - root-key-sentinel-is-ta-20326
  - root-key-sentinel-not-ta-20326
  - Some bogus domain
- In Unbound, BIND and Knot resolver
- No results available yet

# RFC 5011 in Open Source Resolvers

## Yes

- Unbound
- Bind
- Knot Resolver

## No

- PowerDNS Recursor
- Dnsmasq
- systemd-resolved

# RFC 5011 Compliance Test




deckard

Test harness for DNS software.

★ Star 5 HTTPS `https://gitlab.labs.nic.cz/knot/deckard.git` 📄

Files (6.9 MB) Commits (459) Branches (19) Tag (1) Readme BSD 2-clause "Simplified" License CI configuration

master deckard History Find file 🔄

 Merge branch 'draft-ietf-dnsop-kskroll-sentinel-01' into 'master' ... 9c7c6313 📄  
Petr Špaček authored 6 days ago

Name	Last commit	Last update
📁 ci	Mac OS: remove readlink calls	2 months ago
📁 contrib	contrib: update libswrap to avoid problems with CMak...	6 months ago

# Test Scenarios

- Happy Path
- Un-publish before signing
- Roll-back after signing
- Revocation of old key
- Early re-introduction of old key
- Un-revoked old key
- Late re-introduction of old key
- Missing new key
- Non-writeable state directory
- Resolver restarts
- Restarts with non-writable state directory
- Late installation with old key only
- Late installation with both keys
- Post-roll installation with old key only
- Post-roll installation with new key only
- Happy path with forwarding to a non-validating resolver
- Happy path while forwarding to a non-DNSSEC resolver

# Unbound

## Versions

First production release	1.0.0	2008-05-20
RFC 5011 since	1.4.0	2009-11-26
Latest release	1.7.1	2018-05-03
Current Debian stable	1.6.0	2016-12-15

# Unbound

## Findings

### Late Installation

- Only trusts the new trust anchor after the 30 days' hold down.
- **Even if the new trust anchor is provided on installation.**
- Fixed in 1.6.5 (2017-08-21).

### Re-introduction of Old Key

- accepts the old key after remove and add hold-downs.

# Unbound

## Operational

RFC 5011 needs to be explicitly enabled

- *trust-anchor-file* v. *auto-trust-anchor-file*

Non-writable state directory

- initially: logs error and carries on until next restart
- 1.5.4 (2015-07-09): logs error and stops

# Unbound

## Do I have the new KSK?

auto-trust-anchor-file

```
# cat /var/lib/unbound/root.key
; autotrust trust anchor file
;;id: . 1
;;last_queried: 1525860779 ;;Wed May 9 12:12:59 2018
;;last_success: 1525860779 ;;Wed May 9 12:12:59 2018
;;next_probe_time: 1525901662 ;;Wed May 9 23:34:22 2018
;;query_failed: 0
;;query_interval: 43200
;;retry_time: 8640
. 172800 IN DNSKEY 257 3 8
AwEAAaz/tAm8yTn4Mfeh5eyI96WSVextTBAvkMgJzkKT0iw1vkIbzxef3+/4RgW0q7HrxRixHlFlEx0LAJf...mLvN7SWXgnLh4+B5xQ
lNVz80g8kvArMtNR0xVQUcASnIDdd5LKyWbRd2n9WGe2R8PzgCmr3EgVLRjyBxwezF0jLHwVN8efS3rCj/EwgVIWgb9tarPVUDK/
b58Da+sqqls3eNbuV7pr+eoZG+SrDK6nWeL3c6H5Apxz7LjVC1uTIdSIXxuOLYA4/
ilBmSVIzudWfdRUfhHdY6+cn8HFRm+2hM8AnXGXws9555KrUB5qihylGa8subX2Nn6UwNR1AkUTV74bU= ;{id = 20326 (ksk),
size = 2048b} ;;state=2 [ VALID ] ;;count=0 ;;lastchange=1525860756 ;;Wed May 9 12:12:36 2018
. 172800 IN DNSKEY 257 3 8
AwEAAagAIKlVZrpC6Ia7gEzah0R+9w29euxhJhVVLOy0bSEW008gcCjFFVOUTf6v58fLjwBd0YI0EzrAcQqBGCzh/
RStIo08g0NfnfL2MTJRkxoXbfDaUeVPQuYEhg37NZWAJQ9VnMVDXP/VHL496M/QZxkjf5/
Efucp2gaDX6RS6CXpoY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3LQpzw5h0A2hzCTMjJPJ8LbqF6dsV6DoBQzgul0sGI
cGOYl70yQdXfZ57reISQageu+ipAdTTJ25AsRTAoub80NGcLmqrAmRLKBP1dfwhYB4N7knNnulqQA+Uk1ihz0= ;{id = 19036
(ksk), size = 2048b} ;;state=2 [ VALID ] ;;count=0 ;;lastchange=1525860756 ;;Wed May 9 12:12:36
2018
```

new KSK

old KSK



# Bind 9

## Versions

First production release	9.0.0	2000-09-16	
First still usable release*	9.6.2	2010-03-01	*supporting RSASHA256
RFC 5011 since	9.7.0	2010-02-16	
RFC 5011 working since*	9.7.1	2010-06-17	*according to our test
Latest release	9.12.1	2018-03-14	
Current Debian stable	9.10.3	2015-09-16	

# Bind 9

## Findings

### Re-introduction of Old Key

- initially: accepts the old key even before the hold-downs have passed.
- 9.10.2 (2015-02-25): never accepts revoked keys again

### Post-roll Installation with Old Key

- initially: resolver goes insecure instead of bogus
- 9.10.4 (2016-04-28): fixed

# Bind 9

## Operational

RFC 5011 needs to be explicitly enabled

- *trusted-keys* v. *managed-keys*

Non-writable state directory

- initially: repeatedly logs and carries on; upon restart goes insecure
- 9.8.1 (2011-08-31): logs error once and carries on; upon restart goes bogus

# Bind 9

## Do I have the new KSK?

```
# rndc secroots
# cat /var/cache/bind/named.secroots
09-May-2018 11:01:55.792
```

```
Start view _default
```

**new KSK**

```
./RSASHA256/20326 ; managed
./RSASHA256/19036 ; managed
```

**It does 5011**

**old KSK**

# Knot Resolver

## Versions

First production release	1.0.0	2016-06-21
RFC 5011 since	1.0.0	*
Latest release	2.3.0	2018-04-23
Current Debian stable	1.2.0	2017-01-25

\* in our tests first working in 1.2.0, faulty in 1.2.2, then working again in 1.5.0

# Knot Resolver

## Findings\*

### Re-introduction of old key

- accepts the old key after remove and add hold-downs

### Late installation with old key only

- 1.2.0: accepts the new key during the add hold-down but not after
- 1.2.5: accepts the new key for one day only

### Revocation of old key

- 1.5.0: accepts the old key for only day after removal from DNSKEY record.

\* last test was on 1.2.0

<https://www.nlnetlabs.nl/>

# Knot Resolver

## Operational

Trust anchors are always update via RFC 5011. \o/

### Non-writable state directory

- initially: stops at start with *permission* denied
- 1.5.1: same message but keeps running and goes bogus one day late
- 2.0.0: stops at start again.
- Resolver restarts
  - 1.2.0: add hold-down restart with every restart if trust anchor is kept but config directory recreated, otherwise bogus three days after key roll.
  - 2.0.0: fixed

# Knot Resolver

## Do I have the new KSK?

```
# cat /etc/knot-resolver/root.key
      172800 DNSKEY 257 3 8
AwEAAagAIKlVZrpC6Ia7gEzah0R+9W29euxhJhVVL0yQbSEW008gcCjFFVQUTf6v58f
LjwBd0YI0EzrAc0qBGCzh/
RStIo08g0NfnfL2MTJRkxoXbfDaUeVPQuYEhg37NZWAJQ9VnMVDxP/VHL496M/
QZxkjf5/
Efucp2gaDX6RS6CXpoY68LsvPVjR0ZSwzz1apAzvN9dlzEheX7ICJBBtuA6G3LQpzW5
h0A2hzCTMjJPJ8LbqF6dsV6DoB0zgu10sGicG0Y170yQdXfZ57relSQageu+ipAdTTJ
25AsRTAoub80NGcLmqrAmRLKBP1dfwhYB4N7knNnulqQxA+Uk1ihz0= ; Valid: ;
KeyTag: 19036
      172800 DNSKEY 257 3 8
AwEAAaz/tAm8yTn4Mfeh5eyI96WSVextBAvkMgJzkKT0iW1vkIbzxeF3+/4RgW0q7Hr
xRixH1F1Ex0LAJr5emLvN7SWXgnLh4+B5xQ1NVz80g8kvArMtNR0xVQuCaSnIDdD5LK
ywbRd2n9WGe2R8PzgCmr3EgVLRjyBxWezF0jLHwVN8efS3rCj/
EwgvIWgb9tarpVUDK/
b58Da+sqqls3eNbuV7pr+eoZG+SrDK6nWeL3c6H5Apxz7LjVc1uTIdsIXxu0LYA4/
i1BmSVIzuDwfdRUFhHdY6+cn8HFRm+2hM8AnXGXws9555KrUB5qihy1Ga8subX2Nn6U
wNR1AkUTV74bU= ; Valid: ; KeyTag: 20326
```



# PowerDNS Recursor

## Do I have the new KSK?

```
# rec_control get-tas
Configured Trust Anchors:
.
  19036 8 2 49aac11d7b6f6446702e54a1607371607a1a41855200fd2ce1
cdde32f24e8fb5
  20326 8 2 e06d44b80b8f1d39a95c0b0d7c65d08458e880409bbc683457
104237c7f8ec8d
```

---

20326 missing?

- upgrade to at least 4.0.5
- <https://doc.powerdns.com/recursor/dnssec.html#trust-anchors>

# Dnsmasq

## Do I have the new KSK?

- Configuration includes somewhere:

```
dnssec
```

```
trust-anchor=., 19036, 8, 2, 49AAC1...
```

```
trust-anchor=., 20326, 8, 2, E06D44...
```

# systemd-resolved

## Do I have the new KSK?

- Root trust anchors are baked into the source.
- New KSK added in version 233 (2017-03-12), probably backported in various stable distributions.
- `src/resolve/resolved-dns-trust-anchor.c`

```
static int dns_trust_anchor_add_builtin_positive(DnsTrustAnchor *d) {  
    /* ... */  
    r = add_root_ksk(answer, key, 20326, /* ... */);  
    /* ... */  
}
```

# What happens next?

ICANN proposed to proceed to roll on October 11 this year.

- Public comment period ended April 1.
- 20 comments, mostly in favor of going ahead.
- Final plan should be published any day now.