Root KSK Roll Delay Update

And a discussion about Roll Frequency
DNSSEC-Deployment

Roy Arends, Principal Research Scientist

1 November 2017

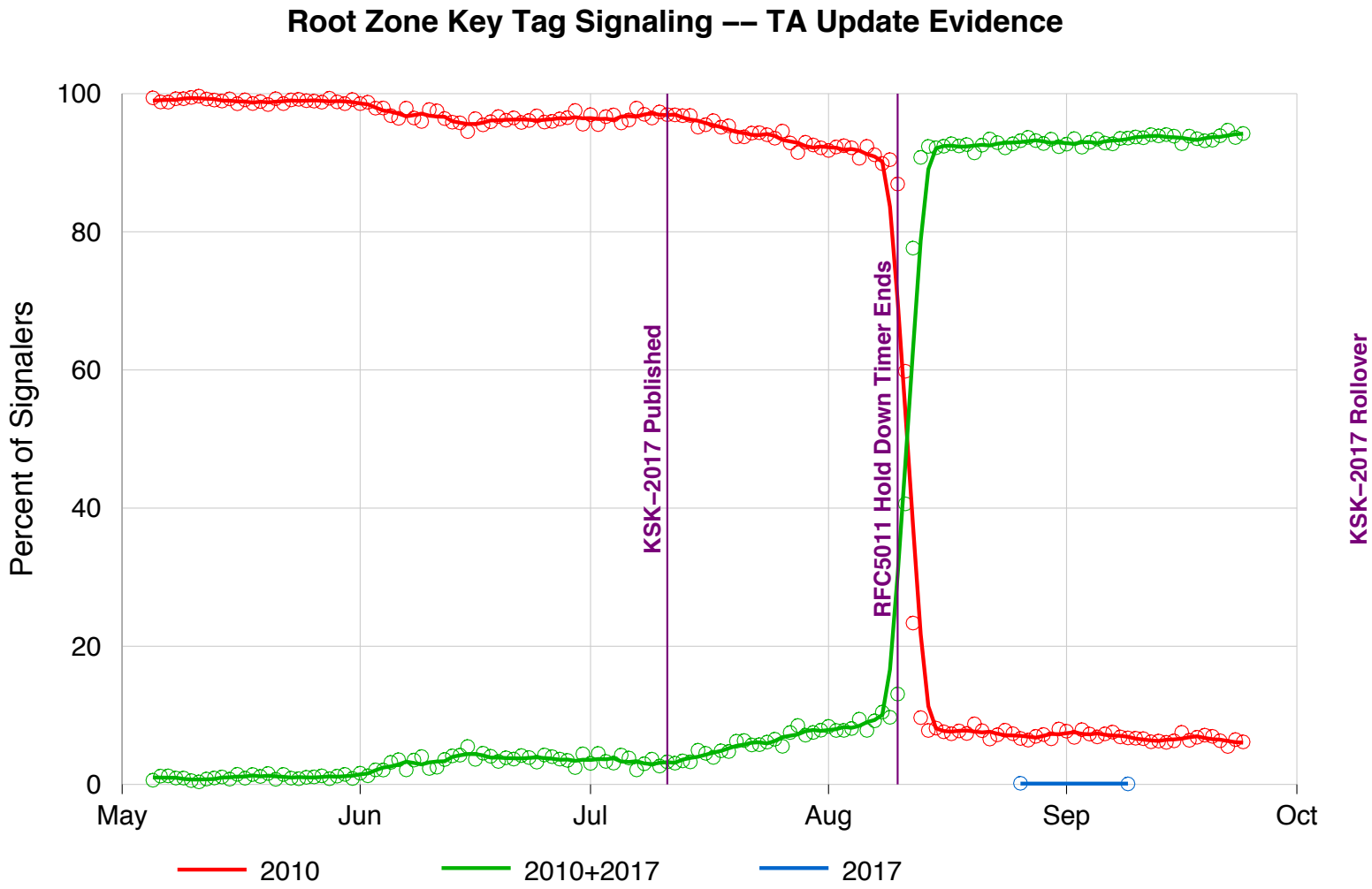
Background

- When you validate DNSSEC signed DNS records, you need a Trust Anchor.
  - A Trust Anchor is a Public Key.
- Public Keys should not live forever.
- These Trust Anchors probably should be periodically renewed (rolled).
  - You can do this automatically or manually.
- However, there was no way for us (ICANN) to check if you have the right key configured.
- Therefore, a multi-year design and outreach effort ensued:
  - Design-team, blogs, outreach, presentations in various venues, plans, vendors and governments were contacted, etc., etc.

The Process

- **11 July 2017**: Introduce the new KSK-2017.
  - Monitor if there are fundamental changes in root-server traffic
  - If not, continue, else fall back.
- **10 August 2017**: “30 day hold-down period ends”
  - Monitor if there are fundamental changes in root-server traffic.
  - If not, continue, else fall back.
- **19 September 2017**: DNSKEY Response size increased due to standard ZSK roll
  - Monitor if there are fundamental changes in root-server traffic.
  - If not, continue, else fall back.

The timeline in a graph



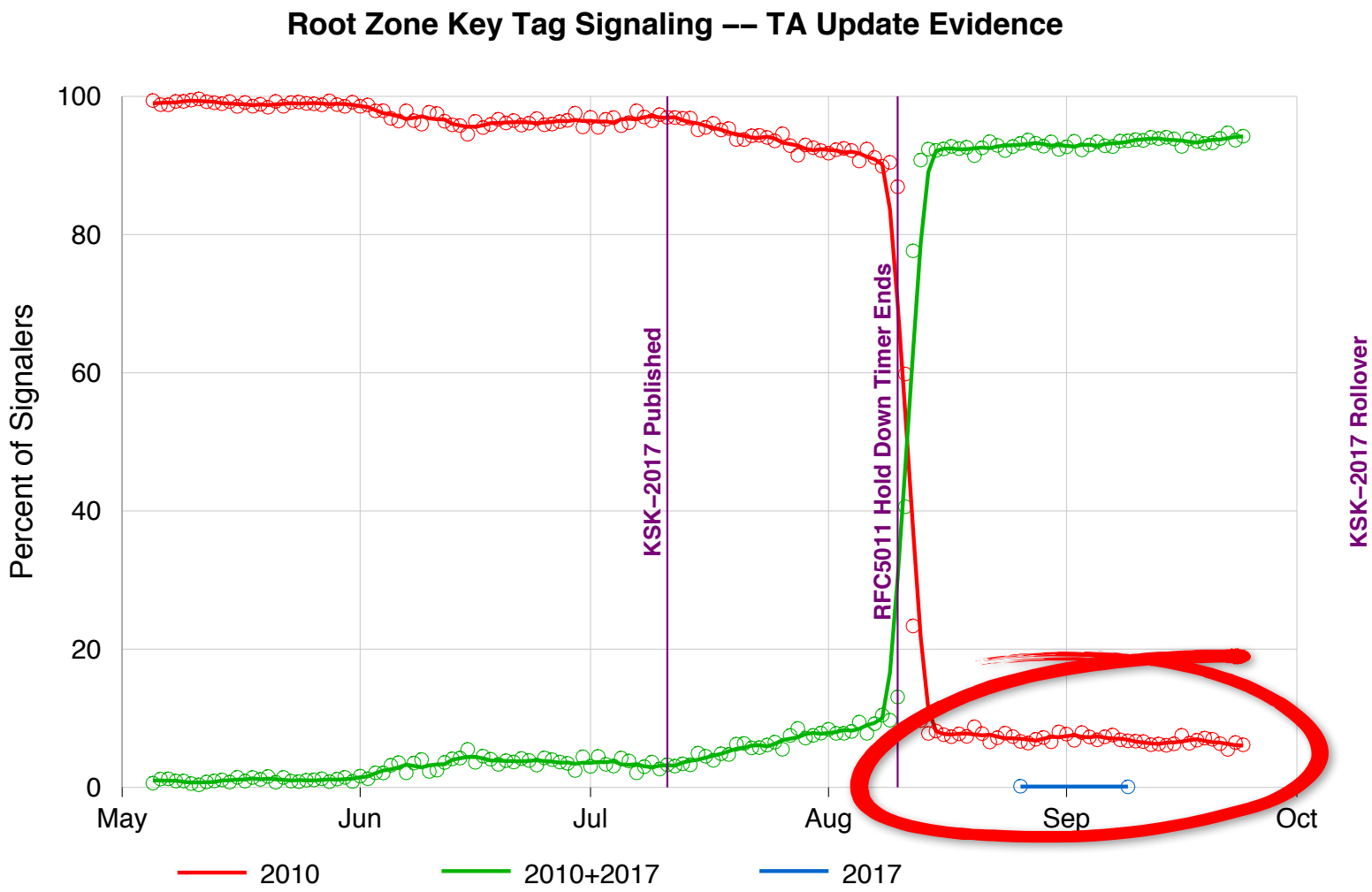
Who has KSK-2017 configured as a trust anchor?

- Until very recently, there was no way to know which trust anchors validators have configured
- *Signaling Trust Anchor Knowledge in DNS Security Extensions (DNSSEC)* is a recent protocol extension that can provide that information
  - Reports trust anchor key tags via EDNS option or DNS query
  - Published as RFC 8145 (April 2017)
- Implementations
  - BIND 9.11 starting with 9.11.0b3 (28 July 2016)
  - BIND 9.10 starting with 9.10.5b1 (11 January 2017)
  - Unbound 1.6.4 (27 June 2017)
  - On by default in BIND (since 28 July 2016) and in Unbound since version 1.6.7 (10 October 2017)
  - No other known implementations

Looking for key tag signaling

- RFC 8145 is so new and validator support so limited that the root KSK roll project team did not expect to get enough data to help with the first root KSK Roll.
  - On average, there are 4.2 Million unique addresses sending queries to root-servers.
  - Given typical deployment curves, it was assumed the dataset would be too small to statistically represent all validating resolvers.
- However…
  - Before the introduction of KSK-2017, RFC8145-able resolvers would send KSK-2010 only.
  - After the hold down period of 30 days, RFC8145-able resolvers would send both KSK-2010 and KSK-2017.
  - Duane Wessels (Verisign, co-author of 8145) started looking at A & J root traffic for this signaling

Wait. What?


Root Zone Key Tag Signaling –– TA Update Evidence
Percent of Signalers
100
80
60
40
20
0
May
Jun
Jul
Aug
Sep
Oct
KSK–2017 Published
RFC5011 Hold Down Timer Ends
KSK–2017 Rollover
2010
2010+2017
2017
Verisign Public
powered by VERISIGN
17

Further analysis by OCTO Research

- ICANN OCTO Research did an analysis similar to Duane's
  - Analyzed query data from B, D, F and L root servers
  - For the entire month of September and October (until the 24th)
- Results:
  - Total number of unique addresses reporting key tag data: **27,084** (out of 4.2 million, 0.57%)
  - Total number that only ever reports KSK-2010: **1631**
  - **6.02% of reporting validators were not ready for the KSK roll on 11 October 2017**
  - Non-zero percentage of reporting validators were announcing **only** KSK-2017 (?!)
- Analysis is complicated
  - Dynamic resolver IPs make the situation look worse by inflating true number of sources
  - Resolvers behind forwarders make the situation look better as they obscure multiple validators behind the forwarder

Why do validators report just KSK-2010?

- Multiple reasons suspected or confirmed:
  1. BIND reports trust anchors even if not validating
  2. Old configurations pre-dating automatic update support
     - E.g., BIND's *trusted-keys* instead of *managed-keys* or *dnssec-validation auto*
  3. Bugs in automatic update or key tag signaling support
     - E.g., announce key tags even if DNSSEC not enabled (DO=0)
  4. Operator error
     - E.g., Docker container keeps booting up with only KSK-2010 and starts 5011 all over again
- We always knew old configurations would be an issue but never had objective data until now
- We worried bugs and operator error were possible but didn't have evidence until now
- Analysis is ongoing
  - Hired a contractor to try to figure out reasons for misconfiguration

Back to the plan and process

- 19 September 2017: DNSKEY Response size increased due to standard ZSK roll
  - Monitor if there are fundamental changes in root-server traffic.
  - If not, continue, else fall back.
- We had received Verisign's report and corroborated it with our own data.
- From the Operational Plan:

  "*The Root Zone Management Partners might also decide to extend any phase for additional quarters. For example, if new information indicates that the next phase may lead to complications, the current phase would be prolonged. This is referred to as an extend scenario.*"
- 27 September 2017: "Extend" scenario kicks in
  - ICANN Announces that the root KSK Rollover is delayed

Issues

- We do not know how representative the set of validators reporting key tag data is compared to the set of all validators
- Validators != end users (or “end systems”), and the impact on end users is what is most important
  - The design team recognized this
- Determining number of end users/systems for a given resolver is hard
  - APNIC’s Google Ad experiment platform-based data will help
- Mitigation is hard
  - We’ve already had a multi-year campaign to reach operators
  - Implementation-specific problems don’t make the problem easier

Next Steps

- We postponed the root KSK roll until we can gather more information and understand the situation better
  - The delay will be at least one quarter
  - We have not yet determined how many quarters to delay
- We will at least partially mitigate
  - Contractor hired to try to track down the 500 resolvers based on IP addresses and understand why misconfiguration is occurring
  - Data collection continues
- We'll need to re-engage/re-tune the communications plan
  - Maybe "PLEASE DO **NOT** REMOVE KSK-2017!!"?

Root KSK Roll Frequency Discussion

Roy Arends, Principal Research Scientist

1 November 2017

Background

- We started to use the current (old) KSK in 2010
  - We (now) plan to start using the new KSK in 2018
- Observations:
  - Some folks are adamant about manually configuring the new trust anchor
  - Some folks are relying on RFC 5011, some are using non-5011 mechanisms
  - Validators may have some configuration issues, e.g.
    - Running on read-only partitions
    - Using the wrong configuration stanza
  - Bugs exist
  - Trust-anchor telemetry is far from optimal

Schools of Thought

- **Roll frequently**
  - Theory: systems will only work if you exercise them
    - Do it now before we see lots of DNSSEC deployment
  - Pros: shakes out bugs
  - Cons: people will turn off DNSSEC
- **Roll infrequently**
  - Theory: if it ain't broke, don't fix it
    - Rely on out-of-band mechanisms to update trust anchors
  - Pros: minimizes changes for resolver operators
  - Cons: if we need to roll, we must depend on operating system updates (etc.)
- Note: Automated rollover won't (currently) help in the face of key compromise/loss
  - RFC 5011 requires an uncompromised key to accept the new key
  - Need a "standby key" (implies a new KMF) and hope it doesn't (also) get compromised/lost

What is the frequency?

- **DNS technical community went with “Roll Frequently”**
- The point of waiting 5 years after the initial roll was to ensure RFC 5011 deployment
  - All major resolver vendors now support it
- Defining a lower and upper bound for frequency
  - 5 years is what is defined in our practice statement: use this as upper bound
    - i.e. not wait longer to roll the key
  - The absolute lower bound is 3 months due to the Key Signing Key ceremonies.
    - This is the roll frequency of the ZSK.
    - If we want a higher frequency, we need to fundamentally change the processes **and validators**

Effects of rolling every 3 months

- Rolling every 3 months means the 3 states are collapsed into a single 3 month slot
  - Introduce key C
  - Start signing with key B
  - Revoke key A
- Due to this collapse, we can't roll back: Once key revoked, it can't be unrevoked.
  - Entire process needs re-design
- Packet size becomes very large.
  - Worst case is 1 ZSKs with a ZSK signature, and 3 KSKs with 2 KSK signatures
  - Minimal DNSKEY Response size is 1986 bytes
    - This breaks several known limits
- Manual or semi automatic deployments will have to update every 3 months

Effects of rolling every 6 months

- This doesn’t have the response size problem.
  - i.e. it is not worse than the current key roll
- This still collapses two stages into one.
  - The remaining two stages are
    - Introducing the new key
    - Revoking the old key and using the new key
- Due to the ”collapse” there is no way to “roll back”
  - We need to re-design the entire process

Effects of rolling every 9 months

- This doesn’t have the response size problem.
- This can use the current design of the roll plan.
- This is the highest frequency that does not incur fundamental changes to the design
  - However, this might still be awkward with regards to timing.
    - Every year, the time slot moves a quarter forward, every four years, there will be two rolls in one year.
    - Likely not optimal for operators due to lack of predictability

Effects of rolling every year

- This doesn’t have the response size problem.
- This can use the current design of the roll plan.
- This is the second highest frequency that does not incur fundamental changes to the design
- Every year, the time frame that the key is known and the date of the roll will be exactly the same, e.g.,
  - Apr 11th introduce the new key
  - Oct 11th stop using the old key
  - Jan 11th revoke the old key
- More predictable, hence likely better for operators

Effects of rolling after more than a year

- No significant difference from the effects of rolling each year
  - Potentially less annoyance of resolver operators
- If on a year boundary (i.e., every 2 years, every 3 years), rolls can be predictable
  - Probably preferable for operators

Engage with ICANN – Thank You and Questions

One World, One Internet

Visit us at **icann.org**

@icann

facebook.com/icannorg

youtube.com/icannnews

flickr.com/icann

linkedin/company/icann

slideshare/icannpresentations

soundcloud/icann