NSEC to NSEC3
KSK algorithm rollover

CZ.NIC
Jaromir Talir / *jaromir.talir@nic.cz*
16.11.2010

CZ DOMAIN REGISTRY

Motivations

- Political
  - Zonewalking was considered as a tool for domainers
- Administrative
  - NSEC3 switch from RSASHA1 requires KSK rollover
  - Opportunity to force DNS operators to switch to validation with root KSK
  - Yet another promotion of DNSSEC
- Technical
  - SHA1 is becoming obsolete (NIST recommendation)

Preparations

- Algorithm RSASHA512 (alg=10)
  - Take the best on the market
- NSEC3 w/o opt-out
  - 15% domains already signed – size is not a big issue
  - Opt-out variant doesn't protect unsecured domains, NSEC does
- NSEC3 parameters
  - Salt rotation with ZSK rollover
  - Size of salt/number of iterations – we looked at NSEC3PARAM RR from other TLDs
  - Size of salt: 68 bits
  - Number of iterations: 10

Preparations

- Upgrading all primary/secondary NS to recent versions of software - Bind and NSD
- Upgrading signer host to have more memory
- Upgrading our own resolvers to recent versions of software – Bind and Unbound – (ODVR and internal)
- Marketing to DNS operators – “Hey! Upgrade resolvers and validate using root KSK and not .CZ KSK!”
  - Testbed with NSEC3 signed zonefile
  - Dedicated website, mailing list, company blog, articles in technical magazines

KSK algorithm rollover

- NSEC3 is incompatible with plain RSASHA1 (alg=5)
- It's not possible to change just ZSK algorithm
- KSK rollover is inevitable
- **But this is algorithm rollover**!
- **Classical prepublishing rollover method cannot be used!**
- **Simple publication of DNSKEY with different algorithm in zonefile makes the zone bogus (invalid)!**
- In reality – different resolver software behaves differently
  - BIND – doesn't treat it as bogus zone (not RFC compliant)
  - Unbound – does treat it as bogus zone (RFC compliant)

KSK algorithm rollover

- RFC 3045 – section 2.2
- “There MUST be an RRSIG for each RRset using at least one DNSKEY of each algorithm in the zone apex DNSKEY RRset. The apex DNSKEY RRset itself MUST be signed by each algorithm appearing in the DS RRset located at the delegating parent (if any).”
- (1) You need to sign **each RRset** with new DNSKEY
- (2) You need to put signatures into zonefile **before** DNSKEY
- (3) You need to send DS upstream **after** previous steps

KSK algorithm rollover

- 3.8.2010
  - We signed zonefile with new KSK/ZSK (+old KSK/ZSK), published this big double-signed zonefile and waited TTL for RRSIG
  - We inserted new KSK/ZSK into zonefile and waited TTL of DNSKEY
  - We sent exchange request to IANA to insert new DS and remove the old ones. (processed in two days!)
- 24.8.2010
  - We removed old KSKs from zonefile and waited TTL of DNSKEY
  - We signed zonefile just with new KSK/ZSK and waited TTL of RRSIG
  - We signed zonefile once again using NSEC3 instead of NSEC

Conclusion

- Manual process – not supported by tools (OpenDNSSEC, ZKT...)
- Testing is crucial – different vendors, different versions
- We found bug in Bind - #22309
  - When old RSASHA1 were removed and signatures were still there, Bind didn't return AD bit
  - Will be released in 9.4-ESV-R4, 9.6.2-P3, 9.6-ESV-R3, 9.6.3 (b2/rc1), 9.7.2-P3, 9.7.3, 9.8.0.

Questions??

jaromir.talir@nic.cz