ATHENE
National Research Center
for Applied Cybersecurity

Long-term Solutions to KeyTrap Vulnerabilities

Elias Heftrig, Haya Schulmann, Niklas Vogel, Michael Waidner

Outline

- Recap on KeyTrap
- Issues with Short-Term Mitigations
- Towards Long-Term Solutions

Recap on KeyTrap Attacks

DoS by DNSSEC Validation

„A potentially Internet-killing vulnerability“

- Internet Pioneer during Disclosure

- High impact
  - Resolvers could be stalled up to 16h with just a single response
- Low resources
  - Host a malicious domain and serve a malicious zone file
- All tested DNSSEC implementations found vulnerable
  - Resolvers, Libraries, Zonefile Checkers, …
- Abundance of vulnerable networks

  → appx. 1/3 of web clients worldwide use validating resolvers
- Patching against KeyTrap required tight coordination with a multi-vendor, >30 heads task force

Highest-Impact Attack Vector

> It is possible for more than one DNSKEY RR to match the conditions above. In this case, the validator cannot predetermine which DNSKEY RR to use to authenticate the signature, and it MUST try each matching DNSKEY RR until either the signature is validated or the validator has run out of matching public keys to try.
>
> RFC4035, Section 5.3.1. "Checking the RRSIG RR Validity"

> When multiple RRSIGs cover a given RRset, Section 5.3.3 of [RFC4035] suggests that "the local resolver security policy determines whether the resolver also has to test these RRSIG RRs and how to resolve conflicts if these RRSIG RRs lead to differing results".
> This document specifies that a resolver SHOULD accept any valid RRSIG as sufficient, and only determine that an RRset is Bogus if all RRSIGs fail validation.
> If a resolver adopts a more restrictive policy, there's a danger that properly signed data might unnecessarily fail validation due to cache timing issues. Furthermore, certain zone management techniques, like the Double Signature Zone Signing Key Rollover method described in Section 4.2.1.2 of [RFC6781], will not work reliably. Such a resolver is also vulnerable to malicious insertion of gibberish signatures.
>
> RFC6840 Section 5.4. "Caution about Local Policy and Multiple RRSIGs"

- "Eager validation" built into DNSSEC
  - Try all possible DNSKEYs for an RRSIG until one works
  - Try all possible RRSIGs for an RRset until one works
- Specification implies complex algorithms over expensive public-key crypto operations → CPU resource exhaustion

Fundamental Problem Exposed by KeyTrap

Openness of DNS(SEC) protocol semantics allows for a plethora of KeyTrap-like attack vectors

- Exploitation of DS hashing and RRSIG validation
- Exploitation of valid and invalid signatures
- Attacks covering different RRsets (cnf. Protocol semantics)

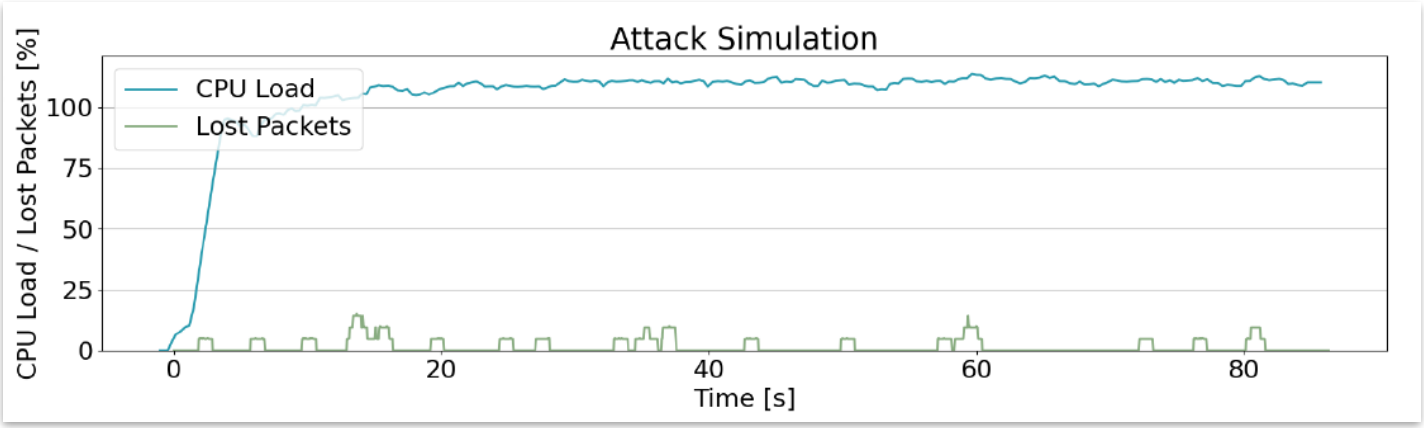
CPU resource exhaustion has never been properly addressed in DNSSEC until KeyTrap

- RFC4033 and RFC4035 generally warned about resource exhaustion attacks
- NSEC3 specification initially addressed resource requirements from SHA1 iteration counts
- Ideation of a DNSKEY-only attack vector by Dutch Bachelor‘s student (not weaponized)
- RRSIG-based CPU resource exhaustion attack exploiting RRSIGs over NSEC RRs in previous work

The *Short-term* Fixes against KeyTrap address the attack vectors but introduce (so-far) unmanaged complexity

Issues with Short-term Mitigations

Architectural Containment



➢ All-valid RRSIGs attack on patched BIND9

Scheduling-based countermeasures

- Intermitting long-running validations to allow other tasks in the pipeline to proceed
  → Still allows waste of (low-priority) CPU cycles – economic attacks?

Limiting Cryptographic Operations

Limits encompass the numbers of …

- RRSIGs tried to validate a given RRset
- DNSKEYs tried with a given RRSIG
- DS RRs tried to validate a given DNSKEY
- Failed or attempted validations per message
- RRSIG and DS validations per resolution

> The amount of work which a resolver will do in response to a client request must be limited to guard against errors in the database, such as circular CNAME references, and operational problems, such as network partition which prevents the resolver from accessing the name servers it needs.
>
> RFC1035, Section 7.1 "Transforming a user request into a query"

→ Per-resolution limits extend general DNS resolver instructions from RFC1035 to DNSSEC

Problems with current per-resolution Limits

Inconsistent selection of limits

- Limits and their values are hardcoded or set by configuration file
- Desirable and (arguably necessary) to adapt to individual resolver requirements
- Problematic in absence of a mechanism to signal and adapt name server responses to these limits

→ factor of unreliability, disincentivizing domain-side use of DNSSEC

Introduces dependencies from DNS to DNSSEC → Layer Violation

- Adds complexity to the already complex DNS(SEC)
- Restrict scalability of DNS (“DNS Security Restrictions”?)
- Hamper future DNS protocol development
- Managing validation complexity in face of (per-resolution) limited validation budgets is challenging

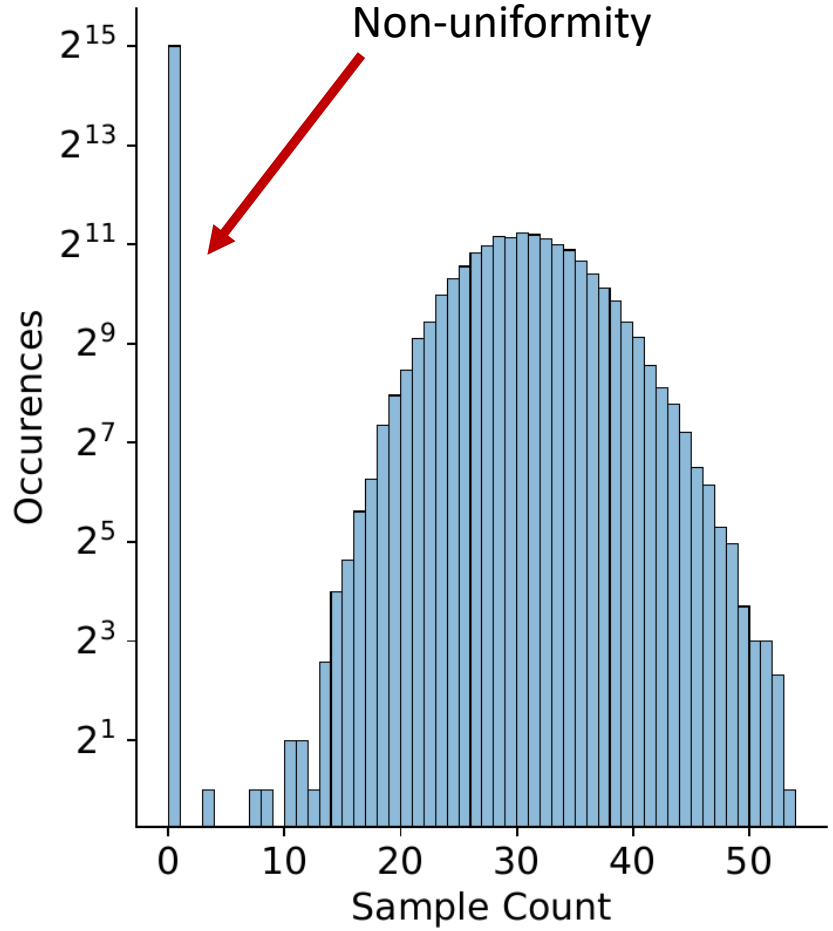
Factors Driving Complexity of Validation

Number of RRsets requiring validation in responses

- Introduction of new record types (e.g. DELEG)
- Elective validation / scrubbing
- Openness to future DNS use cases

KeyTag collisions

- Induce ‘natural’ validation failures
- Make validation complexity a matter of probability
- Tags don’t necessarily follow a uniform random distribution
  - Collision probability depends on DNSSEC algorithm



➢ Frequencies of KeyTag observations in 1M dice-rolled RSASHA256 keys

Elective Validation

```
; <<>> DiG 9.18.28-0ubuntu0.24.04.1-Ubuntu <<>> @ns2.isc.org. +dnssec +nord +nocookie +nocrypto -t MX -q isc.org
;; ANSWER SECTION:
isc.org.                300     IN      MX      5 mx.pao1.isc.org.
isc.org.                300     IN      MX      10 mx.ams1.isc.org.
isc.org.                300     IN      RRSIG   MX 13 2 300 20241020081431 20241006080830 27566 isc.org. [omitted]

;; AUTHORITY SECTION:
isc.org.                7200    IN      NS      ns2.isc.org.
isc.org.                7200    IN      NS      ns1.isc.org.
isc.org.                7200    IN      NS      ns3.isc.org.
isc.org.                7200    IN      NS      ns.isc.afilias-nst.info.
isc.org.                7200    IN      NS      nsp.dnsnode.net.
isc.org.                7200    IN      RRSIG   NS 13 2 7200 20241020081431 20241006080830 27566 isc.org. [omitted]

;; ADDITIONAL SECTION:
ns1.isc.org.            7200    IN      A       149.20.2.26
ns2.isc.org.            7200    IN      A       199.6.1.52
ns3.isc.org.            7200    IN      A       51.75.79.143
ns1.isc.org.            7200    IN      AAAA    2001:500:6b:2::26
ns2.isc.org.            7200    IN      AAAA    2001:500:60:d::52
ns3.isc.org.            7200    IN      AAAA    2001:41d0:701:1100::2c92
ns1.isc.org.            7200    IN      RRSIG   A 13 3 7200 20241020171729 20241006162954 27566 isc.org. [omitted]
ns1.isc.org.            7200    IN      RRSIG   AAAA 13 3 7200 20241020171729 20241006162954 27566 isc.org. [omitted]
ns2.isc.org.            7200    IN      RRSIG   A 13 3 7200 20241020171729 20241006162954 27566 isc.org. [omitted]
ns2.isc.org.            7200    IN      RRSIG   AAAA 13 3 7200 20241020171729 20241006162954 27566 isc.org. [omitted]
ns3.isc.org.            7200    IN      RRSIG   A 13 3 7200 20241020171729 20241006162954 27566 isc.org. [omitted]
ns3.isc.org.            7200    IN      RRSIG   AAAA 13 3 7200 20241020171729 20241006162954 27566 isc.org. [omitted]

;; Query time: 7 msec
;; SERVER: 199.6.1.52#53(ns2.isc.org.) (UDP)
;; WHEN: Fri Oct 11 14:45:22 CEST 2024
;; MSG SIZE  rcvd: 1174
```

- Non-minimal authoritative response from well-managed domain

Factors Driving Complexity of Validation

Crypto-agility

- Future algorithms that increase CPU load may require global revision of local validation limits
- Different crypto libraries varying in CPU requirements

Additional promoters of complexity

- Varying depth of delegation
  - Domains may require multiple resolutions to get resolved (corner case bugs)
- Cross-zone coordination
- Depth of recursion (esp. CNAMEs)

Towards Long-term Solutions

Managing Validation Budgets

Set a global minimum per-resolution validation budget in the specification

- Not considering elective validations or cache
- Reflecting current operational insights and updated over time

→ allows inter-zone budget alignment

- Caveat: needs to consider aliasing

Introduce EDNS0 options to signal…

- Total and current validation budgets from resolvers to name servers
- Validation budget depletion error from resolvers to clients

→ supports global monitoring of validation budgets at domains and Internet nodes

Outlawing KeyTag collisions

- Demand KeyTag to uniquely identify a DNSKEY in a zone
  - Blunt confrontation to RFC4034, requiring the opposite
- Just changing semantics of current records would need worldwide coordination

  → hard to enforce without breaking things

Solution Approaches

- RFC3755-style introduction of new key record type
- New DNSSEC algorithm semantics (credit: M. Andrews)

> However, it is essential to note that the key tag is not a unique identifier. It is theoretically possible for two distinct DNSKEY RRs to have the same owner name, the same algorithm, and the same key tag. The key tag is used to limit the possible candidate keys, but it does not uniquely identify a DNSKEY record. Implementations MUST NOT assume that the key tag uniquely identifies a DNSKEY RR.
>
> RFC4034, Appendix B "Key Tag Calculation"

Thank you!

For more information, see our publications

- CCS '24: The Harder You Try, The Harder You Fail: The KeyTrap Denial-of-Service Algorithmic Complexity Attacks on DNSSEC
- ANRW '24: Protocol Fixes for KeyTrap Vulnerabilities