Garbage Prevention

- DNSSEC Pre-Publication Consistency Checks -

Peter Koch <koch@denic.de>

Joint CENTR/OARC Meeting, Wien, 30 October 2011

Background

- `DE` zone signed with DNSSEC as of 31 May 2011
  - DURZ like rollout (DUdeZ in our case ...)
  - `DS` RR in the root zone in early June (IPv6, anyone?)
  - `NSEC3`
    - With opt-out
  - DNSSEC Parameters retained
    - except `NSEC3` hash iteration count

- Several domains *survived* the DNSSEC Testbed
  - `DNSKEY` data was already in the registry
  - ~ 230 signed delegations
  - ~ 200.000 domains signed (auth data)
  - ~ 350.000 `NSEC3`, ~ 700.000 `RRSIG`

Motivation

- Late 2010/Early 2011 saw several DNSSEC induced incidents across TLD land
  - Hard to find innovative bugs
  - We already had our our *bad zone day*

- Strong desire to maintain stability
  - Counter added complexity
  - Conservative approach
    - E.g., a name server is a name server is a name server ...

- Data quality addressed by predelegation checks

Precautions

- Avoid troubles by proactive checking
  - Consistency with the *sans DNSSEC* case
  - Protocol correctness
- Build, steal, or what else?
  - Several ccTLDs working on similar projects
  - Discussions, exchange of ideas
  - Potential incompatibilities
    - Zone size
    - Operational model (full zone signing)
  - Eventually rollout plan trumps

Design Criteria

- Diversity
  - Code, Language
  - Libraries
  - Personnel
- Zone data only
  - No *Trust Anchors* available
  - No access to registry DB
  - Within publication chain (no live queries)
- Focus on DNSSEC signer output

Test Groups

- DB Content
  - No orphaned DS RRs
  - DS RR syntax correct
  - Number of DS RRs within acceptable range (heuristics)
- High level consistency
  - Signer output – DNSSEC == Signer input
- DNSSEC consistency

Test Groups: High Level

- Signer output – DNSSEC == Signer input

- Canonicalize zone (`named-compilezone`)
  - Hash the result
- Remove all DNSSEC data (except DS RRs) from signed zone
  - Canonicalize, hash
- Compare results
  - For extra safety, compare with hash(empty)

Test Groups: DNSSEC Consistency

- Address DNSSEC Signer output

- Does #`NSEC3` RRs match #auth data + #ENTs?
- Do `NSEC3` RRs form a single closed chain?
- Does #`RRSIG` match #auth RRSets?
- Are `NSEC3` parameters consistent (and do they match `NSEC3PARAM`)?
- Do all keys and signatures exist at the zone apex?
- Are all `RRSIG` inception and expiration dates within reasonable (configurable) bounds?
- Do all `RRSIG`s validate (independent of time) against ZSK?
- Does the `SOA` RR‘s signature fully validate (starting at KSK)?

Test Groups: DNSSEC Consistency Software

- Consistency check tool implemented in C
  - Uses `ldns` (kudos to NLnetLabs)
  - DNSSEC signer is Java based (Verisign)
- Implemented and tested by independent ad-hoc team

DNSSEC Pre-Publication Alarms triggered to date

DNSSEC Pre-Publication Checks Next Steps

- Software refactoring
- Improve runtime
  - Compared to signing time
- ... and scalability
- Incremental checks
  - NSEC3 makes life interesting
  - ... as does auth zone data
- Cooperation
  - „dnssexy“ or others

?

<http://www.denic.de/dnssec>