            Certification Authority (CA) Key Rollover in
              the Resource Public Key Infrastructure (RPKI)

Abstract

   This document describes how a Certification Authority (CA) in the
   Resource Public Key Infrastructure (RPKI) performs a planned rollover
   of its key pair.  This document also notes the implications of this
   key rollover procedure for relying parties (RPs).  In general, RPs
   are expected to maintain a local cache of the objects that have been
   published in the RPKI repository, and thus the way in which a CA
   performs key rollover impacts RPs.

Status of This Memo

   This memo documents an Internet Best Current Practice.

   This document is a product of the Internet Engineering Task Force
   (IETF).  It represents the consensus of the IETF community.  It has
   received public review and has been approved for publication by the
   Internet Engineering Steering Group (IESG).  Further information on
   BCPs is available in Section 2 of RFC 5741.

   Information about the current status of this document, any errata,
   and how to provide feedback on it may be obtained at
   http://www.rfc-editor.org/info/rfc6489.

Table of Contents

1.  Introduction

   This document describes an algorithm to be employed by a
   Certification Authority (CA) in the Resource Public Key
   Infrastructure (RPKI) [RFC6480] to perform a rollover of its key
   pair.

   This document defines a conservative procedure for such entities to
   follow when performing a key rollover.  This procedure is
   "conservative" in that the CA's actions in key rollover are not
   intended to disrupt the normal operation of relying parties (RPs) in
   maintaining a local cached version of the RPKI distributed
   repository.  Using this procedure, RPs are in a position to be able
   to validate all authentic objects in the RPKI using the validation
   procedure described in [RFC6480] at all times.

1.1.  Terminology and Concepts

   It is assumed that the reader is familiar with the terms and concepts
   described in "Internet X.509 Public Key Infrastructure Certificate
   and Certificate Revocation List (CRL) Profile" [RFC5280], "X.509
   Extensions for IP Addresses and AS Identifiers" [RFC3779], the
   profile for RPKI Certificates [RFC6487], and the RPKI repository
   structure [RFC6481] .

   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.  CA Key Rollover Procedure

   A CA in the RPKI is an entity that issues CA and end-entity (EE)
   certificates and Certificate Revocation Lists (CRLs).  A CA instance
   is associated with a single key pair [RFC6487], implying that if key
   rollover is a regularly scheduled event, then, over time, there will
   be many CA instances.  The implication in the context of key rollover
   is that, strictly speaking, a CA does not perform a key rollover per
   se.  In order to perform the equivalent of a key rollover, the CA
   creates a "new" instance of itself, with a new key pair, and then
   effectively substitutes this "new" CA instance into the RPKI
   hierarchy in place of the "old" CA instance.

   Note that focus of this procedure is planned key rollover, not an
   emergency key rollover, e.g., promoted by a suspected or detected
   private key compromise.  However, the procedure described here is
   applicable in emergency key rollover situations, with the exception
   of the "Staging Period" duration.

   There are several considerations regarding this procedure that MUST
   be followed by a CA performing a key rollover operation.  The
   critical consideration is that the RPKI has potential application in
   the area of control of routing integrity [RFC6480], and key rollover
   should not cause any transient hiatus in which an RP is led to
   incorrect conclusions regarding the authenticity of attestations made
   in the context of the RPKI.  A CA cannot assume that all RPs will
   perform path validation and path discovery in the same fashion;
   therefore, the key rollover procedure MUST preserve the integrity of
   the CRL Distribution Points (CRLDP), Subject Information Access
   (SIA), and Authority Information Access (AIA) pointers in RPKI
   certificates.

   In the procedure described here, the CA creates a "new" CA instance,
   and has the associated new public key published in the form of a
   "new" CA certificate.  While the "current" and "new" CA instances
   share a single repository publication point, each CA has its own CRL
   and its own manifest.  Initially, the "new" CA publishes an empty CRL
   and a manifest that contains a single entry for the CRL.  The
   "current" CA also maintains its published CRL and manifest at this
   repository publication point.

   The CA performing key rollover waits for a period of time to afford
   every RP an opportunity to discover and retrieve this "new" CA
   certificate, and store it in its local RPKI repository cache
   instance.  This period of time is termed the Staging Period.  During
   this period, the CA will have a "new" CA instance, with no
   subordinate products, and a "current" CA instance that has issued all
   subordinate products.  At the expiration of the Staging Period, the

"new" CA instance MUST replace all (valid) subordinate products of
the "current" CA instance, overwriting the "current" subordinate
products in the CA's repository publication point.  When this process
is complete, the "current" CA instance is retired, and the "new" CA
instance becomes the "current" CA.

During the transition of the "current" and "new" CA instances, the
"new" CA instance MUST reissue all subordinate products of the
"current" CA.  The procedure described here requires that, with the
exception of manifests and CRLs, the reissued subordinate products be
published using the same repository publication point object names,
effectively overwriting the old objects with these reissued objects.
The intent of this overwriting operation is to ensure that the AIA
pointers of subordinate products at lower tiers in the RPKI hierarchy
remain correct, and that CA key rollover does not require any
associated actions by any subordinate CA.

There are three CA states described here:

CURRENT:
    The CURRENT CA is the active CA instance used to accept and
    process certificate issuance and revocation requests.  The
    starting point for this algorithm is that the key of the CURRENT
    CA is to be rolled over.

NEW:
    The NEW CA is the CA instance that is being created.  The NEW CA
    is not active, and thus does not accept nor process certificate
    issuance and revocation requests.  The NEW CA SHOULD issue a CRL
    and an EE certificate in association with its manifest to provide
    a trivial, complete, and consistent instance of a CA.

OLD:
    The CA instance is in the process of being removed.  An OLD CA
    instance is unable to process any certificate issuance and
    revocation requests.  An OLD CA instance will continue to issue
    regularly scheduled CRLs and issue an EE certificate as part of
    the process of updating its manifest to reflect the updated CRL.

    To perform a key rollover operation, the CA MUST perform the
    following steps in the order given here.  Unless specified
    otherwise each step SHOULD be performed without any intervening
    delay.  The process MUST be run through to completion.

    1. Generate a new key pair for use by the NEW CA.  Because the
       goal of this algorithm is key rollover, the key pair generated
       in this step MUST be different from the pair in use by the
       CURRENT CA.

2. Generate a certificate request with this key pair and pass the
   request to the CA that issued the CURRENT CA certificate.  This
   request MUST include the same SIA extension that is present in
   the CURRENT CA certificate.  This request, when satisfied, will
   result in the publication of the NEW CA certificate.  This
   (NEW) CA certificate will contain a subject name selected by
   the issuer, which MUST be distinct from the subject name used
   in the CURRENT CA certificate.  The Certificate Practice
   Statement (CPS) for the issuer of the NEW CA certificate will
   indicate the time frame within which a certificate request is
   expected to be processed.

3. Publish the NEW CA's CRL and manifest.

   The steps involved here are:

   -  Wait for the issuer of the NEW CA to publish the NEW CA
      certificate.

   -  As quickly as possible following the publication of the NEW
      CA certificate, use the key pair associated with the NEW CA
      to generate an initially empty CRL, and publish this CRL in
      the NEW CA's repository publication point.  It is
      RECOMMENDED that the CRL for the NEW CA have a nextUpdate
      value that will cause the CRL to be replaced at the end of
      the Staging Period (see in Step 4 below).

   -  Generate a new key pair, and generate an associated EE
      certificate request with an AIA value of the NEW CA's
      repository publication point.  Pass this EE certificate
      request to the NEW CA, and use the returned (single-use) EE
      certificate as the NEW CA's manifest EE certificate.

   -  Generate a manifest containing the new CA's CRL as the only
      entry, and sign it with the private key associated with the
      manifest EE certificate.  Publish the manifest at the NEW
      CA's repository publication point.

   -  Destroy the private key associated with the manifest EE
      certificate.

4. The NEW CA enters a Staging Period.  The duration of the
   Staging Period is determined by the CA, but it SHOULD be no
   less than 24 hours.  The Staging Period is intended to afford
   an opportunity for all RPs to download the NEW CA certificate
   prior to publication of certificates, CRLs, and RPKI signed
   objects under the NEW CA.  During the Staging Period, the NEW
   CA SHOULD reissue, but not publish, all of the products that

were issued under the CURRENT CA.  This includes all CA
certificates, EE certificates, and RPKI signed objects.
Section 4 describes how each reissued product relates to the
product that it replaces.  During the Staging Period, the
CURRENT CA SHOULD continue to accept and process certificate
issuance requests and MUST continue to accept and process
certificate revocation requests.  If any certificates are
issued by the CURRENT CA during the Staging Period, they MUST
be reissued under the NEW CA during this period.  Any
certificates that are revoked under the CURRENT CA MUST NOT be
reissued under the NEW CA.  As noted above, in the case of an
emergency key rollover, a CA will decide whether the 24 hour
minimal Staging Period interval is appropriate, or if a shorter
Staging Period is needed.  As the Staging Period imposes no
additional burden on Relying Parties, there is no stipulated or
recommended maximum Staging Period.

5. Upon expiration of the Staging Period, the NEW CA MUST publish
   the signed products that have been reissued under the NEW CA,
   replacing the corresponding products issued under the CURRENT
   CA at the NEW CA's repository publication point.  This
   replacement is implied by the file naming requirements imposed
   by [RFC6481] for these signed products.  The trivial manifest
   for the NEW CA (which contained only one entry, for the NEW
   CA's CRL) is replaced by a manifest listing all of these
   reissued, signed products.  At this point, the CURRENT CA
   becomes the OLD CA, and the NEW CA becomes the CURRENT CA.  Use
   the OLD CA to issue a manifest that lists only the OLD CA's
   CRL.  It is anticipated that this step is very brief, perhaps a
   few minutes in duration, because the CA has reissued all of the
   signed products during the Staging Period.  Nonetheless, it is
   desirable that the activities performed in this step be viewed
   as atomic by RPs.

6. Generate a certificate revocation request for the OLD CA
   certificate and submit it to the issuer of that certificate.
   When the OLD CA certificate is revoked, the CRL for the OLD CA
   is removed from the repository, along with the manifest for the
   OLD CA.  The private key for the OLD CA is destroyed.

3.  Relying Party Requirements

   This procedure defines a Staging Period for CAs performing a key
   rollover operation.  This period is defined as a period no shorter
   than 24 hours.

RPs who maintain a local cache of the distributed RPKI repository
MUST perform a local cache synchronization operation against the
distributed RPKI repository at regular intervals of no longer than 24
hours.

4.  Reissuing Certificates and RPKI Signed Objects

   This section provides rules a CA MUST use when it reissues
   subordinate certificates and RPKI signed objects [RFC6488] as part of
   the key rollover process.  Note that CRLs and manifests are not
   reissued, per se.  They are generated for each CA instance.  A
   manifest catalogues the contents of a publication point relative to a
   CA instance.  A CRL lists revoked certificates relative to a CA
   instance.  Key rollover processing for CRLs and manifests is
   described above, in Section 3.

4.1.  CA Certificates

   When a CA, as part of the key rollover process, reissues a CA
   certificate, it copies all of the field and extension values from the
   old certificate into the new certificate.  The only exceptions to
   this rule are that the notBefore value MAY be set to the current date
   and time, and the certificate serial number MAY change.  Because the
   reissued CA certificate is issued by a different CA instance, it is
   not a requirement that the certificate serial number change in the
   reissued certificate.  Nonetheless, the CA MUST ensure that each
   certificate issued under a specific CA instance (a distinct name and
   key) contains a unique serial number.

4.2.  RPKI Signed Objects

   An RPKI signed object is a Cryptographic Message Syntax (CMS) signed-
   data object, containing an EE certificate and a payload (content)
   [RFC6488].  When a key rollover occurs, the EE certificate for the
   RPKI signed object MUST be reissued, under the key of the NEW CA.  A
   CA MAY choose to treat this EE certificate the same way that it deals
   with CA certificates, i.e., to copy over all fields and extensions,
   and MAY change only the notBefore date and the serial number.  If the
   CA adopts this approach, then the new EE certificate is inserted into
   the CMS wrapper, but the signed context remains the same.  (If the
   signing time or binary signing time values in the CMS wrapper are
   non-null, they MAY be updated to reflect the current time.)
   Alternatively, the CA MAY elect to generate a new key pair for this
   EE certificate.  If it does so, the object content MUST be resigned
   under the private key corresponding to the EE certificate.  In this
   case, the EE certificate MUST contain a new public key and a new
   notBefore value, and it MAY contain a new notAfter value, but all
   other field and extension values, other than those relating to the

   digital signature and its associated certificate validation path,
   remain unchanged.  If the signing time or binary signing time values
   in the CMS wrapper are non-null, they MAY be updated to reflect the
   current time.

   As noted in Sections 2.1.6.4.3 and 2.1.6.4.4 of [RFC6488], the
   presence or absence of the signing-time and/or the binary-signing-
   time attribute MUST NOT affect the validity of the RPKI signed
   object.

5.  Security Considerations

   No key should be used forever.  The longer a key is in use, the
   greater the probability that it will have been compromised through
   carelessness, accident, espionage, or cryptanalysis.  Infrequent key
   rollover increases the risk that the rollover procedures will not be
   followed to the appropriate level of precision, increasing the risk
   of operational failure of some form in the key rollover process.
   Regular scheduling of key rollover is generally considered to be a
   part of a prudent key management practice.  However, key rollover
   does impose additional operational burdens on both the CA and the
   population of RPs.

   These considerations imply that in choosing lifetimes for the keys it
   manages, a CA should balance security and operational impact (on
   RPs).  A CA should perform key rollover at regularly scheduled
   intervals.  These intervals should be frequent enough to minimize the
   risks associated with key compromise (noted above) and to maintain
   local operational proficiency with respect to the key rollover
   process.  However, key lifetimes should be sufficiently long so that
   the (system-wide) load associated with key rollover events (across
   the entire RPKI) does not impose an excessive burden upon the
   population of RPs.  RPs are encouraged to maintain an accurate local
   cache of the current state of the RPKI, which implies frequent
   queries to the RPKI repository system to detect changes.  When a CA
   rekeys, it changes many signed objects, thus impacting all RPs.

6.  Acknowledgements

   The authors would like to acknowledge the review comments of Tim
   Bruijnzeels and Sean Turner in preparing this document.

7.  References

7.1.  Normative References

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

   [RFC3779]   Lynn, C., Kent, S., and K. Seo, "X.509 Extensions for IP
               Addresses and AS Identifiers", RFC 3779, June 2004.

   [RFC5280]   Cooper, D., Santesson, S., Farrell, S., Boeyen, S.,
               Housley, R., and W. Polk, "Internet X.509 Public Key
               Infrastructure Certificate and Certificate Revocation List
               (CRL) Profile", RFC 5280, May 2008.

   [RFC6480]   Lepinski, M. and S. Kent, "An Infrastructure to Support
               Secure Internet Routing", RFC 6480, February 2012.

   [RFC6481]   Huston, G., Loomans, R., and G. Michaelson, "A Profile for
               Resource Certificate Repository Structure", RFC 6481,
               February 2012.

   [RFC6487]   Huston, G., Michaelson, G., and R. Loomans, "A Profile for
               X.509 PKIX Resource Certificates", RFC 6487, February
               2012.

7.2.  Informative References

   [RFC6488]   Lepinski, M., Chi, A., and S. Kent, "Signed Object
               Template for the Resource Public Key Infrastructure
               (RPKI)", RFC 6488, February 2012.

Authors' Addresses

   Geoff Huston
   APNIC

   EMail: gih@apnic.net
   URI:   http://www.apnic.net


   George Michaelson
   APNIC

   EMail: ggm@apnic.net
   URI:   http://www.apnic.net


   Stephen Kent
   BBN Technologies
   10 Moulton St.
   Cambridge, MA  02138
   USA

   EMail: kent@bbn.com