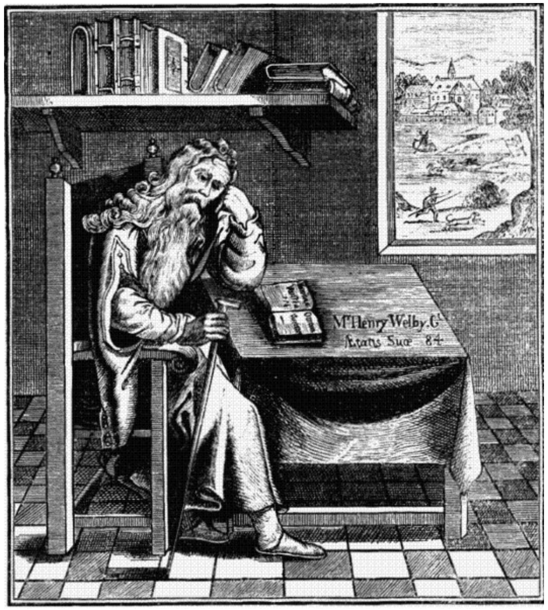
Measuring the KSK Roll

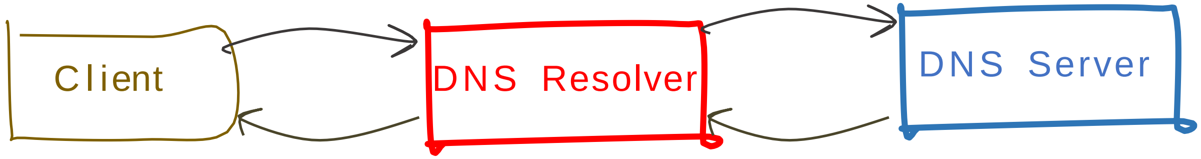
Geoff Huston
APNIC Labs

KSK Roll Measurement Objective

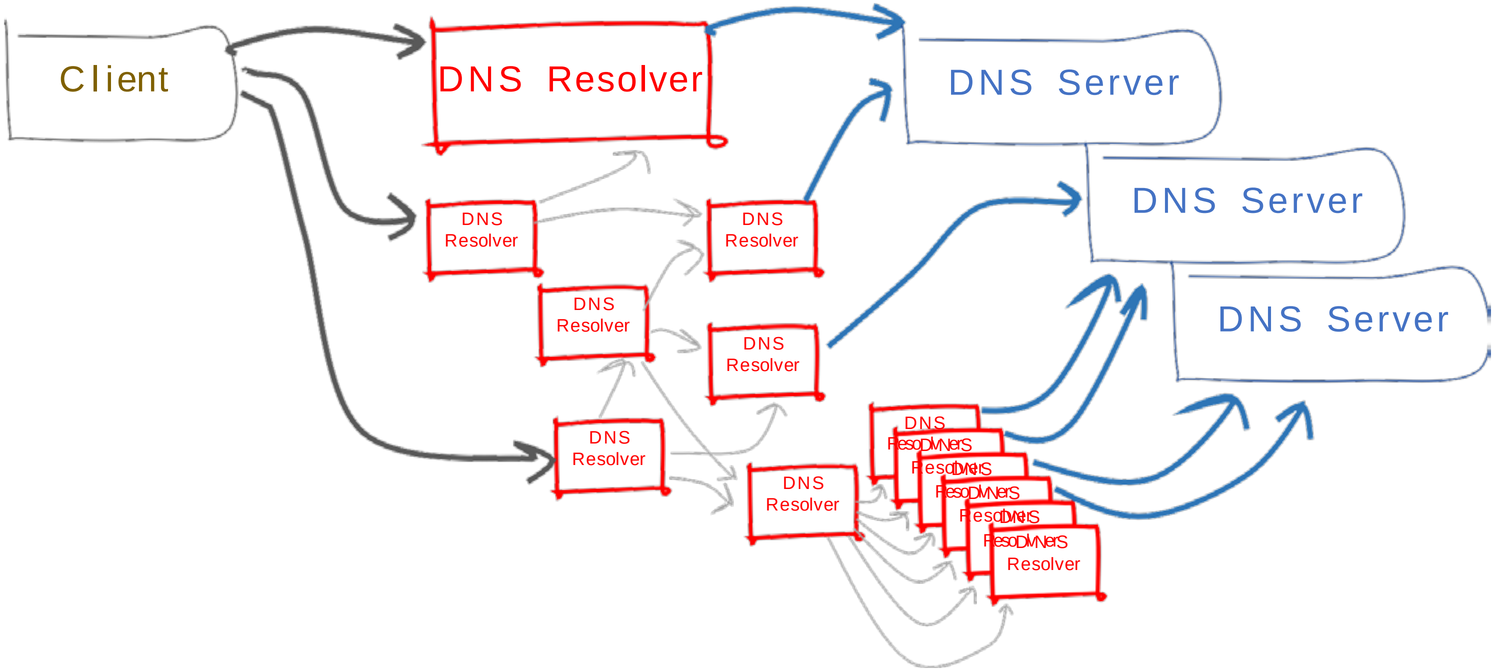
What number of users are at risk of being impacted by the KSK Roll?



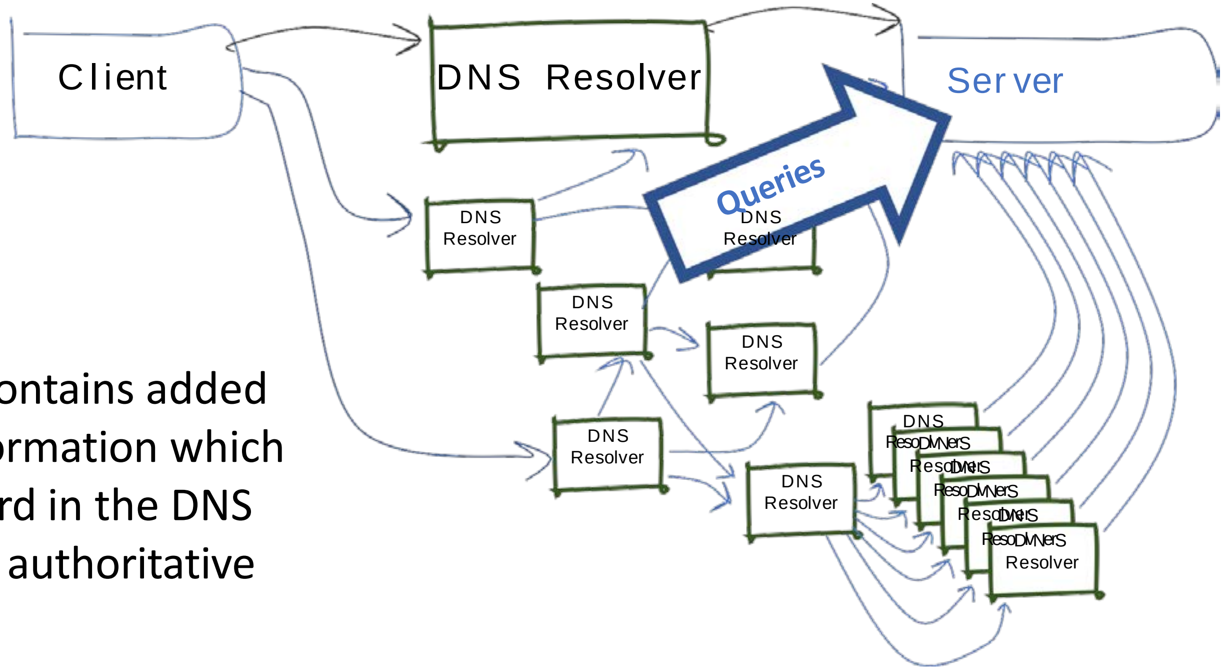
What we would like the DNS to be



What we suspect is more like theDNS

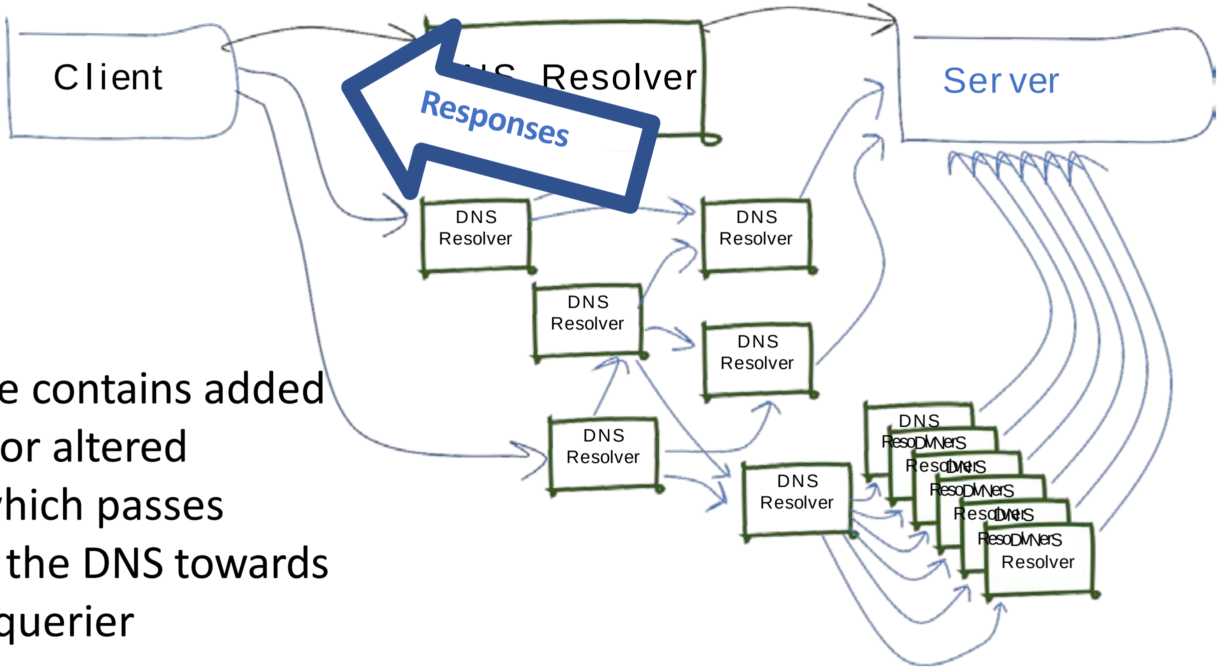


Signalling via Queries



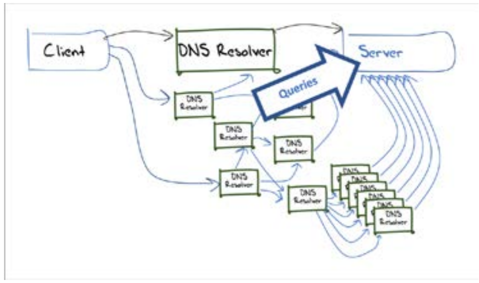
The query contains added resolver information which passes inward in the DNS towards the authoritative server(s)

Signalling via Responses



The response contains added information or altered behavious which passes backward in the DNS towards the original querier

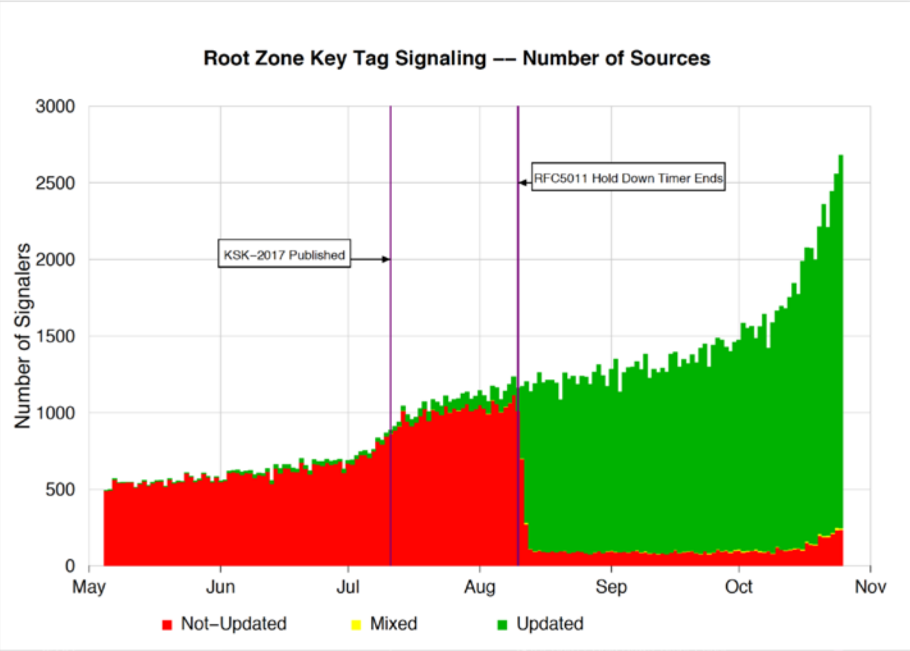
Measuring Resolvers via RFC 8145 Signaling

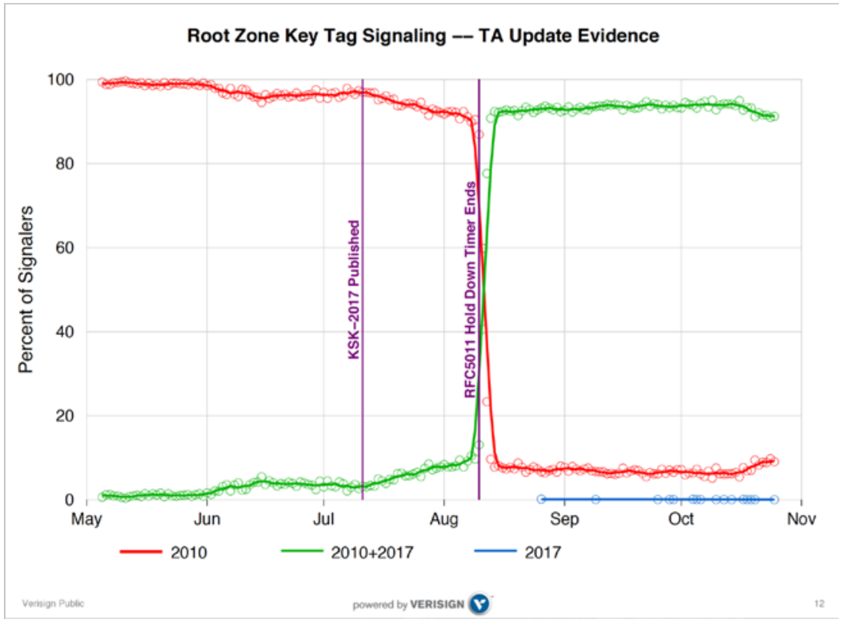


Getting resolvers to report on their local trusted key state

- A change to resolver behavior that requires deployment of new resolver code
- Resolvers that support the RFC 8145 signal mechanism periodically include the key tag of their locally trusted keys into a query directed towards the root servers

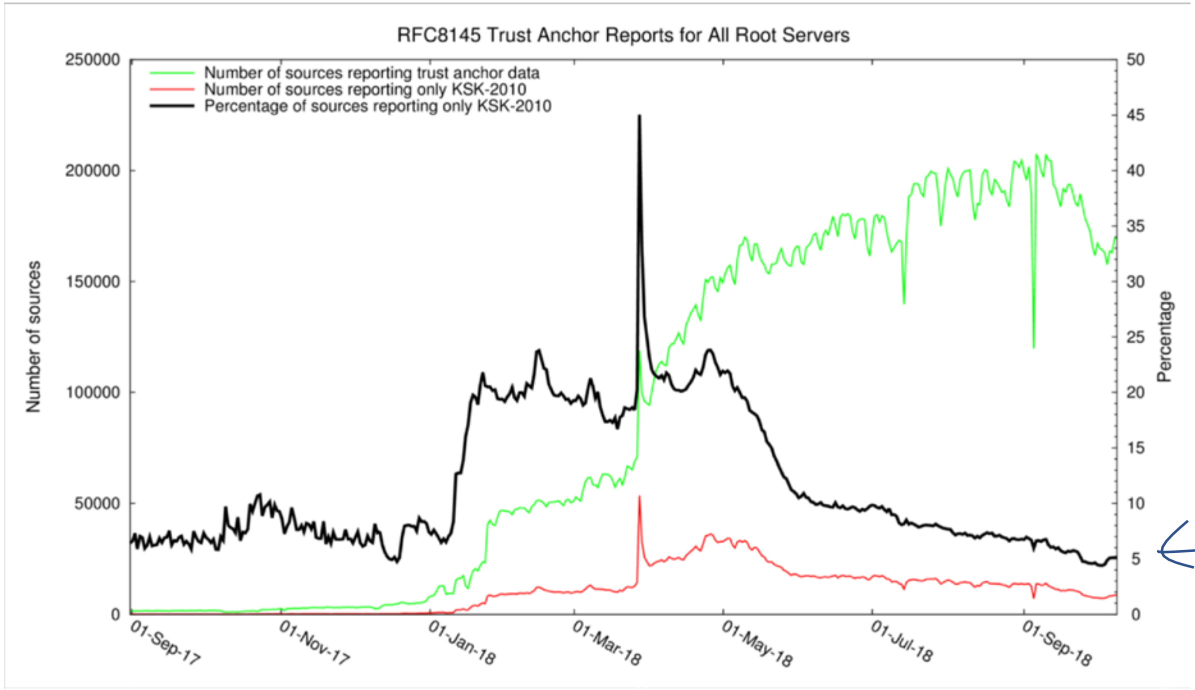
What did we see at (some) roots?


Root Zone Key Tag Signaling –– Number of Sources


Root Zone Key Tag Signaling –– TA Update Evidence

Duane Wessels VeriSign RFC 8145 Signaling Trust Anchor Knowledge In DNS Security Extensions
Presentation to DNSSEC Workshop @ ICANN 60 – 1 Nov 2017
https://schd.ws/hosted_files/icann60abudhabi2017/ea/Duane%20Wessels-VeriSign-RFC%208145-Signaling%20Trust%20Anchor%20Knowledge%20in%20DNS%20Security%20Extensions.pdf

12 months of RFC8145 signalling



Yes, with just a few days to go this mechanism was still reporting 5% ‘breakage’

http://root-trust-anchor-reports.research.icann.org

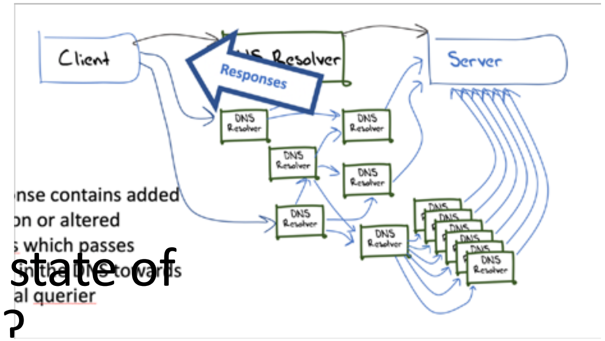
What is this saying?

- Its clear that there is some residual set of resolvers that are signalling that they have not yet learned to trust the new KSK key
- But its not clear if:
  - This is an accurate signal about the state of this resolver
  - This is an accurate signal about the identity of this resolver
  - How many users sit ‘behind’ this resolver
  - Whether these uses rely solely on this resolver, or if they also have alternate resolvers that they can use
  - What proportion of all users are affected

Why?

- Because the DNS does not disclose the antecedents of a query
    - If A forwards a query to B, who queries a Root Server then if the query contains an implicit signal (as in this case) then it appears that B is querying, not A
    - At no time is the user made visible in the referred query
- Because caching
    - If A and B both forward their queries via C, then it may be that one or both of these queries may be answered from C’s cache
    - In this case the signal is being suppressed
- Because its actually measuring a cause, not the outcome
    - Its measuring resolvers’ uptake of the new KSK, but is not able to measure the user impact of this

User-Side Measurement



Can we devise a DNS query that could reveal the state of the trusted keys of the resolvers back to the user?

- Not within the current parameters of DNSSEC and/or resolver behaviour

User-Side Measurement

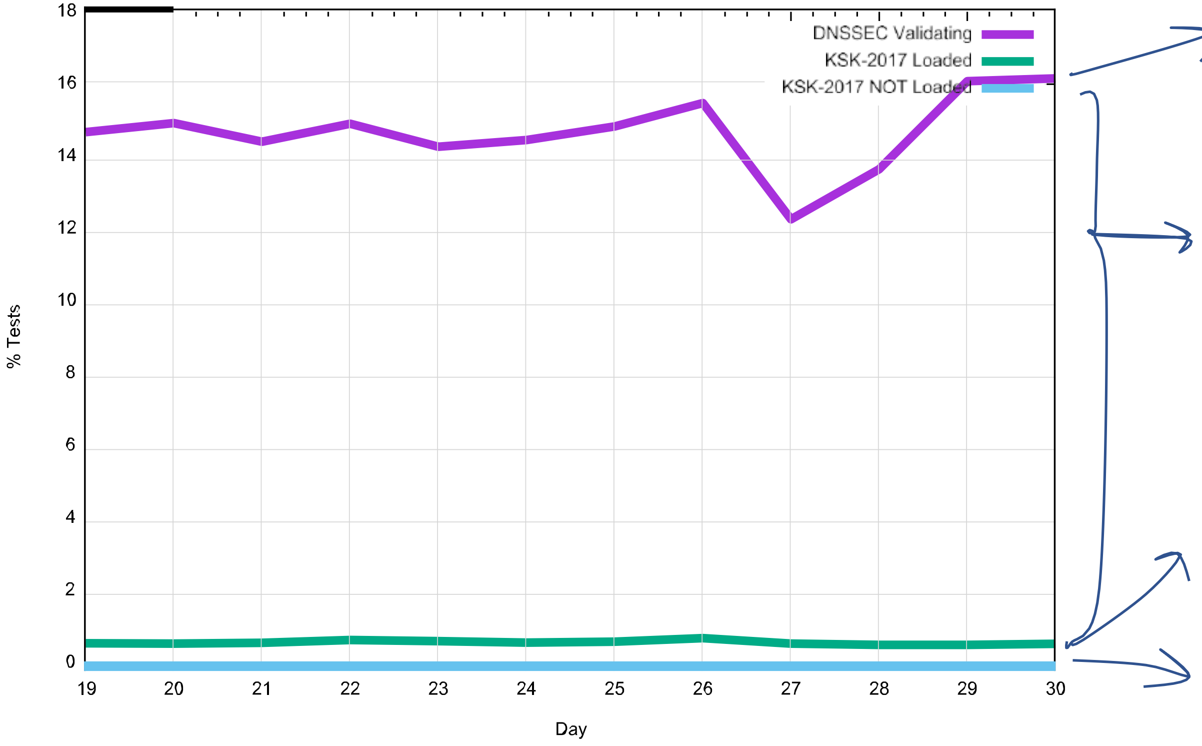
Can we devise a DNS query that could reveal the state of the trusted keys of the resolvers back to the user?

- What about a change to the resolver’s reporting of validation outcome depending on the resolver’s local trusted key state?
  - If a query contains the label **“root-key-sentinel-is-ta-<key-tag>**” then a validating resolver will report validation failure if the key is NOT in the local trusted key store
  - If a query contains the label **“root-key-sentinel-not-ta-<key-tag>”** then a validating resolver will report validation failure if the key IS in the local trusted key store

DNS + Web

- How can you tell if a user is able to resolve a DNS name?
  - Be the user (get the user to run a script of some sort)
  - Look at the DNS server AND the Web server
    - The Web object is fetched only when the DNS provides a resolution answer
    - But the opposite is not necessarily the case, so there is a noise component in such an approach

Prior to the KSK Roll



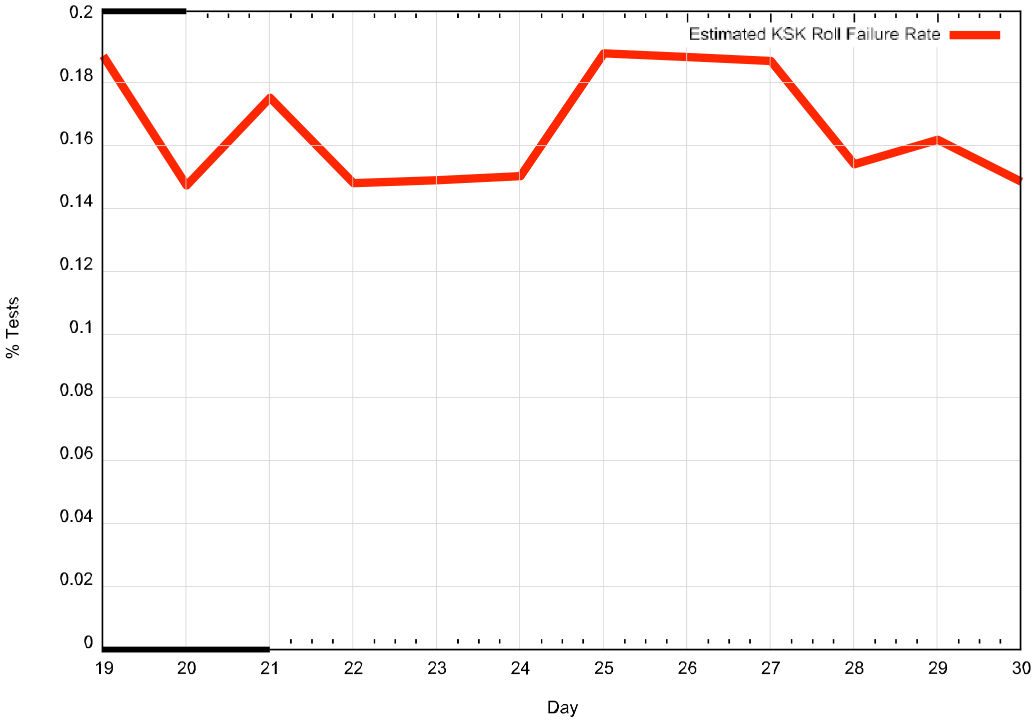
16% of users use DNSSEC-validating resolvers

15% of users do not report their KSK trust-state

0.5% of users report KSK-2017 loaded

0.005% of users report KSK-2017 NOT loaded

Possibly Affected Users



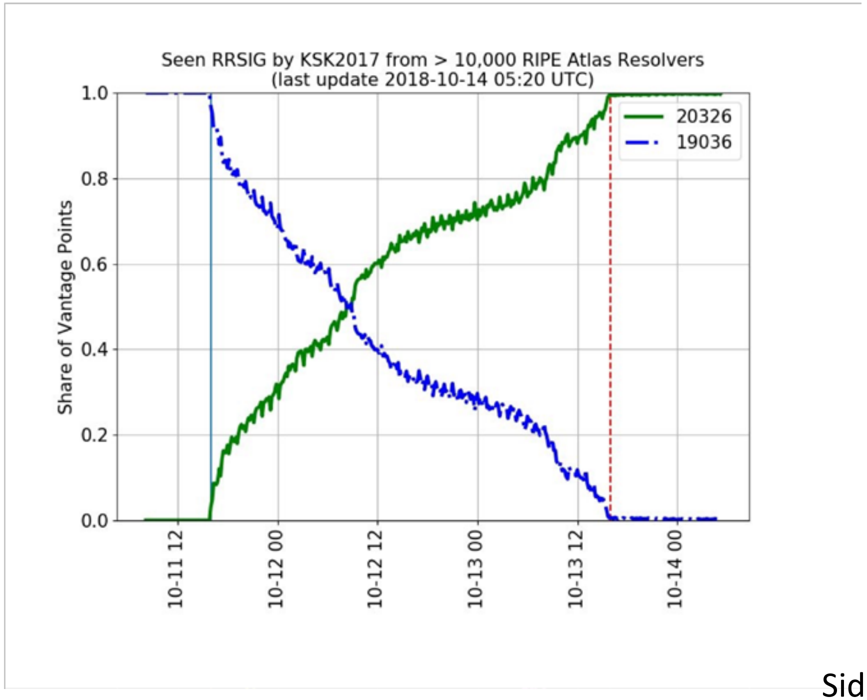
Between 0.1% to 0.2% of users are reporting that their resolvers have not loaded KSK-2017 as a trust anchor

The measurement has many uncertainties and many sources of noise so this is an upper bound of the pool of users who may encounter DNS failure due to to the KSK roll

There is still uncertainty in this measurement

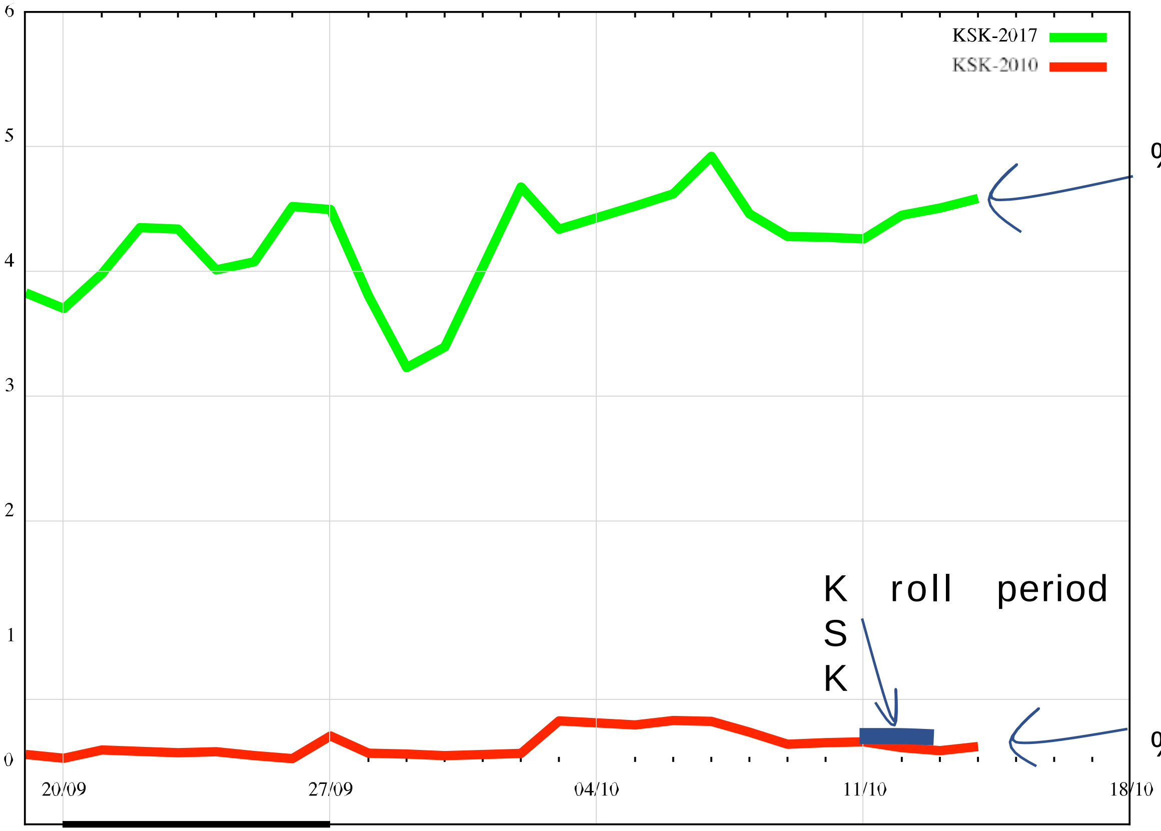
- Not all resolvers will pre-provision KSK-2017 using RFC 5011 automated trust mechanisms – they may elect to load the new trsut anchor at the time of the roll manually
  - And we cannot measure the difference between a resolver that has a broken implementation of RFC5011 and a resolver that is being managed manually
- Only recently upgraded resolvers have this test behaviour included
  - But the resolvers we worry about are the crufty ones at the bottom of the rack that have been all but forgotten!

What happened



Sidn Labs Atlas Measurement

What we saw



% of folk that reported "good"

KSK roll period

% of folk that reported "bad"

What did we learn

- Last minute attempts to change DNS behaviours are pretty futile
  - You only see updated infrastructure, but we are worried about aging infrastructure
- Measuring resolvers is NOT the same as measuring users
- We were lucky this time
  - However it wasn’t only luck as much effort was expended on publishing the KSK roll

Keep It Rolling

- Validating resolvers with static KSK 2010 keys are now dead resolvers
- Resolvers are either using 5011 or manually loading the key
  - 5011 resolvers should be fine
  - Manually loaded resolvers will need to be regularly tended
  - How loud do we need to shout and for how long to get the manually loading resolvers to switch to 5011 automated load?
- Maybe we just need to keep rolling every year
  - That way we train the manual loaders to keep up!

Thanks!