redhat.

CURRENT STATE OF DNS, DNSSEC AND DNS PRIVACY

Connections, Bangalore, India
November 8, 2017

Paul Wouters,
RHEL Security

IETF: DNS AREAS

- DNSSEC - Data integrity through insecure intermediaries
- DPRIVE - Data privacy through (semi)trusted intermediary
- DNSOPS - Making it all work in the weirdest networks
- DANE – DNSSEC based authentication
  - TLSA, OPENPGPKEY, SMIMEA, IPSECKEY

DNS TERMINOLOGY

- Good reading to familiarize yourself with DNS:
  - RFC 7719: DNS Terminology
  - draft-ietf-dnsop-terminology-bis

DNSSEC IN ONE SLIDE:

- DNSKEY: Public Key that signed signature in DNS record
- RRSIG: DNS record signatures (itself a DNS record)
- DS: Delegation Signer: expect this hash(DNSKEY) at child zone
- NSEC/NSEC3: Denial of existence (think offline signing)

- ROOT key signs "." (root) and DS below (eg DS "com")
- COM key signs "com." and DS below (eg DS "redhat.com")
- redhat.com key signs "www.redhat.com" A/AAAA records

- Only serve RRSIGs, NSECs if query has DNSSEC OK (DO) bit

DNSSEC IN ONE (OKAY TWO) SLIDES:

- Updating DNSKEY requires updating DS in parent
  - key rollover, slow humans, manual process
- Use two keys:
  - Zone Signing Key (ZSK) for all zone data except DNSKEYs
  - Key Signing Key (KSK) for all DNSKEYs in the zone
  - Roll ZSK monthly or quarterly
    - Parent does not need to be informed
  - Roll KSK yearly (or never-ish)
    - Communicate with parent (using humans or CDS/CDNSKEY)

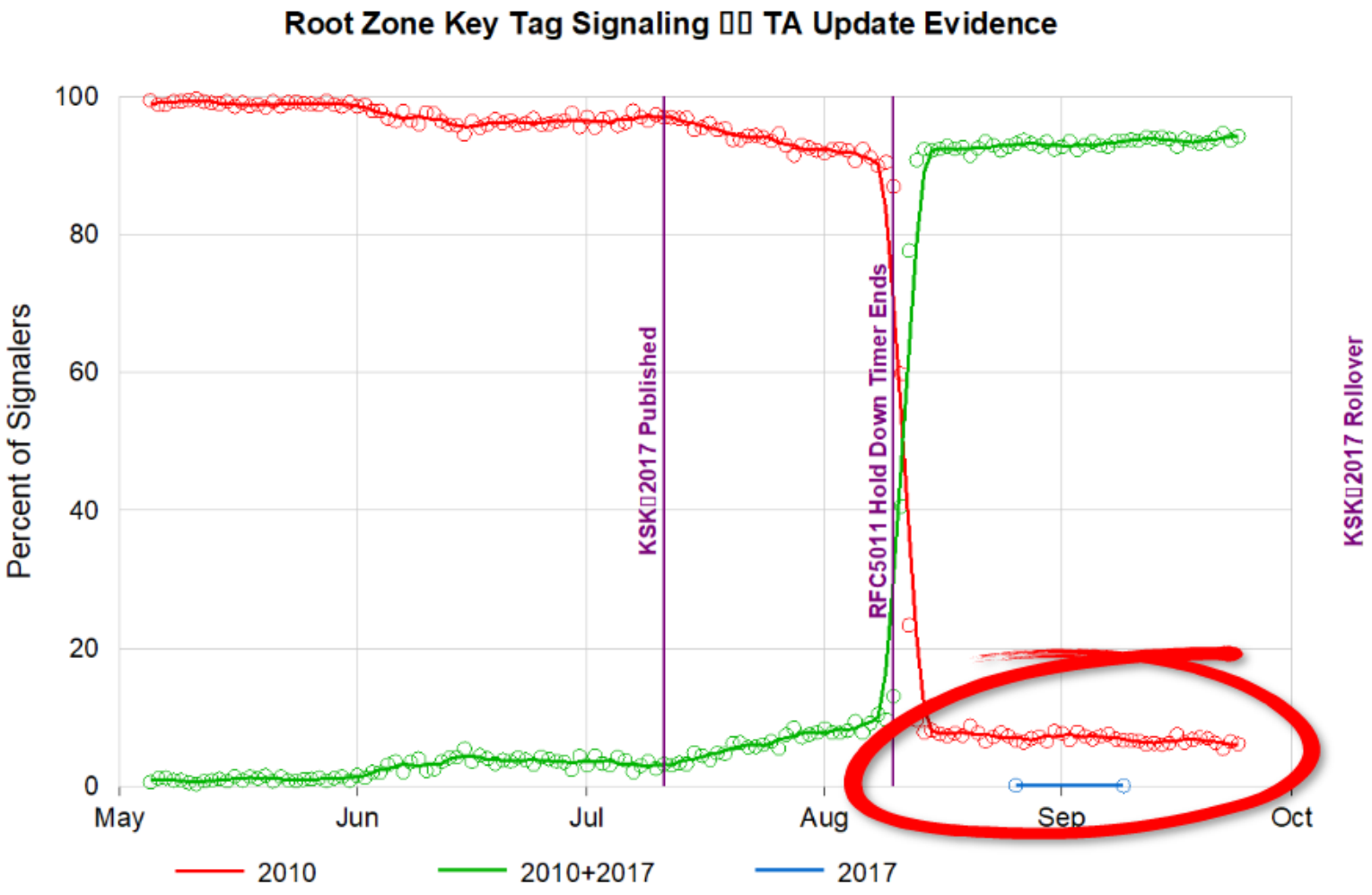
DNSSEC: THE DELAY OF THE DNSSEC ROOT KEY ROLLOVER

- Root wants to rollover KSK
- This key is hardcoded in software
- Key can update in protocol via RFC 5011 support
  - Current key signs future key, must see 30 days
  - Then also trust future key. Revoke bit on old key
  - Switch signing from old to new key
  - Remove old key

- How do we know if resolvers are on new key?

DNSSEC: TWO CANARIES

- RFC 8145: Signaling trust anchor knowledge
- Place EDNS key tag in query for DNSKEY "."
  - EDNS(0) is a hop by hop option
  - Old servers would drop option
  - Middleware boxes are not smart
  - Can show difference between stub and forwarder
- Send special key tag query to auth server for "."
  - Subject to aggressive negative caching


Root Zone Key Tag Signaling □□ TA Update Evidence
Percent of Signalers
100
80
60
40
20
0
May
Jun
Jul
Aug
Sep
Oct
KSK□2017 Published
RFC5011 Hold Down Timer Ends
KSK□2017 Rollover
2010
2010+2017
2017

ANALYSIS BY ICANN OCTO RESEARCH

- 27,084 IP addresses out of 4.2 million sent key tags
- 1,631 (6%) of reporting validators were not ready for KSK roll on October 11, 2017
- Analysis is complicated
  - Dynamic resolver IPs makes situation look worse
  - Resolvers behind forwarders makes situation look better

POSSIBLE CAUSES FOR TRUST FAILURE

- BIND configuration using *trusted-keys* instead of *managed-keys*
- RFC 5011 support requires write access but writing failed
- Puppet and ansible always overwrite old state back
- Containers (eg docker) always start from old state
- Bind sends keytag when DNSSEC disabled
  - Will not do 5011 because disabled, therefore no key update
  - But won't break since not validating (false positive)

DNSSEC: WHAT TO DO WITH LOCAL TRUST ANCHOR AND DS RECORDS?

- Local configuration always trumps DS record
- As long as any trusted key or DS matches, it's OK
- DS trumps local configuration

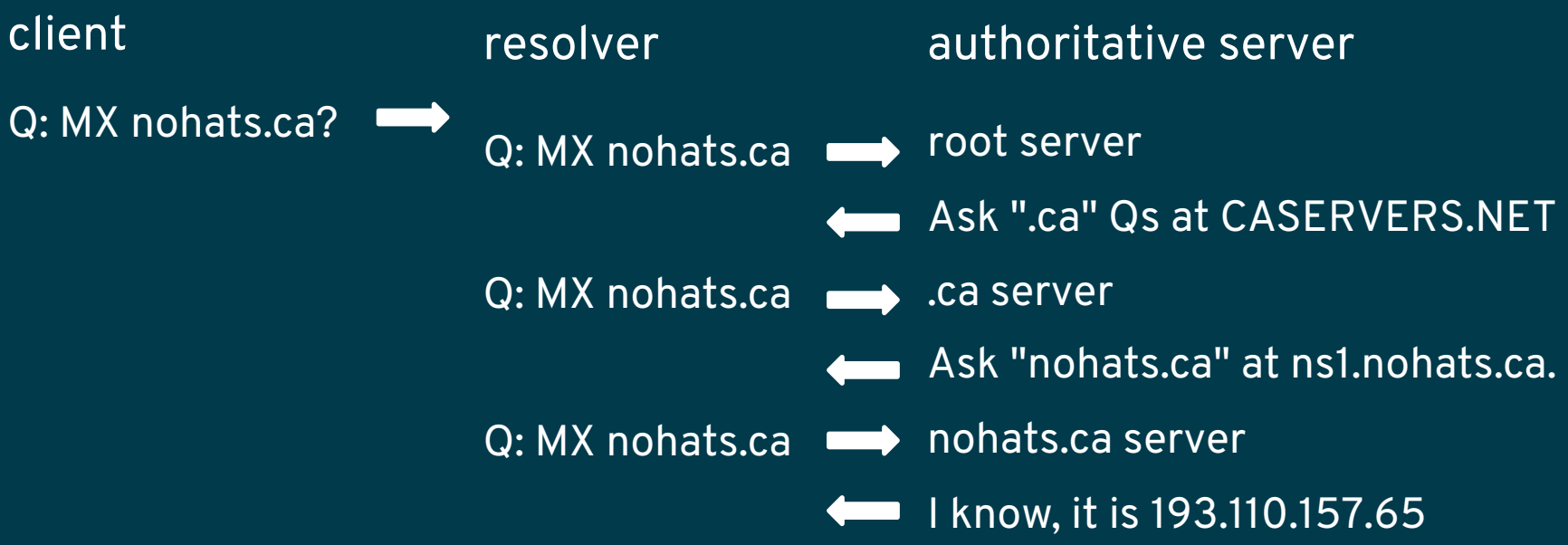
- Use cases:
  - DNSSEC root key rollover (and old config files)
  - Internal-only zones, split DNS views

DPRIVE: DNS QUERY PRIVACY

- DNS over TLS and DTLS (RFC 7858)
  - stub to resolver
    - via trusted party (think 8.8.8.8)
    - via semi-trusted party (think coffee shop chain)
    - opportunistic (encrypt to untrusted party)
  - resolver to authoritative
    - only via trusted keys (published via DNSSEC)
- DNS padding (RFC 7830)
  - EDNS extension to hide DNS packet length

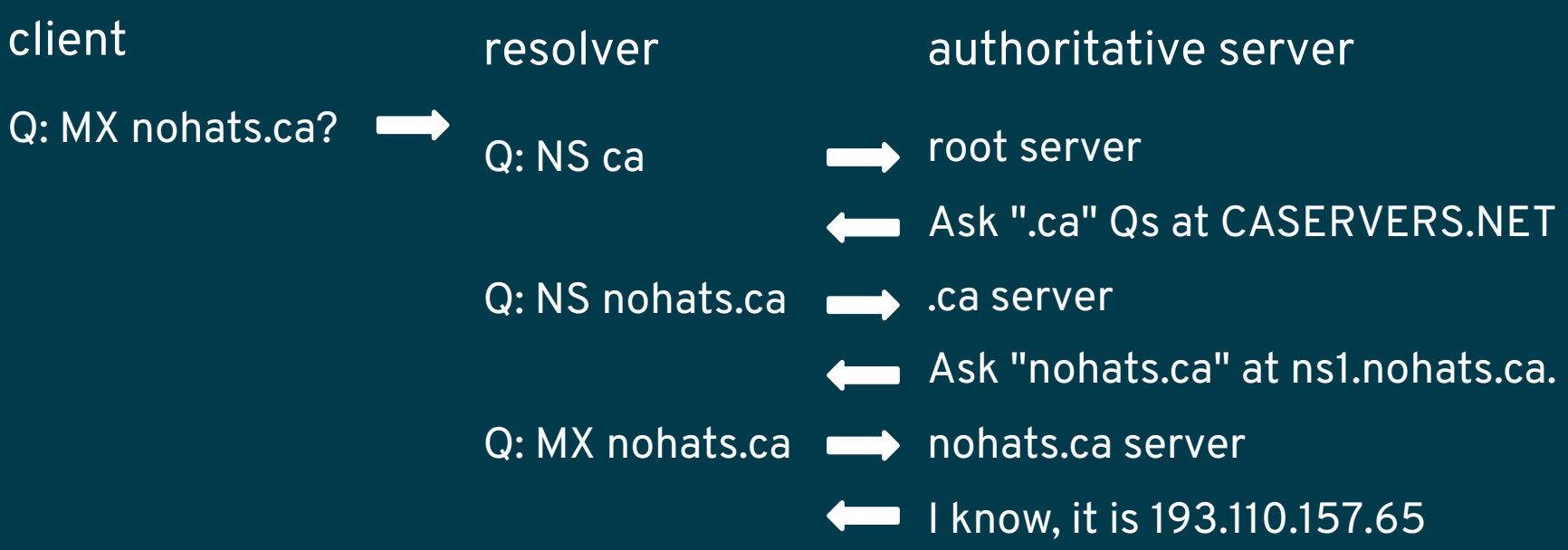
DPRIVE: QUERY MINIMALISATION (RFC 7816)

Traditional query with empty cache:



DPRIVE: QUERY MINIMALISATION (RFC 7816)

Minimized query with empty cache:



THE DRAMA OF SPECIAL NAMES (RFC 6761)

- Apple squatted the .local domain for Rendezvous/Multicast DNS/RFC 6762
- WG published method for requesting Special Name after the fact
- ICANN ran New Generic TLD project - 2,500 domains coming in at $185k
- Problem summary: draft-adpkja-dnsop-special-names-problem

- non-DNS protocols want a name space in the DNS reserved
- DNS names is all that endusers know - there is no real other namespace
- IETF Special Name becomes "free shortcut" overriding ICANN process

- Groups that applied for a Special Name:
  - TOR (.onion)
  - GNUnet (.gnu)
  - Others (.zkey .exit .noconnect .i2p)

- .onion reserved by IETF with RFC 7686, RFC 6761 process frozen afterwards

THE RESPONSE POLICY ZONE (RPZ)

- draft-ietf-dnsop-dns-rpz
- Implements a DNS firewall
- Allows for channels / subscriptions
- Rewrites DNS on the fly
- Breaks DNSSEC
- Goal within IETF: "documents existing practise"
  - Prevent / delay a version that does not break DNSSEC

DNSOPS: THE EDNS CLIENT SUBNET

- RFC 7871: Client Subnet in DNS Queries
- Resolvers can pass on IP address/range of DNS client
- Helps resolver network to resolve to proper geographic region
- Danger of tracking endusers
- client can "opt out" (but can it really?)

- Also "documents existing practise"

RUNNING YOUR OWN ROOT SERVER

- RFC 7706 describes how to run your own
- Free software implementation: http://localroot.isi.edu

MISCELLANEOUS DNS DRAFTS

- draft-ietf-dnsop-extended-error
- draft-ietf-dnsop-let-localhost-be-localhost
- draft-bellis-dnsop-xpf
- draft-fujiwara-dnsop-additional-answers
- draft-ietf-dnsop-rfc5011-security-considerations
- draft-mglt-dnsop-dnssec-validator-requirements
- draft-huston-kskroll-sentinel
- draft-dupont-dnsop-rfc2845bis
- draft-wkumari-dnsop-internal

QUESTIONS?