DNSSEC – The Journey at a Crossroads

A Personal View of the state of the Extensions

Edward Lewis

IDS 2023
5 September 2023

The Title

- *DNSSEC: Maybe it's the Journey and not the Destination*
  - A 2008 lament that DNSSEC progress was slow and getting slower
  - Listed the benefits the effort had yielded and wondered if there was a will to progress
- 15 years later, time to revisit this idea

My Perspective

- Wrote first (and second) DNSSEC zone signer (1996-1997)
- Wrote the first DNSSEC validator (1997)
- Attended the first DNSSEC Deployment meeting (1998)
- Ran many operational workshops (1999-2004)
- *DNSSEC: Maybe it's the Journey and not the Destination* (2008)
- TCR for Root KSK Ceremonies (2010-2014)
- Measuring DNSSEC records in use at TLDs since 2011
- 2002-> worked for DNS registries, DNS hosting, and now ICANN

DNSSEC – The Journey and The Crossroads

- Initial development: mid 1990's
- First meeting on DNSSEC deployment: April 1, 1998
- Current baseline definition: 2004

- In 2023:
  - Validation: APNIC Labs measures (world wide) around 30%
  - Signing: 4% of .COM (and many other TLDs) names have DS records
  - Criticism that DNSSEC is too hard to run and solves a non-problem
  - Lots of minor updates to the extensions actively proposed

Is DNSSEC still needed?

- The state of DNS is much better than when DNSSEC development started
  - Better software, operating procedures
- We have TLS. Is application security what we need?
  - Can the Internet be used securely without a trusted naming (and routing) system?
- Can trusted code run on untrusted machines?
  - Can code be self-reliant, decrypt itself when it needs to run?

Why is this Important for Emerging Technologies?

- To accommodate emerging technologies
  - Should they have to build in their security layers?
  - Or should they work on a secured base?
  - How well-secured?

- How could we make DNSSEC ready for emerging technologies?

Opinion

- I think we still need DNSSEC
  - But the current form is not working out
- For emerging technologies
  - Provide a secured, level-playing field
- The goals of DNSSEC are sound; but something is flawed
  - The design, for the 1990's environment, isn't fitting right
  - Operating systems and cryptography have evolved
  - The field of DNS operations hasn't just evolved, it began

What DNSSEC was Set to Solve

- Data Authenticity
  - That the data was as the zone administrator published
- Data Integrity
  - That the entire answer was obtained
- Negative Answer Proof
  - This seems an odd goal, but the DNS allowed for empty responses
  - Empty is hard to secure

DNS Security Strategy

- Data Security
  - Digital signatures and distribution of public keys (DNSSEC)
- Channel Security
  - Message security (TSIG and more)
- Platform Security
  - OS, host, facility, business processes

DNSSEC Hopes

- Backwards compatible
  - DNSSEC was foreseen as following a slow adoption curve
  - Co-existence with un-signed DNS was a must
- Be as flexible to counter discipline enforced on the DNS
  - Bend, but don’t break, when it comes to “secure”
- Be operations friendly
  - This was a driver for the early workshops

The State of the Internet when DNSSEC Began

- Host security was weak
  - Private keys had to be air-gapped away from the network
- Cryptography
  - Export-restricted, patent-encumbered technologies
- Lots of non-standard extensions to the DNS Protocol
- DNS-as-a-service market did not exist
- Middleboxes (firewalls) were new/controversial
- No anycast routing

Impact on DNSSEC Design

- No name server access to private keys
  - All responses had to be pre-computed on a non-connected machine
- Had to accommodate all known protocol elements
  - The protocol was not widely understood
- Create ”name order” (sorting)
- Incorporate wall-clock time, mix with TTL rules
- No consideration for changing operators (modern market)
- No concern about response size (middleboxes)

Securing Negative Answers (DNSSEC Goal #3)

- Have to pre-compute all answers, not knowing the query
  - ”Here is what I have, you can see the data you want is not here”
  - Enables zone walking
  - Requires a sorted order of names in a zone
    - This one point is why BIND 9 replaced BIND 8 in the late 1990’s
- NSEC3 w/opt-out and Wildcards have never ”gotten along”
  - A corner case that could not be resolved

Securing Synthesized Responses (Wildcards)

- A “generic” response record created for synthesized answers
  - Allowance made for a different owner name, via label count
  - The “upper labels” of the query name had to match the wildcard (source of synthesis) name, “lower/leaf” labels were excluded
  - The data (RDATA) field was fixed to one value

- Records in a message response
  - Have to show the process was followed, not just the result
  - The reason multiple, signed negative records are needed

Cryptography and Key Handling

- Zones were assumed to run with multiple security algorithms
  - Validator still had to know what to expect
  - Response size was not considered

- A lot of design effort was spent on the child-parent exchange
  - Should the keys be at the parent or child?
  - What signaled “child is not signed”?

Time

- DNSSEC created the need for absolute time
  - Inception/Expiration of signatures
  - Thwart replay attacks, limit damage from hijack
- DNS already had TTL, relative time
  - Limiting TTL values kept data fresh, useful when changing records
- Mixing absolute and relative times is not easy (clipping TTL)
- Hijacking using far-future expiration times was not foreseen

Workshops

- After publishing the initial base definition
  - Series of workshops used to make it operable
  - DS resource record, functional roles created, KSK and ZSK

- Predated the emergence of DNS operations
  - Predated EPP (provisioning) protocol
  - Major DNS hosting companies established 1999-2001
  - Participants were still primarily protocol developers and research

The Crossroads

- DNSSEC addresses needed goals and has a solid design
- But the operations world has different needs today
  - Option: Force fit what is needed upon DNSSEC's implemented framework
  - Option: Go back to the first goals and reimagine approach
- To be deployed, must be *operations-friendly*

What Is Needed in Operations?

- Low-risk activities
  - Operators' chief job it to keep a service up and running
- Easy to monitor, quick to fix
  - When things break, fast restoration is the goal
- Tools with Default Values
  - Operation staffs are not software developer staffs
- Justification
  - Risk/reward must be clear
  - Convince the agency that approves operational changes

Operations Friendly

- What does this mean?
  - Easy to deploy, simple, low-configuration
  - Easy to co-exist, does not negatively impact other systems
  - Easy to maintain, tools available to monitor, raise alert
  - Easy to fix, limit mean time to repair
  - Easy to “get it right”, hard to accidently break
  - Easy to gain approval from change approval boards
  - Easy tech-refresh, change providers, re-deploy, automate
  - And easy to explain and understand

How Has DNS Changed?

- Next slides will walk through the changed world of DNS

On-line signing

- A brilliant idea ruled out of bounds during early development
- Vendor lock-in (a bit) as a result of not being standard
  - No standard for key sharing within a zone's different operators
  - Vendors provide means to avoid customers being locked in
- Could design a "standards way" to do on-line signing

With On-line signing…

- Can tailor response to the query name and type
- Major impact is on negative answers
  - No need to sort a zone
  - No need for a type bitmap
  - Never have to see the “whole zone”: friendly to high churn zones
  - Any change impacts just one name
  - Synthesizing response need not alter the RDATA
  - No need for hashing names

Cautions with on-line signing

- The key is vulnerable to exposure, do we need a special negative answer key? Would this increase the size of the DNSKEY resource record set?
- Can the same ZSK work for the signatures on the server and any pre-generated signatures? What about a “Common Signing Key (CSK)” set up?

- There are commercial deployments doing on-line signing, so there are working examples

Parent-Child Key Exchange

- Automating a roll of a Secure Entry Point (aka KSK) key is a work in progress
  - CDS and CDNSKEY proposals
  - CSYNC too, in the spirit that DNSSEC is grafted on top of DNS
- These proposals are still being tinkered with
  - CDS/CDNSKEY defined using polling, with an event-driven mechanism in proposal

Parent-Child Key Exchange Progress

- Although this work is in progress, progress is slow
  - Lack of clarity in the registry, registrant (zone admin) and DNS operator triangle
    - What happens when a change is barred by a registration lock/policy?
  - Real or perceived policy barriers regarding registry work with operators
  - In a study to determine how DNSSEC operators manage keys
    - Finding periods for ZSK was easy, many examples of operator rolling ZSK
    - Finding periods for KSK impossible, even TLD operators are reluctant to roll KSK

Outsourcing DNS Hosting

- Zone admins off-load their work to one/more providers
  - DNS-as-a-service
  - Might be multiple
  - May include DNSSEC signing of the zone
  - “Multi-signer” is one name for this
- Zone admins want to change their providers
  - Besides the ability to share responsibilities between providers
  - Need to be able to roll from one provider to another
  - “Domain name transfers” is one version of this

Multi-signer Considerations

- Validation has to succeed in a caching environment
  - Has to be a way for multi-signers to share the same key set
- More keys means the DNSKEY resource record set grows
  - Can each provider have it's own keys? Maybe, maybe not
  - Is there space enough for specialized on-line-only keys?
- DNSSEC rules as written now, make multi-signer difficult
  - Response size impacts

Trust Anchor Considerations

- Trust Anchors are owned/managed by validators
  - Most operators of validators rely on what comes in software distribution
- “Automated Updates of DNSSEC Trust Anchors”
  - Overloads DNSKEY resource record meaning
  - Relies on validators knowing to look for trust anchor signals
  - Has never been used to change DNS security algorithms
- Need an explicit approach to Trust Anchor “suggesting”

Opinions on the Onward Path

- Further determine what is “operations friendly”
- Question old taboos
- Add versioning to the protocol to accommodate change
- Explore needed improvements, judge the effort to get there
- Measure success by deployment rates, operator adoption

Engage with ICANN

Thank You and Questions

Visit us at **icann.org**
Email: edward.lewis@icann.org

@icann
linkedin/company/icann
facebook.com/icannorg
slideshare/icannpresentations
youtube.com/icannnews
soundcloud/icann
flickr.com/icann
instagram.com/icannorg