DANE SMTP and OPS open issues

Viktor Dukhovni
Two Sigma
&
Wes Hardaker

IETF 89, London
March 2014

TLS discovery

- SMTP (pre-DANE) TLS
  - Opportunistic and unauthenticated
  - STARTTLS downgrade
  - Unsafe post-MX name checks
  - Too many (and yet too few) trusted CAs
- DANE opportunistic TLS
  - Enables downgrade-resistant TLS
  - Provided TLSA can be used for discovery
  - No significant increase in DNS workload
  - SMTP tolerates modest latency
  - MTA hosts can use proximate resolvers
  - Many DNS lookups are already being done
    - RBL, RHSBL, DNSWL, SPF, DKIM, PTR, ...

DANE-EE(3) cert semantics

- Goals:
  - Server operator chooses policy and timing of key rotation
- Skip name checks (DNSSEC binding)
- Skip CT (no CAs to log)
- Decisions:
  - Do the below depend on the selector?
  - Ignore expiration date with either or both?
  - Ignore EKU “purpose” with either or both?
  - Match TLSA and ignore “everything”else?

DANE-TA(2) semantics

- Selector
  - Cert(0) and SPKI(1) vs. TA cert content?
  - SPKI(1): only SPKI covered by TLSA
- Bare key: SPKI(1) Full(0)
  - Must clients support this
    - absent corresponding cert in peer chain?
  - If bare keys not supported:
    - why not always publish a digest?

Digest Algorithm Agility

- Use only best mtype != 0 per CU+selector?
- Which mtype (digest) is the best?
  - It is the client's policy!
- Handling of non-conforming records?
  - Suppose TLSA RRset has 2 x “3 1 1” and 1 x “3 1 2”
  - Likely just “3 1 2” is not enough
  - Good RRsets have ***n*** x “3 1 1” and same ***n*** x “3 1 2”
- Which document?
  - SMTP, OPS, SRV, DANEbis

CNAME processing

- Expanded CNAME as preferred TLSA base domain
  - Better support for hosting
  - Kerberos precedent, easier to administer
  - Name checks work with TLD DNAMEs
- Fallback to unexpanded CNAME when expansion is “insecure”

TLSA lookup suppression

- Avoid TLSA lookup
  - When TLSA base domain has “insecure” A/AAAA record or “insecure” CNAME
  - Safe enough:
    - We don't expect DLV between base domain and _port._proto prefix
  - Rationale:
    - “Insecure” DNS load-balancers

Avoid mixed PKI modes

- Not much sense to support both
  - PKIX-TA(0) or PKIX-EE(1),

    AND
  - DANE-TA(2) or DANE-EE(3)
- Either fragile for lack of root CA certs
- Or fragile due to DNSSEC exposure
- Protocol specification or application should choose one pair, not all four.

Normative Language Issues

- Right place for MUST/SHOULD/MAYs?
- Some affect:
  - DANE generic
  - SMTP specifically
  - Operational concerns
- Choices:
  - Put normative generics in SMTP specifically
    - Other protocols will need to copy the text
  - Put normative generics in -ops BCP
  - Put normative generics in DANEbis