                    **DNSSEC Validators Requirements**
            **draft-mglt-dnsop-dnssec-validator-requirements-00.txt**


Abstract

   DNSSEC provides data integrity and authentication for DNSSEC
   validators.  However, without valid trust anchor(s) and an acceptable
   value for the current time, DNSSEC validation cannot be performed.
   This document lists the requirements to be addressed so resolvers can
   have DNSSEC validation can be always-on.

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 17, 2014.

Copyright Notice

Table of Contents

## 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.  Introduction

   DNSSEC [RFC4033], [RFC4034], [RFC4035] adds data authentication and
   integrity checks to DNS [RFC1034], [RFC1035].  For signature
   validation, DNSSEC requires a valid trust anchor such as the Key
   Signing Key (KSK) (the Root Zone KSK for example) and an appropriated
   time.

   Currently few effort have been made to describe mechanisms that
   guarantee how a DNSSEC validator can be provisioned with the
   appropriated KSKs and time so that DNSSEC validation can always be
   activated.  A device that is badly configured or provisioned
   performing DNSSEC validation may result in disabling the DNS service
   of the device, and then most of its communications.  As a results,
   non administrated devices that implement DNSSEC validation always
   need heuristics that disable the DNSSEC validation.  This results in
   an implicit rule that can be stated as: "if DNSSEC validation is

performed correctly then do DNSSEC otherwise disable validation and
switch to DNS".  In a security point of view, this is unacceptable.

This document considers unmanaged devices performing DNSSEC
validation and details scenarios where DNSSEC validation cannot be
performed properly by the device.  In other words, this means that
FQDN properly signed are rejected.  From these scenarios, this
document derives requirements so DNSSEC Validators can have DNSSEC
always activated.

## 3.  Terminology

This document uses the following terminology:

- DNSSEC Validator:  the entity that performs DNSSEC resolution and
     performs signature validation.

## 4.  Time derivation and absence of Real Time Clock

With M2M communication some devices are not expecting to embed Real
Time Clock (Rasberry Pi is one example of such devices).  When these
devices are re-plugged the initial time is set to January 1 1970.
Other devices that have clocks that may suffer from time derivation.
All these devices cannot rely on their time estimation to perform
DNSSEC validation.

Requirement 1: DNSSEC validator MUST be provided means to
appropriately update their time.

## 5.  Unplugged devices during Trust Anchor KSKs roll over

In this section we consider a regular Trust Anchor KSK roll over as
described in [RFC6781] and [RFC5011].  Unlike regular KSKs, Trust
Anchor KSK does not have to update the DS RRset in the parent zone.
According to [RFC6781], if TTL_K is the TTL associated to the Trust
Anchor KSK and associated RRSIGs, the time of key roll over is around
TTL_K with double signed KSKs, and 2 x TTL_K in the case of single
singed KSK.

The Root Zone KSK is an example of Trust Anchor KSK and at the time
of writing the KSK has a TTL of 172800 seconds which means 2 days.
This means that 2 days would be sufficient to perform a Trust Anchor
KSK roll over.  [RFC5011] recommends to advertise the new / old key
for 30 days.  This means that a device unplugged for two months may
not be aware of a regular Trust Anchor KSK rollover.

A DNSSEC validation may be properly configured by the manufacturer to
perform DNSSEC validation.  The device may for example be configured

by the manufacturer shipped to resellers that store the device for a
few months, years before selling the devices to the final end user.
Similarly, an operational device may be unplugged for a while for
maintenance reason, or to be used as a fall back.  This fall back is
never expect to happen and may happens years after.

Suppose a KSK complete key roll-over occurs (for example at the Root
Zone) while the device is offline.  Once plugged again, the device
will attempt to validate DNSSEC signature with the old Trust Anchor
KSK.

The key point in this example is that the device is booted and does
not rely on cached information.

Requirement 2: DNSSEC Validator MUST be able to check the validity of
their Trust Anchor KSKs.

Requirement 3: DNSSEC Validator MUST be able to retrieve their Trust
Anchor KSKs.

We think these requirements are not restrictive to the Root Zone KSK,
but to any KSK.  In fact 1) because it is not always possible to
build a trusted delegation between the Root Zone and any sub zone.
Then 2) the chain of trust between a zone and the root zone may not
be always there.  At last, 3) the DNS(SEC) resolution may be critical
for infrastructure management.  A company "Example" may for example
address all its devices under the domain example.com and may not want
disruption to happen if the .com delegation cannot be validated for
any reason.  Such companies may provision there DNSSEC Validator with
the Trust Anchor KSK for the zone example.com in addition to the
regular DNSSEC delegation.

Note that providing Trust Anchor KSKs is a crucial operation and can
be used a vector of attack.  As a result, this operation MUST be
performed cautiously.

## 6.  Emergency Key rollover

By emergency key roll over, this paper designates any rollover that
are not performed as described in Section 4.1 of [RFC6781] and that
results in differences between data stored in the cache of the DNSSEC
Validators and the authoritative servers (see section 4.2 in
[RFC6781])

Emergency key roll over can be intentionally performed or result from
a unexpected behavior in the publishing/validation chain.  This is
out of scope of this paper to understand the reasons/motivation for
such key roll over.  This document assumes such situation are likely

   to happen and list the requirement so DNSSEC Validator can recover
   from such situations.

## 6.1.  Invalid cached ZSK

   An emergency ZSK rollover may result in a new ZSK with associated new
   RRSIG published in the authoritative zone, while DNSSEC Validator may
   still cache the old value of the ZSK.  For a RRset not cached, the
   DNSSEC Validator performs a DNSSEC query to the authoritative server
   that returns the RRset signed with the new ZSK.  The DNSSEC Validator
   validates the signature with the old ZSK which results in an invalid
   signature check.

   Suppose that the old ZSK has been corrupted and that old RRsets have
   been spoofed.  Until the ZSK TTL expires, the DNSSEC Validator
   considers the spoofed RRsets as valid and the newly signed RRsets as
   invalid.

   Requirement 4: DNSSEC Validator MUST be able to be informed a ZSK
   MUST be flushed from cache.

   Note that if the DNSSEC Validator receives an indication that a ZSK
   is not valid anymore, it is expected to flush its cache entries of
   the old ZSK as well as all entries that have been validated by the
   old ZSK.  This does not lead to impersonation of ZSK, at most it
   generates some additional DNSSEC resolutions and validations.

## 6.2.  Invalid cached RRSIG

   This would mean the DNSSEC Validator caches a new ZSK, but still has
   a RRset with a RRSIG signed with the old ZSK.

   This situation should not happen as when a ZSK is renewed all RRsets
   validated by the old ZSK are flushed from the cache.

## 6.3.  Invalid cached KSK

   Consequences of invalid KSK are similar to ZSK.  None of the RRSIG
   can be validated even the one signing the ZSK. (cf. Section 6.1)

   Requirement 5: DNSSEC Validator MUST be able to be informed a KSK
   MUST be flushed from cache.

## 6.4.  Invalid DS

   The DS RRset is stored in the parent zone to build a chain of trust
   with the child zone.  This DS RRset can be invalid because its RDATA

(KSK) is not anymore used in the child zone or because the DS is
badly signed and cannot be validated by the DNSSEC Validator.

In both cases the child zone is considered as insecure and the valid
child zone's KSK should become a Trust Anchor KSK.

Requirement 6: DNSSEC Validator MUST be able to be informed a KSK
SHOULD be trusted as a Trust Anchor KSK.

## 7.  Invalid RRSIG

A zone may have been badly signed, which means that the KSK or ZSK
cannot validate the RRSIG associated to the RRsets.  This may not be
due to a key roll over, but to an incompatibility between the keys
(KSK or ZSK) and the signatures.

Requirement 7: DNSSEC Validator MUST be able to be informed that a
KSK or a ZSK MUST NOT be used for RRSIG validation.  Unlike
"flushing", "MUST NOT be used" means the issue is not a
synchronization issue, but that legitimate keys are invalid.  Such
Keys are known as Negative Trust Anchors
[I-D.livingood-negative-trust-anchors].

This means that the zone for a given time will be known as "known
insecure".  The DNSSEC Validator is not expected to perform signature
validation for this zone.  It is expected that this information is
associated to a Time To Live.

Note that, this information may be used as an attack vector to
impersonate a zone, and must be provided in a trusted way, by a
trusted party.

If a zone has been badly signed, the administrator of the
authoritative DNS server may resign the zone with the same keys or
proceed to an emergency key rollover.  If the signature is performed
with the same keys, the DNSSEC Validator may notice by itself that
RRSIG can be validated.  On the other hand if a key rollover is
performed, the newly received RRSIG will carry a new key id.  Upon
receiving a new key id in the RRSIG, the DNSSEC Validator is expected
to retrieve the new ZSK/KSK.  If the RRSIG can be validated, the
DNSSEC Validator is expected to remove the "known insecure" flag.

However, if the KSK/ZSK are rolledover and RRSIG cannot be validated,
it remains hard for the DNSSEC Validator to determine whether the
RRSIG cannot be validated or that RRSIG are invalid.  As a result:

Requirement 8: The DNSSEC Validator MUST be able to be informed that
a KSK or a ZSK is known "back to secure".

8.  Private KSK

   DNS(SEC) authoritative server are expected to provide public data.
   Corporate networks or home networks may want to provide different
   views for their intranet and the public Internet.  This was commonly
   known as "split-dns".  This means that private networks like home
   networks or corporate networks will not be able to use DNSSEC on the
   private view.

   Requirement 9: DNSSEC Validator MUST be able to be provided KSK for
   private use.

   In that case, the DNSSEC Validator may have two KSKs, one for the DNS
   public view and one for the private view.

9.  Requirements

   The document lists the following requirements:

   - Requirement 1:  DNSSEC validator MUST be provided means to
        appropriately update their time.

   - Requirement 2:  DNSSEC Validator MUST be able to check the validity
        of their Trust Anchor KSKs.

   - Requirement 3:  DNSSEC Validator MUST be able to retrieve their
        Trust Anchor KSKs.

   - Requirement 4:  DNSSEC Validator MUST be able to be informed a ZSK
        MUST be flushed from cache.

   - Requirement 5:  DNSSEC Validator MUST be able to be informed a KSK
        MUST be flushed from cache.

   - Requirement 6:  DNSSEC Validator MUST be able to be informed a KSK
        SHOULD be trusted as a Trust Anchor KSK.

   - Requirement 7:  DNSSEC Validator MUST be able to be informed that a
        KSK or a ZSK MUST NOT be used for RRSIG validation.

   - Requirement 8:  The DNSSEC Validator MUST be able to be informed
        that a KSK or a ZSK is known "back to secure".

   - Requirement 9:  DNSSEC Validator MUST be able to be provided KSK
        for private use.

## 10.  IANA Considerations

   There are no IANA consideration for this document.

## 11.  Security Considerations

   There is no security consideration for this document.

## 12.  Acknowledgment

   The need to address DNSSEC issues on the resolver side started in the
   Home Networks mailing list and during the IETF87 in Berlin.  Among
   others, people involved in the discussion were Ted Lemon, Ralph
   Weber, Normen Kowalewski, and Mikael Abrahamsson.  People involved in
   the email discussion initiated by Jim Gettys were, with among others,
   Paul Wouters, Joe Abley and Michael Richardson.

   The current document has been initiated after a discussion with Paul
   Wouter and Evan Hunt.

## 13.  References

### 13.1.  Normative References

   [RFC1034]  Mockapetris, P., "Domain names - concepts and facilities",
              STD 13, RFC 1034, November 1987.

   [RFC1035]  Mockapetris, P., "Domain names - implementation and
              specification", STD 13, RFC 1035, November 1987.

   [RFC2119]  Bradner, S., "Key words for use in RFCs to Indicate
              Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC4033]  Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "DNS Security Introduction and Requirements", RFC
              4033, March 2005.

   [RFC4034]  Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Resource Records for the DNS Security Extensions",
              RFC 4034, March 2005.

   [RFC4035]  Arends, R., Austein, R., Larson, M., Massey, D., and S.
              Rose, "Protocol Modifications for the DNS Security
              Extensions", RFC 4035, March 2005.

   [RFC5011]  StJohns, M., "Automated Updates of DNS Security (DNSSEC)
              Trust Anchors", STD 74, RFC 5011, September 2007.

**13.2**.  **Informational References**

[I-D.livingood-negative-trust-anchors]
           Livingood, J. and C. Griffiths, "Definition and Use of
           DNSSEC Negative Trust Anchors", draft-livingood-negative-
           trust-anchors-06 (work in progress), February 2013.

   [RFC6781]  Kolkman, O., Mekking, W., and R. Gieben, "DNSSEC
           Operational Practices, Version 2", RFC 6781, December
           2012.

**Appendix A**.  **Document Change Log**

   [RFC Editor: This section is to be removed before publication]

   -00: First version published.

Author's Address

   Daniel Migault
   Orange
   38 rue du General Leclerc
   92794 Issy-les-Moulineaux Cedex 9
   France

   Phone: +33 1 45 29 60 52
   Email: daniel.migault@orange.com