Comparing TLD DNSSEC Practices with RFCs

Edward Lewis
Neustar, Inc.

At the IEPG on the day before the 83rd IETF

March 25, 2012

neustar.

Purpose of this talk

- I've been monitoring the DNSSEC deployments of the root and TLD zones for some time
- Presented a "what I did/am doing" talk at APRICOT and some summary comments at ICANN
- This time, before the IETF, I thought it would be interesting to compare the observations made to the RFCs that have been published recommending how DNSSEC should be run

Why?

- Originally, I did this in response to a question
  - DNSSEC has a few operational parameters, what settings should I use? E.g., what kind of keys, how long?
- The root zone and TLDs are working examples
  - Perhaps not the best match for "the usual DNS'ers"
  - But working examples nonetheless
- Later the emphasis moved on to studying the various choices of the TLDs
- Now, it's interesting to see how much a role the experience of workshopping DNSSEC and RFC publication play

IETF documents

- RFC 4641 "DNSSEC Operational Practices"
  - This document was published in late 2006, quoting:
    - *this document should therefore explicitly not be seen as representing 'Best Current Practices'*
    - *The suggested key sizes should be safe for the next 5 years*
  - Still, this document is cited in RFPs, seeking conformance
- RFC 5155 "(DNSSEC) Hashed Authenticated Denial of Existence" or "the NSEC3 RFC"
  - Has a few operational recommendations
- RFC 4509 "Use of SHA-256 in DNSSEC DS RRs"
  - Has a recommendation relating to transition

The observations

- Collection of data since late June 2011 and ongoing
  - First presented at APRICOT 2012
- Hourly looks "smoothed" to daily to capture operational policy and ignore network events
- Conversion of raw responses into useable data (like converting the key into the footprint and size), looking for the lifetime of records
- Counting stats like algorithms used, calculating "averages" such as number of records active at a time

Observations in brief (trend)

Date	07/01	09/01	11/01	01/01	03/01	03/13
Root&TLD	299	300	300	302	302	303
Signed	64	65	73	78	80	80
With DS	59	61	62	71	73	74
RSA/SHA1	38	38	40	40	41	41
RSA/SHA256	23	24	30	35	36	36
1K-long ZSK	62	63	70	75	77	77
2K-long KSK	56	56	65	72	74	74
"AND"	55	55	63	70	72	72

"AND" means using both a 1K ZSK ***and*** a 2K KSK

From a summary at ICANN

- "Most common" choices (not universal), as measured in late February:
  - RSA SHA-1 "old guard", RSA-SHA-256 "newbies"
  - ZSK/KSK approach
    - 1024 bit ZSK, 2048 bit KSK
  - One ZSK and one KSK active and present
  - NSEC3 over NSEC
    - with 1 iteration
    - 4 byte (8 hex char) salts
    - rarely/never changed
  - DS record added 3 weeks after DNSKEY appears

RFC 4641 "Operat'l Pract..."

- RFC 4641 is a fine document, it is a discussion of how to run DNSSEC
  - It is general, not focused on the particular use case of the root and TLDs but they are mentioned
  - It is showing its age - recommendations have exceeded self-imposed milestones
- But seeing it as a conformance document is ~~difficult~~ impossible
  - Does not profess to be prescriptive
  - Does not contain succinct, concise, testable requirements

RFC 4641 - 3.1.1 "Keys"

- *Differentiating between the KSK and ZSK functions...*
  - The assumption here is that keys will be split into the two roles
    - The only TLD that did not split the functions converted to this model converted in September.
    - All TLDs, at least when interacting with the IANA root, now employ a KSK/ZSK model
- *...the KSK can be distinguished from a ZSK by examining the flag field in the DNSKEY RR.*
  - True.

RFC 4641 - 3.1.1 "Keys"

- *The KSK can be made stronger*
  - Of the 80 [March 11, 2012] signed zones, only 5 use the same key lengths for KSK and ZSK, rest make KSK longer
- *A KSK can have a longer key effectivity period*
  - In my measurements, no KSK has been gone through it's entire effectivity period yet, and only 5 zones have never changed their ZSK - not the same 5. So, roughly "yes."
- *This allows for signature validity periods on the order of days*
  - Only 13 zones have signature validity that fits the "1 week" bucket

RFC 4641 - 3.1.1 "Keys"

- *The Key Signing Key ... key effectivity period can be on the order of years, we suggest planning for a key effectivity on the order of a few months so that a key rollover remains an operational routine.*
  - Since June, no KSK has progressed throughout its entire effectivity period. I.e., no one follows this recommendation.
  - This comment should be added - ZSKs are rolled, most TLDs roll them on the order of a month or three, so "operational routine" is exercised, just not involving the IANA interface

RFC 4641 - 3.2 "Key Gen"

- *Careful generation of all keys is a sometimes overlooked but absolutely essential element in any cryptographically secure system.*
  - Three times a pair of public keys shared the same footprint (aka keyid)
  - Twice the pair differed in algorithm (so they aren't a match)
  - One pair does share a footprint and algorithm - but is not the same key pair! Phew...
    - Footprints are not unique to keys!

RFC 4641 - 3.2 "Key Gen"

- *From a purely operational perspective, a reasonable key effectivity period for Key Signing Keys is 13 months, with the intent to replace them after 12 months. An intended key effectivity period of a month is reasonable for Zone Signing Keys.*
  - For ZSK, a slight majority use 1 month. A sizable majority use 2 months and some three. The ratio is about 3:2 between one and two months.
- *Key effectivity periods can be made very short, as in a few minutes.*
  - Not in anyone's wildest dreams! ;)

RFC 4641 - 3.4 "Algorithm"

- *We suggest the use of RSA/SHA-1 ...SHA-256 ...as soon as these algorithms are available...*
  - This recommendation is evident in the observations. Mid-last summer, RSA/SHA-1 was dominate and even today still holds a slight edge over RSA/SHA-256
  - But of the newly signed zones, vast majority used RSA/SHA256
    - In June it was nearly 2:1 SHA1: SHA256. It's almost break even now, with only 3 SHA1's added vs.13 SHA256's.

RFC 4641 - 3.5 "Key Sizes"

- *...we come to the following recommendations about KSK sizes: ... 2048 bits for high-value domains. ... The suggested key sizes should be safe for the next 5 years.*
  - "The next five years" expired in September 2011
  - Four TLDs use KSKs smaller than 2048 bits
  - Two use KSKs larger than 2048 bits
  - Leaving 74 of 80 currently signed zones using 2048

RFC 4641 - 3.5 "Key Sizes"

- *As ZSKs can be rolled over more easily (and thus more often), the key sizes can be made smaller. But as said in the introduction of this paragraph, making the ZSKs' key sizes too small (in relation to the KSKs' sizes) doesn't make much sense. Try to limit the difference in size to about 100 bits.*
  - Again, most TLDs use ZSKs that are smaller than KSKs, and most use the recommended 1024 and 2048 lengths
  - The comment to limit the difference to 100 is not observed by anyone

RFC 4641 - 4.1.1 "Time"

- *We suggest the Maximum Zone TTL of your zone data to be a fraction of your signature validity period.*
  - In general this is true, but at least one TLD publishes all signatures with the same expiration date
  - When that date is near, all signature durations extend past it
- *We suggest the signature publication period to end at least one Maximum Zone TTL duration before the end of the signature validity period.*
  - Seems to only apply when "batch" signing, registries exhibit dynamic signing, so this does not really apply anymore. (Admittedly, this isn't measured by the observations.)

RFC 4641 - 4.1.1 "Time"

- *We suggest the Minimum Zone TTL to be long enough to both fetch and verify all the RRs in the trust chain.*
  - Not really measureable
- *Slave servers will need to be able to fetch newly signed zones well before the RRSIGs in the zone served by the slave server pass their signature expiration time.*
  - Not measured, not a problem for TLDs

RFC 4641 - 4.2.1.x "ZSK roll"

- *Pre-publish key rollover: ... the new key is published in the key set and thus is available for cryptanalysis attacks. A small disadvantage is that this process requires four steps.*
- *Double signature ZSK rollover: ... this may be prohibitive if you have very big zones.*
  - Rarely has the SOA set for a TLD used more than one signature, indicating TLDs universally opt for the Pre-publish roll when changing ZSKs

RFC 4641 - 4.2.2.x "KSK roll"

- *For the rollover of a Key Signing Key, the same considerations as for the rollover of a Zone Signing Key apply. However, we can use a double signature scheme... zone size considerations do not apply.*
  - Making a general statement is premature, only a few KSK rolls have been observed
  - Rarely is a TLD seen changing the number of DS records, indicating that few if any do not follow this recommendation

RFC 4641 - other sections

- RFC 4641 has a lot of discussion on aspects that are either not measureable or have not been exercised
  - Comments on private key storage - how this is done is not apparent in the protocol
  - Key compromise - this has not happened
    - One zone did use a bad key, but this was discovered in early testing and was corrected by "rebooting" DNSSEC

RFC 5155 - "NSEC3"

- *...it is strongly RECOMMENDED that Opt-Out be used sparingly.*
  - Four TLDs use NSEC3 only for access to Opt-Out
- *It is RECOMMENDED that the salt be changed for every re-signing.*
  - 48 have never changed, 10 changed once in a while, 3 change nearly daily (63 total, 62 are NSEC3 today, one stopped)
- *A zone owner MUST NOT use a value higher ... for iterations for the given key size: 1024b=> 150; 2048b=> 500; 4096b=>2,500*
  - One zone uses 150, rest are distributed between 0 and 20

RFC 4509 - "SHA256 for DS"

- *...zone operators should consider deploying both SHA-1 and SHA-256 based DS records. This should be done for every DNSKEY for which DS records are being generated. Whether to make use of both digest types and for how long is a policy decision that extends beyond the scope of this document.*
  - Based on March 19 data
  - 74 Zones have a DS set in the root zone
  - 69 have a SHA-256, 40 SHA-1
  - The overlap - this has been consistent since the survey began
    - 34 have only a SHA-256
    - 5 have only a SHA1
    - 45 have both

Summary of the RFCs

- RFC 4641
  - The (signed) TLDs are operating within the spirit of the document
  - The document itself has passed at least one expiration timer
  - The document isn't totally consistent, and is not a set of requirements
- RFC 5155
  - A mix of protocol definition and operational, with 1 of 3 operaional recommendations followed
- RFC 4509
  - Only about half heed the advice on the transition

Summary of the work

- The RFCs have in general two weaknesses when it come to reflecting the true nature of operating DNSSEC
  - A good quantification of the nature of cryptographic parameters
  - Assuming "batch" operations when "incremental" has become the norm. The world of TLD operations has evolved rapidly in the past few years

- A plea to those creating "compliance requirements"
  - Make sure the cited documents have requirements
  - Make sure the documents are relevant to you needs

That's it...

- Q&A time