The Trust Anchor Key Renewal Method
Applied
to DNS Security

Presentation by Thierry Moreau, CONNOTECH

DNSSEC mission:

- **DNS data origin authentication**
- **integrity assurance**
- **authenticated negative answers**

DNSSEC omissions, missing and wanted at IETF-64:

- **privacy on negative answers, i.e. NSEC3**
- **key management**

  |

  ----> **a void to be filled**

  - **... hence TAKREM by CONNOTECH**
  - **... a "fortuitous" invention**
  - **... from security consulting to SAKEM to TAKREM**

Trust Anchor Key Rollover
Hard Technology Requirements

Golden rule: ☞ **DNS resolver automation** ☜

security automation => cryptography assisted solution

why roll anchor keys in the first place?

- **cryptographic strength concerns,**
- **security operations concerns,**
- **survivability concerns: a scheduled rollover is**
  - **a rehearsal for an emergency rollover,**
  - **an alternative for it.**

any scheme has a "catastrophic failure mode" i.e. when the rollover crypto breaks

signature chaining over time	**... fails on ...**	**a private key compromise (weak link)**
long-lasting master key	**... fails on ...**	**a master key compromise**
pre-announced fingerprints of keys	**... fails on ...**	**key compromise in dormant state**

Trust Anchor Key Rollover
Soft Technology Requirements

operational:

- **rollover tied to zone manager procedures**
- **resolver programs are user agents, no "MUST" for end-users**

☞ specify zone manager procedures ☜

☞ that enable resolvers to roll KSK with on-going trust ☜

"the DNS devil is in the details"

deployment requirements

- **root**
  - **ultimately, rollover required only for the root**
  - **put something trusted in ISC bind release before live deployment**
- **other "islands of trust"**
  - **large number of islands of trust**
  - **transition from island of trust to normal signed zone**
  - **(no going back to unsigned parent)**

TAKREM in DNS -- Rollover Procedure

zone manager rollover preparation

- retrieves a future trust anchor key from dead storage,
- publishes new DNSKEY (SEP=1) RR,
- ~~gets the parent to sign the DS RR~~, and/or publishes SDDA RR for the new KSK
- uses the new DNSKEY (SEP=1) RR to sign DNSKEY RRset

resolver state before rollover, integrity protected configuration:

- trusted digests for future trust anchor keys
- current trust anchor key

resolver procedure

- encounters an unknown DNSKEY (SEP=1) RR
- checks parental DS RR (up to a trust anchor)
- if trusted digests are configured for the zone - and - no validated parental DS RR
- then queries the zone SDDA RRset
  - validates DNSKEY + SDDA against trusted digests
  - if validated, accepts DNSKEY RR as new current T.A.K.

TAKREM validation failure:

- ignore the DNSKEY RR as a new current trust anchor key
- flag the current DNS request as bogus

TAKREM in DNS
Solution Properties

zone management procedures

- **simple zone management procedures**
- **manifest procedures for the auditor function**
  - **"who guards the guards"**
- **emergency rollover same as scheduled rollover**
  - **(except for the key revocation issue)**

DNS protocol properties

- **single DNSKEY (SEP=1) RR**
- **no need for resolvers to stay on-line as with chained rollovers**
- **smooth transition from island of security to normal zone**
- **uses simple DNS lookup for the SDDA RRset**

Next Steps

TAKREM for DNSSEC available as an IETF DNSEXT contribution:

- change control handed over to IETF
- best alternative to long-lasting trust anchor key

commitment to bring T.A.K. rollover in DNSEXT scope

- going past the "get it done anyhow" attitude
- define overall requirements, given "catastrophic failure mode"

detailed requirements definition

- is anything mandatory beyond root zone management procedures?
- resolver requirements
  - --> user interface for indeterminate DNS results
- TAKREM tames the "resolver trust anchor pop-up paradox"

the end

Appendix -- Cryptographic Principles

context of crypto usage

- very infrequent use
- don't automate infrequent/low volume operations
  - handling keys on the server side ---> manual procedures
  - fully automated on resolver side
- intended to support long-term trust in resolvers ---> larger crypto parameter sizes

TAKREM

- uses a "pre-announced fingerprint" approach
- long-term resistance to cryptanalysis:
  - hidden hash primitive selection (from a function family)
  - variable cryptographic strength

MASH Modular Arithmetic Secure Hash

- based on the Rabin-Williams one-way trapdoor function
  - (other being RSA variant, discrete log and elliptic curve)
- i.e. best number-theoretic support ([Boneh, Bernstein, ...])
- an ISO/IEC standard, development process supported by academia
- modulus selection like Rabin-Williams or RSA
- too slow for anything but infrequent use
  - (consistent with avoiding e.g. SHA-1 collapse threat)