dnsop
Internet-Draft
Intended status: Standards Track
Expires: August 20, 2017

W. Hardaker
Parsons, Inc.
W. Kumari
Google

February 16, 2017

Security Considerations for RFC5011 Publishers
draft-hardaker-rfc5011-security-considerations-03

Abstract

This document describes the math behind the minimum time-length that a DNS zone publisher must wait before using a new DNSKEY to sign records when supporting the RFC5011 rollover strategies.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 20, 2017.

Copyright Notice

Copyright (c) 2017 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust's Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

1. Introduction 2
   1.1. Requirements notation 3
2. Background 3
3. Terminology 3
4. Timing associated with RFC5011 processing 3
5. Denial of Service Attack Considerations 4
   5.1. Enumerated Attack Example 4
      5.1.1. Attack Timing Breakdown 5
6. Minimum RFC5011 Timing Requirements 6
7. IANA Considerations 7
8. Operational Considerations 7
9. Security Considerations 8
10. Acknowledgements 8
11. Normative References 8
Appendix A. Changes / Author Notes. 8
Authors' Addresses 9

1. Introduction

RFC5011 [RFC5011] defines a mechanism by which DNSSEC validators can extend their list of trust anchors when they've seen a new key published in a zone. However, RFC5011 [intentionally] provides no guidance to the publishers of DNSKEYs about how long they must wait before switching to the newly published key for signing records. Because of this lack of guidance, zone publishers may derive incorrect assumptions about safe usage of the RFC5011 DNSKEY advertising and rolling process. This document describes the minimum security requirements from a publishers point of view and is intended to compliment the guidance offered in RFC5011 (which is written to provide timing guidance solely to the Validating Resolvers point of view).

To verify this lack of understanding is wide-spread, the authors reached out to 5 DNSSEC experts to ask them how long they thought they must wait before using a new KSK that was being rolled according to the 5011 process. All 5 experts answered with an insecure value, and thus we have determined that this lack of operational guidance is causing security concerns today. We hope that this document will rectify this understanding and provide better guidance to zone publishers that wish to make use of the RFC5011 rollover process.

One important note about ICANN's upcoming 2017 KSK rollover plan for the root zone: the timing values chosen for rolling the KSK in the root zone appear completely safe, and are not in any way affected by the timing concerns introduced by this draft

1.1. Requirements notation

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Background

The RFC5011 process describes a process by which a Validating Resolver may accept a newly published KSK as a trust anchor for validating future DNSSEC signed records. This document augments that information with additional constraints, as required from the DNSKEY publication point of view. Note that it does not define any other operational guidance or recommendations about the RFC5011 process from a publication point of view and restricts itself to solely the security and operational ramifications of switching to a new key too soon. Failure of a DNSKEY publisher to follow the minimum recommendations associated with this draft will result in potential denial-of-service attack opportunities against validating resolvers.

3. Terminology

Trust Anchor Publisher The entity responsible for publishing a DNSKEY that can be used as a trust anchor.

4. Timing associated with RFC5011 processing

RFC5011's process of safely publishing a new key and then making use of that key falls into a number of high-level steps:

1. Publish a new DNSKEY in the zone but continue to sign with the old one.

2. Wait a period of time.

3. Begin using the new DNSKEY to sign the appropriate resource records.

4. Optionally mark the older DNSKEY as revoked and publish the revoked key.

This document discusses step 2 of the above process. Some interpretations of RFC5011 have erroneously determined that the wait time is equal to RFC5011's "hold down time".

This document describes an attack based on this (common) erroneous belief, which results in a denial of service attack against the zone if that value is used.

5. Denial of Service Attack Considerations

If an attacker is able to provide a RFC5011 validating engine with past responses, such as when it is in-path or able to otherwise perform any number of cache poisoning attacks, the attacker may be able to leave the RFC5011-compliant validator without an appropriate DNSKEY trust anchor. This scenario will remain until an administrator manually fixes the situation.

The following timeline illustrates this situation.

5.1. Enumerated Attack Example

The following example settings are used in the example scenario within this section:

TTL (all records)	1 day
DNSKEY RRSIG Signature Validity	10 days
Zone resigned every	1 day

Given these settings, the following sequence of events depicts how a Trust Anchor Publisher that waits for only the RFC5011 hold time timer length of 30 days subjects its users to a potential Denial of Service attack. The timing schedule listed below is based on a new trust anchor (a Key Signing Key (KSK)) being published at time T+0. All numbers in this sequence refer to days before and after such an event. Thus, T-1 is the day before the introduction of the new key, and T+15 is the 15th day after the key was introduced into the fictitious zone being discussed.

In this dialog, we consider two keys being published:

K_old	The older KSK being replaced.
K_new	The new KSK being transitioned into active use, using the RFC5011 process.

In this dialog, the following actors are playing roles in this situation:

Zone Signer	The owner of a zone intending to publish a new Key-Signing-Key (KSK) that will become a trust anchor by validators following the RFC5011 process.
RFC5011 Validator	A DNSSEC validator that is using the RFC5011 processes to track and update trust anchors.

Attacker An attacker intent on foiling the RFC5011 Validator's ability to successfully adopt the Zone Signer's K_new key as a new trust anchor.

5.1.1. Attack Timing Breakdown

The following series of steps depicts the timeline in which an attack occurs that foils the adoption of a new DNSKEY by a Trust Anchor Publisher that revokes the old key too quickly.

T-1 The last signatures are published by the Zone Signer that signs only K_old using K_old. The Attacker queries for, retrieves and caches this keyset and corresponding signatures.

T-0 The Zone Signer adds K_new to his zone and signs the zone's key set with K_old. The RFC5011 Validator retrieves the new key set and corresponding signature set and notices the publication of K_new. The RFC5011 Validator starts the (30-day) hold-down timer for K_new.

T+5 The RFC5011 Validator queries for the zone's keyset per the Active Refresh schedule, discussed in Section 2.3 of RFC5011. Instead of receiving the intended published keyset, the Attacker successfully replays the keyset and associated signatures that they recorded at T-1. Because the signature lifetime is 10 days (in this example), the replayed signature and keyset is accepted as valid (being only 6 days old) and the RFC5011 Validator cancels the hold-down timer for K_new.

T+10 The RFC5011 Validator queries for the zone's keyset and discovers K_new again, signed by K_old (the attacker is unable to replay the records cached at T-1, because they have now expired). The RFC5011 Validator starts (anew) the hold-timer for K_new.

T+15,T+20, and T+25 The RFC5011 Validator continues checking the zone's key set and lets the hold-down timer keep running without resetting it.

T+30 The Zone Signer knows that this is the first time at which some validators might accept K_new as a new trust anchor, since the hold-down timer of a RFC5011 Validator not under attack that had queried and retrieved K_new at T+0 would now have reached 30 days. However, the hold-down timer of our attacked RFC5011 Validator is only at 20 days.

T+35 The Zone Signer (mistakenly) believes that all validators following the Active Refresh schedule (Section 2.3 of RFC5011) should have accepted K_new as a the new trust anchor (since the

hold down time of 30 days + 1/2 the signature validity period would have passed). However, the hold-down timer of our attacked RFC5011 Validator is only at 25 days; The replay attack at T+5 means its new hold-time timer actually started at T+10, and thus at this time it's real hold-down timer is at T+35 - T+10 = 25 days, which is less than the RFC5011 required 30 days.

T+36 The Zone Signer, believing K_new is safe to use, switches their active signing KSK to K_new and publishes a new DNSKEY set signature signed with K_new. Non-attacked RFC5011 validators, with a hold-down timer of at least 30 days, would have accepted K_new into their set of trusted keys. But, because our attacked RFC5011 Validator still has a hold-down timer for K_new at 26 days, it will fail to accept K_new as a trust anchor and since K_old is no longer being used, all the KSK records from the zone signed by K_new will be treated as invalid. Subsequently, all keys in the key set are now unusable, invalidating all records in the zone of any type and name.

6. Minimum RFC5011 Timing Requirements

Given the attack description in Section 5, the correct minimum length of time required for the Zone Signer to wait before using K_new is:

```
waitTime = addHoldDownTime
           + MIN(3 * (DNSKEY RRSIG Signature Validity) / 2,
                 3 * MAX(original TTL of K_old DNSKEY RRSet) / 2,
                 15 days)
           + 2 * MAX(TTL of all records)
```

The most confusing element of the above equation comes from the "3 * (DNSKEY RRSIG Signature Validity) / 2" element, but is the most critical to understand and get right.

The RFC5011 "Active Refresh" requirements state that:

> A resolver that has been configured for an automatic update of keys from a particular trust point MUST query that trust point (e.g., do a lookup for the DNSKEY RRSet and related RRSIG records) no less often than the lesser of 15 days, half the original TTL for the DNSKEY RRSet, or half the RRSIG expiration interval and no more often than once per hour.

The important timing constraint that must be considered is the last point at which a validating resolver may have received a replayed the original DNSKEY set (K_old) without the new key. It's the next query of the RFC5011 validator that the assured K_new will be seen. For the following paragraph, we use the more common case where (DNSKEY RRSIG Signature Validity) is greater than (original TTL for the DNSKEY RRSet), although the equation above takes both cases into account.

Thus, the worst case scenario of this attack is when the attacker can replay K_old at just before (DNSKEY RRSIG Signature Validity). If a RFC5011 validator picks up K_old at this this point, it will not have a hold down timer started at all. It's not until the next "Active Refresh" time that they'll pick up K_new with assurance, and thus start their hold down timer. Thus, this is not at (DNSKEY RRSIG Signature Validity) time past publication, but rather 3 * (DNSKEY RRSIG Signature Validity) / 2.

The extra 2 * MAX(TTL of all records) is the standard added safety margin when dealing with DNSSEC due to caching that can take place. Because the 5011 steps require direct validation using the signature validity, the authors aren't yet convinced it is needed in this particular case.

For the parameters listed in Section 5.1, our example:

```
waitTime = 30
           + 3 * (10) / 2
           + 2 * (1)        (days)

waitTime = 47               (days)
```

This hold-down time of 47 days is 12 days longer than the frequently perceived 35 days in T+35 above.

7. IANA Considerations

This document contains no IANA considerations.

8. Operational Considerations

A companion document to RFC5011 was expected to be published that describes the best operational practice considerations from the perspective of a zone publisher and Trust Anchor Publisher. However, this companion document was never written. The authors of this document hope that it will at some point in the future, as RFC5011 timing can be tricky as we have shown. This document is intended

only to fill a single operational void that results in security ramifications (specifically a denial of service attack against an RFC5011 Validator). This document does not attempt to document any other missing operational guidance for zone publishers.

9. Security Considerations

This document, is solely about the security considerations with respect to the Trust Anchor Publisher of RFC5011 trust anchors / keys. Thus the entire document is a discussion of Security Considerations

10. Acknowledgements

The authors would like to especially thank to Michael StJohns for his help and advice. We would also like to thank Bob Harold, Shane Kerr, Matthijs Mekking, Duane Wessels, and everyone else who assisted with this document.

11. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <http://www.rfc-editor.org/info/rfc2119>.

[RFC5011] StJohns, M., "Automated Updates of DNS Security (DNSSEC) Trust Anchors", STD 74, RFC 5011, DOI 10.17487/RFC5011, September 2007, <http://www.rfc-editor.org/info/rfc5011>.

Appendix A. Changes / Author Notes.

From -00 to -02:

Additional background and clarifications in abstract.

Better separation in attack description between attacked and non-attacked resolvers.

Some language cleanup.

Clarified that this is maths (and math is hard, let's go shopping!)

Changed to " <?rfc include='reference....'?> " style references.

From -02 to -03:

Minor changes from Bob Harold

Clarified why 3/2 signature validity is needed

Changed min wait time math to include TTL value as well

Authors' Addresses

Wes Hardaker
Parsons, Inc.
P.O. Box 382
Davis, CA 95617
US

Email: ietf@hardakers.net

Warren Kumari
Google
1600 Amphitheatre Parkway
Mountain View, CA 94043
US

Email: warren@kumari.net